servicenow

Zurich Platform のセキュリティ

最終更新日: **2025年11月10日**

自動翻訳

これらの資料は、翻訳ソフトウェアを使用してお客様の便宜のために翻訳されています。正確な翻訳をご提供できるよう相当な努力を払っておりますが、いかなる自動翻訳も人間の翻訳者に代わることはなく、そのようなことは意図されておりません。翻訳は「現状のまま」提供されています。他言語への翻訳の的確性、信頼性または正確性については、明示または黙示を問わず、いかなる保証も行われません。翻訳ソフトには限界があるため、一部のコンテンツが正確に翻訳されていない場合があります。これらの資料の公用言語は英語です。翻訳の際に生じる相違または不一致は、コンプライアンスまたは履行の目的に関しては拘束力を有さず、法的効力はないものとします。

ここで示したいくつかの例と図は、説明の目的でのみ提供されています。ServiceNow 製品またはサービスへの実際の関連付けやつながりは意図されておらず、推測されるものではありません。

下記の ServiceNow ウェブサイト利用規約をお読みください。 www.servicenow.com/terms-of-use.html

自動翻訳

本社
2225 Lawson Lane
Santa Clara, CA 95054
United States
(408) 501-8550

servicenow

目次

自動翻訳

自動翻訳

自動翻訳

自動翻訳

自動翻訳

自動翻訳

インスタンスの保護

プラットフォームセキュリティは、インスタンスを保護する機能を提供します。

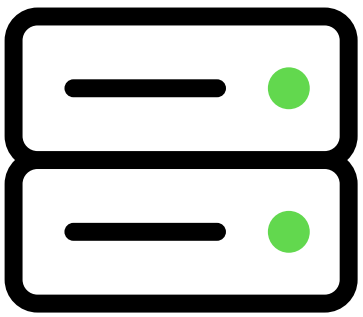
ServiceNow Vault

ライフサイクル全体に渡って機密情報を不正なアクセス、破損、または盗難から保護するデータセキュリティツールの ServiceNow Vault 製品セットを使用します。

プラットフォーム セキュリティ

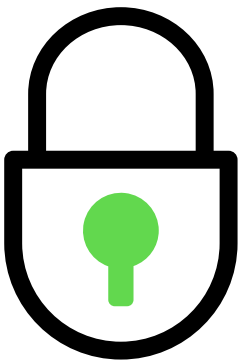
プラットフォームセキュリティは、インスタンスを保護する機能を提供します。

プラットフォームプライバシー



プラットフォームプライバシーは、インスタンス上の機密データをマスクできるようにします。

暗号化



機密データを保護し、規制要件や標準への準拠を維持します。

アクセス管理

アクセス管理を使用すると、ServiceNow インスタンスに安全にアクセスできます。

ID

インスタンス内の ID の詳細を確認します。

自動翻訳

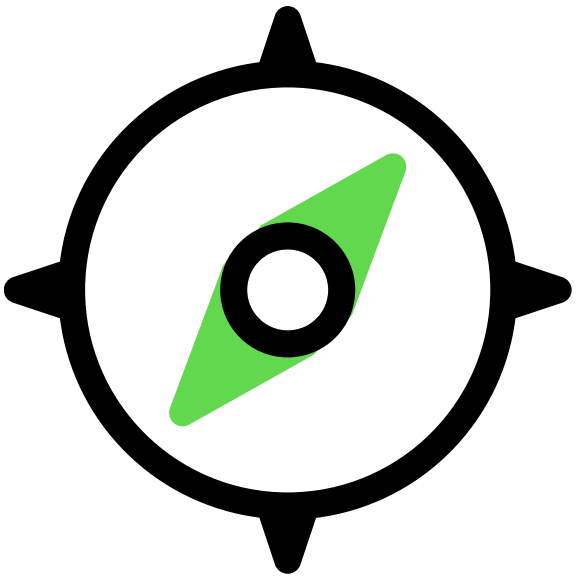
	プラットフォームセキュリティ製品およびソリューションに関するその他のリソース  学習用のその他の Platform Security リソースを確認します。	

ServiceNow Vault

ServiceNow Vault 単一の場所で、高度なセキュリティとプライバシーの要件に対応するセキュリティツールをレビューおよび実装できます。

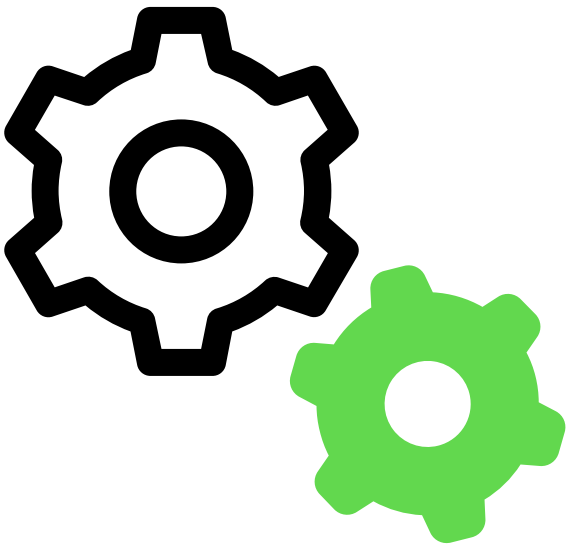
開始するには

ServiceNow Vault の探究



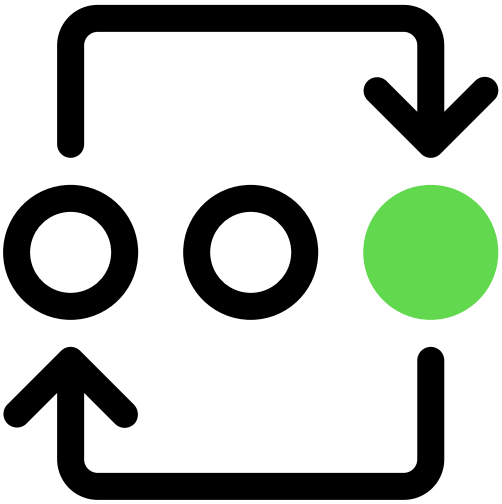
ServiceNow Vaultとその機能について説明します。

ServiceNow Vault の構成



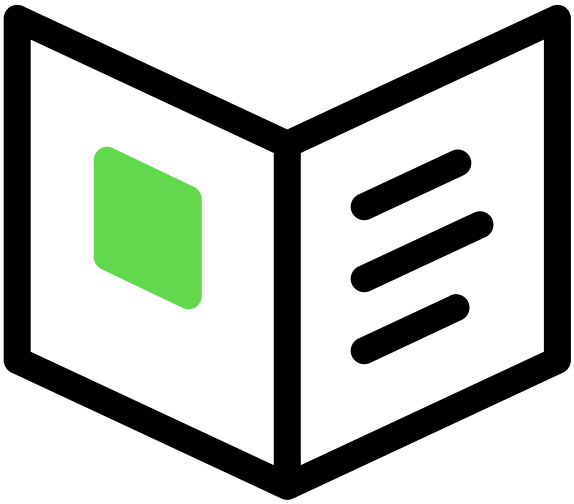
インストールと構成の開始 ServiceNow Vault

ServiceNow Vault ダッシュボード



ServiceNow Vaultダッシュボード を使用してデータセキュリティを 確認する方法について説明します。

ServiceNow Vault tools



で使用されるツールの詳細 ServiceNow Vault

ServiceNow Vault の探索

ServiceNow Vault の詳細と、ServiceNow Vault がデータ保護のニーズにもたらすメリットをご確認ください。

ServiceNow Vaultの概要

ServiceNow Vault は、ライフサイクル全体に渡って機密情報を不正なアクセス、破損、または盗難から保護する一連のデータセキュリティツールです。ServiceNow Vaultを使用して、機密情報の編集と監査のための暗号化、シークレット管理、データプライバシーなどの保護を適用できます。

注:

ServiceNow Vault は、担当者 ServiceNow 本番インスタンスで有効にする必要がある有料プラグインです。 ServiceNow Vault には、ここにリストされているコンポーネントが含まれています。

サブスクリプションを購入するには、ServiceNow アカウントマネージャーにお問い合わせください。サブスクリプションを購入すると、特定のプラグインが自動的にアクティブになります。有料プラグインが自動的にアクティブになっていない場合は、インスタンスの [すべてのアプリケーション] リストから手動でアクティブ化できます。

自動翻訳

ServiceNow Vault のメリット

ServiceNow Vaultのメリット

メリット	機能	ユーザー
包括的な ServiceNow Vault の概要を取得し、データの検出、分類、保護に関連するメトリクスを確認します	ServiceNow Vault コンソールダッシュボード	プラットフォームアドミン
アプリケーションで ServiceNow Vault ツールの使用を簡単に開始するためのガイド付きセットアップ	Guided Vault	プラットフォームアドミン

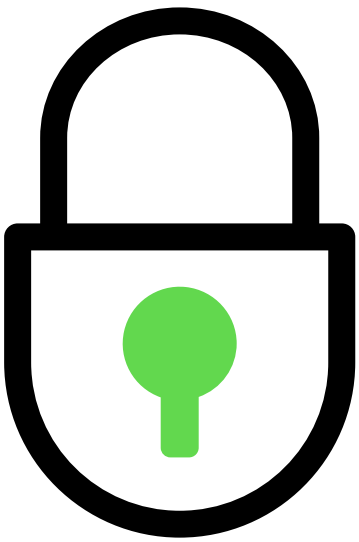
次に探索する内容

ServiceNow Vaultを構成して使用する方法の詳細については、以下を参照してください。

- ServiceNow Vault コンソールダッシュボード
- ServiceNow Vaultの構成
- ServiceNow Vault のガイド付きセットアップを使用する

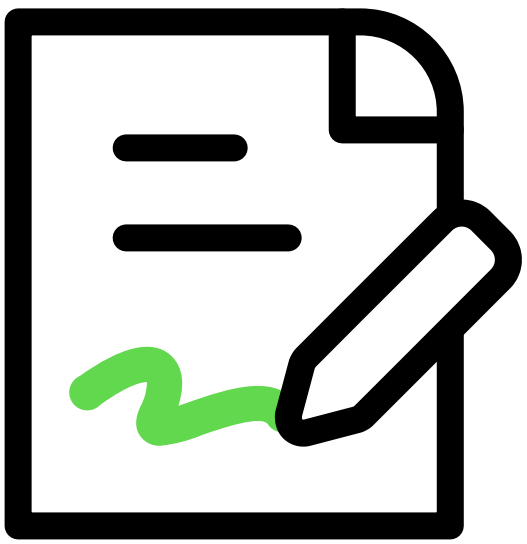
ServiceNow Vault tools

暗号化



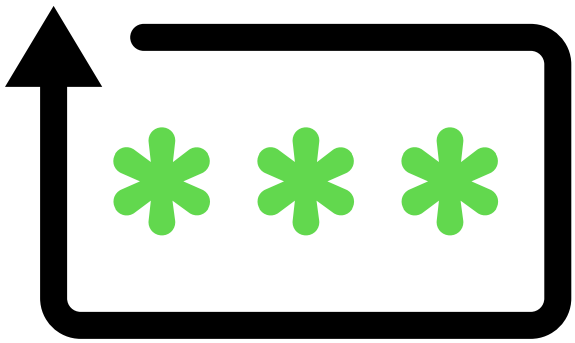
キー管理とフィールド暗号化は、高度に構成可能な暗号化モジュールのスイートです

コード署名



機密性の高いアプリケーション構成データとスクリプトを使用する前に検証できるので、セキュリティの強化に役立ちます。

データプライバシー



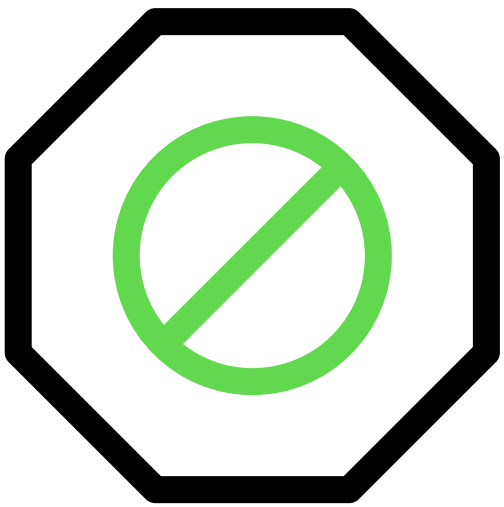
データプライバシープラグインを使用して、本番インスタンスから非本番インスタンスに移行されるときにユーザーデータから個人識別可能情報 (PII) を削除します。

データディスカバリー

データディスカバリープラグインを使用すると、ユーザーデータから個人識別可能情報 (PII) を見つけることができます。このデータを分類して、さらにセキュリティ対策を講じることができます。

ログ

Zero Trust アクセス



ServiceNow セッションアクセスを使用すると、組織は Web セッションのユーザー権限を動的に減らすことができます

自動翻訳

ログエクスポートサービスを使用して、ServiceNow ログデータをエンタープライズログ分析にインポートすることで、セキュリティ、パフォーマンス、およびユーザーエクスペリエンスを向上させます。	

ServiceNow Vaultの構成

インストールと構成の方法を学ぶ ServiceNow Vault

構成の概要

- Vault プラグインのインストール

  インスタンスに ServiceNow Vault をインストールする方法について説明します。

-

  を使用するために必要なロールについて学習し、セットアップします。 ServiceNow Vault

- Vault プラグインのインストール

  ServiceNow Vaultを最大限に活用するために必要なプラグインをインストールして検証します。

ガイド付きセットアップを使用した **ServiceNow Vault** の実装

ガイド付きセットアップでは、ServiceNow インスタンスで ServiceNow Vault を構成するのに役立つ一連のタスクを提供しています。ServiceNow Vaultのガイド付きセットアップを開くには、次の場所に移動します すべて > **Vault** > **Vault** コンソール.

詳しくは、「Guided Vault」を参照してください。

Vault プラグインのインストール

ServiceNow Vault の nessecary プラグインを確認してインストールする方法について説明します。

始める前に
必要なロール：admin

このタスクについて
ServiceNow Vault は他の ServiceNow セキュリティ製品と併用するのが最適です。他のプラグインをインストールして構成し、ツールとメトリクスを最大限に活用します。

手順

1. 移動先 すべて > **Vault** > **Vault** コンソール.
2. ツールのステータスを確認します。

ツールステータス

ステータス	説明
Limited access	ツールには制限付きアクセスライセンスがあります。報告されたメトリクスは影響を受けるか、利用できない場合があります。

ステータス	説明
	ツールはデフォルトでプラットフォームに含まれています
	ツールにライセンスがありません。報告されたメトリクスは影響を受けるか、利用できない場合があります。
	ツールにはフルアクセス権があります。

3. オプション: ステータスが制限付きであるか、ライセンスが必要な場合は、それぞれのインストールページを確認してください。

ツールのインストール

ツール	インストールページ
データディスカバリー	データディスカバリー のアクティブ化
データ分類	プラットフォームに付属
データの匿名化	データプライバシーのアクティブ化 注: データの匿名化は、 データプライバシー
フィールド暗号化	フィールド暗号化 のアクティブ化
○ ゼロトラストアクセス (ZTA) ○ 継続認証 (CA)	○ Zero Trust Access のアクティブ化 ○ 継続認証のアクティブ化

ServiceNow Vault のガイド付きセットアップを使用する

ガイド付きセットアップを使用すると、ServiceNow Vault でアプリケーションの使用を簡単に開始できます。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > **Vault** > **Vault** コンソール.
2. **[Guided vault]** セクションで、アプリケーションの [開始] ボタンを選択します。
3. [アプリ データの選択] で、テーブルを使用してServiceNow Vaultで使用するアプリケーションデータを選択し、[完了したらクラスをプレビューする] を選択します

アプリデータテーブルフィールドを選択

ラベル	説明
テーブル	データが配置されているテーブル。
列	データが配置されている列

ラベル	説明
既存のクラス	データの現在の分類
推奨クラス	データの推奨分類。
アプリケーション	データのアプリケーションスコープ

4. [データ分類のプレビュー] で、データ分類設定をプレビューし、完了したら、 [推奨クラスの適用] ボタンをオンにして同意します。

データ分類テーブルのフィールドのプレビュー

ラベル	説明
テーブル	データが配置されているテーブル。
列	データが配置されている列
最終クラス	データが割り当てられるクラス
アプリケーション	データのアプリケーションスコープ

5. [分類の概要] で、データ分類の結果を確認し、完了したら [次へ] を選択します。

Trouble?
データ分類を使用して、分類に失敗したデータを確認します。

6. [既存データの保護] で、データの保護ポリシーを確認します。

既存のデータテーブルフィールドを保護

ラベル	説明
テーブル	データが配置されているテーブル。
列	データが配置されている列
匿名化	データの匿名化ポリシーがデータに適用されているかどうか、または既にデータに適用されているかどうかを報告します。
フィールド暗号化	フィールド暗号化ポリシーがデータに適用可能かどうか、または既にデータに適用されているかどうかを報告します。
ゼロトラストアクセス	Zero Trust アクセスポリシーがデータに適用可能かどうか、または既に適用されているかどうかを報告します。

7. オプション: [利用可能] を選択すると、該当する列の適用を開始できます。詳細については、Vault のoolsとメトリクス を確認してください。
8. [リアルタイム データの保護] で、リアルタイム データ保護ポリシーを確認し、完了したら [完了としてマーク] を選択します。

リアルタイムデータフィールドを保護

ラベル	説明
名前	???の名前

ラベル	説明
タイプ	データチャネルタイプ。
リアルタイム匿名化	リアルタイム匿名化ポリシーが利用可能かどうか、または既にデータに適用されているかどうかを報告します。

注: リアルタイムデータは、インスタンス全体とすべてのアプリケーションで保護されます。

結果
選択したアプリケーションに、分類されたデータと保護ポリシーが追加されました。また、関連するメトリクスを ServiceNow Vault コンソールダッシュボードに報告するようになりました。

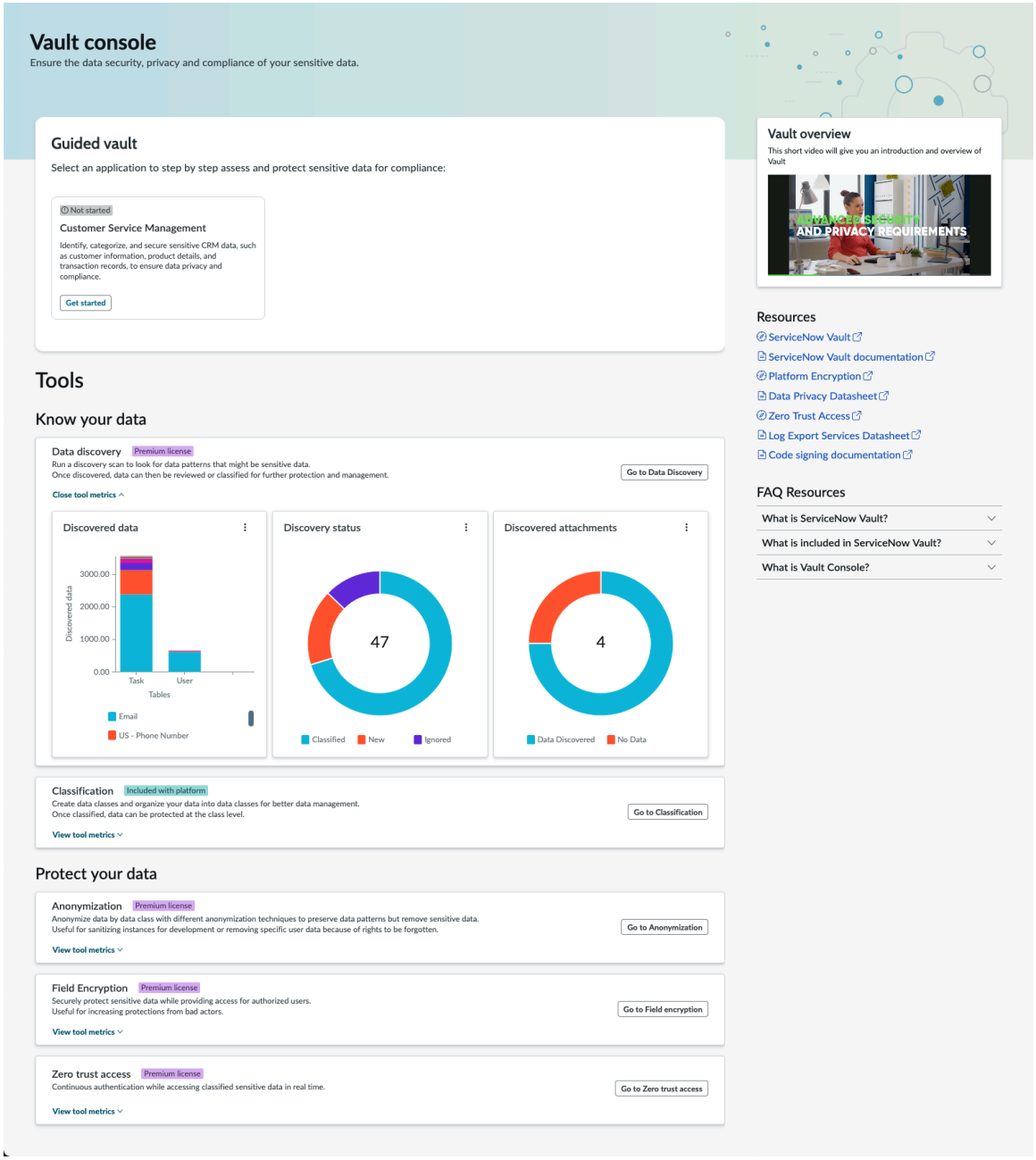
ServiceNow Vault コンソールダッシュボード

ServiceNow Vault コンソールダッシュボードを使用して、ServiceNow Vault セキュリティツールを追跡および管理します。

ダッシュボードを使用すると、機密データのセキュリティ、プライバシー、コンプライアンスを簡単に確認できます。ダッシュボードは、 ServiceNow Vault ツールとそのさまざまなメトリクスに関するレポートに加えて、 ServiceNow Vault互換性のあるアプリケーションのガイド付きセットアップを表示します。

ServiceNow Vaultコンソールダッシュボードにアクセスするには、次に移動します: すべて > **Vault** > **Vault** コンソール

Vault コンソールダッシュボードページ



セクション	サブセクション	説明
Vault の概要とリソース	Vault の概要	の概要をビデオで紹介します ServiceNow Vault
	リソース	使用を支援するための追加リソースを提供 ServiceNow Vault
	FAQ リソース	に関するよくある質問への回答を得る ServiceNow Vault
Guided Vault	アプリケーションのガイド付きセットアップカード	アプリケーションカードの [ようこそ] ボタンを選択して、そのアプリケーションで ServiceNow Vault の使

自動翻訳

セクション	サブセクション	説明
		用を開始します。ガイド付きセットアップの詳細については、こちらをご覧ください。ServiceNow Vault のガイド付きセットアップを使用する
Vault のoolsとメトリクス	ツール情報	ServiceNow Vaultで使用されるツールの簡単な説明とライセンス情報。[移動先] ボタンを選択し、ツールのホームページに移動します。現在ダッシュボードに表示されているツールは次のとおりです。 • データディスカバリー • データ分類 • データの匿名化 • フィールド暗号化 • ゼロトラストアクセス (ZTA)
	ツールメトリクス	ドロップダウンを選択して、ツールに関するさまざまなメトリクスとグラフを確認します。 注: メトリクスの詳細については、それぞれのツールのホームページを確認してください。

Guided Vault

ガイド付き Vault セットアップでは、アプリケーションの機密データを評価して保護するためのステップバイステップのプロセスが提供されます。

ガイド付き Vault セットアップを使用すると、アプリケーションで ServiceNow Vault の使用を簡単に開始できます。セットアップでは、選択したアプリケーションデータが使用されます。セットアップは、まずユーザーが分類のためにアプリケーションデータを確認し、次に分類されたデータに保護ポリシーを実装するのに役立ちます。ServiceNow Vaultガイド付きセットアップの使用方法については、以下を参照してください。ServiceNow Vault のガイド付きセットアップを使用する

ガイド付きセットアップには、次の 5 つのステップがあります。

1. アプリデータを選択
2. データ分類をプレビュー
3. 分類のサマリー
4. 既存のデータを保護
5. リアルタイムデータを保護

ガイド付きボールトのセットアップ後、ボールトコンソールダッシュボードの ツール セクションでアプリケーションデータメトリクスを確認できます。詳細については、「Vault のoolsとメトリクス」を参照してください。

Vault のoolsとメトリクス

機密データを保護および検出するために ServiceNow Vault 使用するツールとメトリクスについて説明します。

ServiceNow Vault は複数のツールと統合して、機密データのセキュリティのまとまりのある概要を提供します。ウィジェットにカーソルを合わせると、報告されたデータをさらに詳しく把握できます。任意のツールの **Go to** ボタンを選択して、それぞれのページに移動します。

データを把握する

ServiceNow Vault は データディスカバリー と データ分類 を使用して、データを理解して把握するのに役立ちます。

ツールとメトリクス

ツール	メトリクス	説明
Discovery (ディスカバリー) データディスカバリーを使用してディスカバリースキャンを実行し、機密データである可能性があるデータパターンを探します。検出されると、データをレビューまたは分類して、さらに保護および管理できます。	検出されたデータ	検出されたデータの詳細を示す棒グラフ。
	ディスカバリーステータス	検出されたデータステータスの割合
	検出された添付ファイル	検出されたデータを含む添付ファイルの割合。
分類 データ分類 データクラスを作成し、データをデータクラスに整理して管理を改善します。分類されたデータはクラスレベルで保護できます。	分類可能なデータ	分類可能なデータの割合
	分類済みデータ	分類済みデータの割合

データの保護

ServiceNow Vault では、データの匿名化、フィールド暗号化、およびゼロトラストアクセスを使用して、データのセキュリティ保護に役立てます。

ツールとメトリクス

匿名化 さまざまな匿名化手法を使用してデータクラスごとにデータを匿名化し、データパターンを保持しますが、機密データは削除します。開発のためにインスタンスをサニタイズしたり、権限を忘れてしまうために特定のユーザーデータを削除したりする場合に便利です。メトリクスは次のとおりです。	リアルタイムグラフ??	
	リアルタイムデータ	匿名化されたリアルタイムデータの量を示す棒グラフ
	匿名化実行時間	

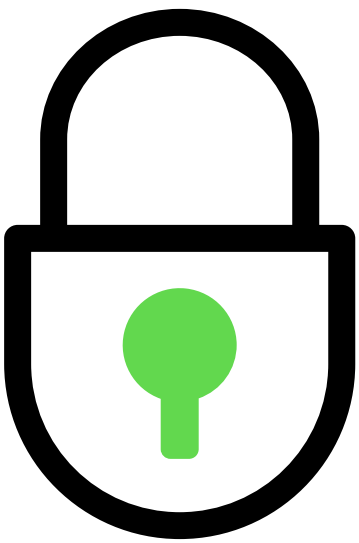
自動翻訳

ツールとメトリクス (続く)

フィールド暗号化 許可されたユーザーにアクセスを提供しながら、機密データを安全に保護します。攻撃者からの保護を強化するのに役立ちます。	暗号化フィールドの分類ステータス	暗号化フィールドステータスの割合
	フィールド暗号化で保護されているクラス	フィールド暗号化で保護されているクラスのチャート。
	アクティブな暗号化キー	アクティブな暗号化キーの数を表示します。 注: このデータを表示するには、 キー管理フレームワーク admin ロールが必要です
ゼロトラストアクセス (ZTA) 分類された機密データにリアルタイムでアクセスする際の継続認証。	継続認証の分類ステータス	継続的認証分類ステータスの割合
	継続認証で保護されているクラス	継続的な認証で保護されているクラスの数をクラス別に表示します。

すべての **ServiceNow Vault** ツール

暗号化



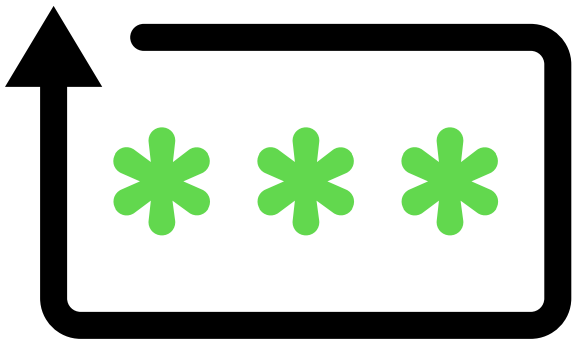
キー管理とフィールド暗号化は、高度に構成可能な暗号化モジュールのスイートです

コード署名

機密性の高いアプリケーション構成データとスクリプトを使用する前に検証できるので、セキュリティの強化に役立ちます。

自動翻訳

データプライバシー



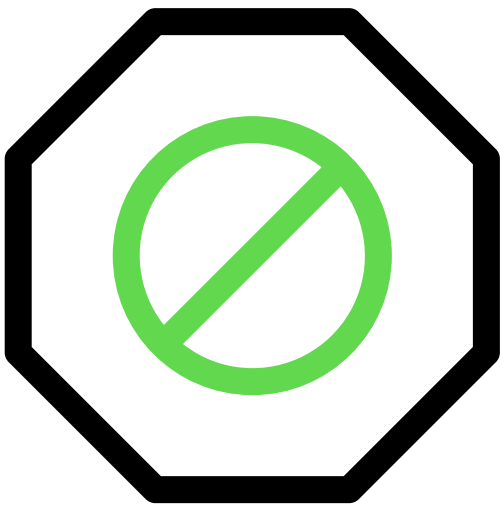
データプライバシープラグインを使用して、本番インスタンスから非本番インスタンスに移行されるときにユーザーデータから個人識別可能情報 (PII) を削除します。

データディスカバリー

データディスカバリープラグインを使用すると、ユーザーデータから個人識別可能情報 (PII) を見つけることができます。このデータを分類して、さらにセキュリティ対策を講じることができます。

ログ

Zero Trust アクセス



ServiceNow セッションアクセスを使用すると、組織は Web セッションのユーザー権限を動的に減らすことができます

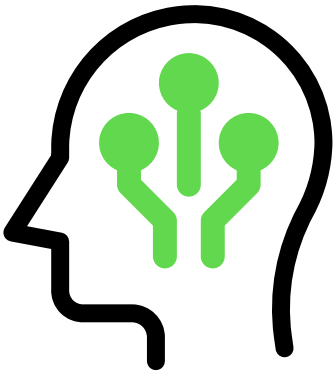
自動翻訳

ログエクスポートサービスを使用して、ServiceNow ログデータをエンタープライズログ分析にインポートすることで、セキュリティ、パフォーマンス、およびユーザーエクスペリエンスを向上させます。

プラットフォームセキュリティ

プラットフォームセキュリティは、インスタンスを保護する機能を提供します。

セキュリティセンター

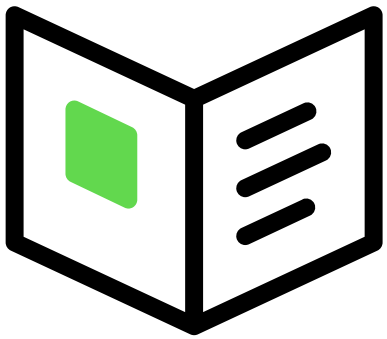


アドミニストレーターは最高レベルのセキュリティ体制を継続的に維持し、安全でないイベントや動作を簡単に監視できます。

ログエクスポートサービス

ログエクスポートサービス (LES) を使用すると、インスタンスシステムおよびアプリケーションログをエンタープライズセキュリティ分析ツールにシームレスにエクスポートできます。このサービスは、セットアップと保守が容易な分析ツールとの非常にスケーラブルでほぼリアルタイムの統合を実現します。

ログ



ログモジュールは、インスタンス内で発生するトランザクションやイベントのトラブルシューティングとデバッグに使用できるさまざまなログを提供します。

シークレット管理

ServiceNow シークレット管理を使用し、ビジネスニーズに合わせてパスワードへのアクセスを詳細に管理します。

コード署名

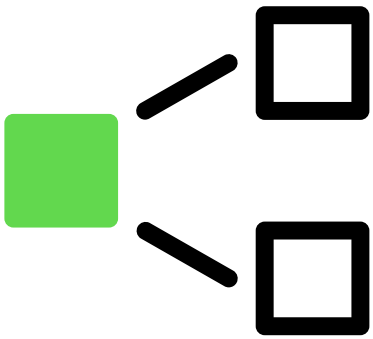
コード署名は、データの信頼性と整合性を確認するために後でチェックされるデータのデジタル署名を作成します。

アンチウイルススキャン機能

自動翻訳

アンチウイルススキャンを使用すると、インシデント、問題、ストーリーなど、システムレコードの添付ファイルによって持ち込まれる可能性のあるウイルス感染からインスタンスを保護できます。

HTML Sanitizer

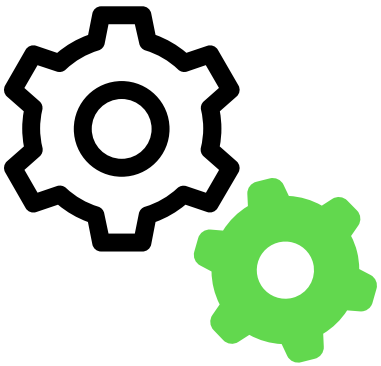


HTML フィールドと翻訳された HTML フィールドの HTML マークアップをサニタイズして不要なコードを削除し、クロスサイトスクリプト攻撃などのセキュリティ上の懸念から保護します。

監査

監査が有効にされたテーブルのレコードの変更を追跡します。デフォルトでは、インシデントテーブル、変更テーブル、および問題テーブルなどに対して行われた変更が追跡されます。

高セキュリティ設定



高セキュリティ設定とは、インスタンスで利用可能な複数のセキュリティオプションを指します。

仮想プライベートネットワーク (VPN)

ハードニング設定

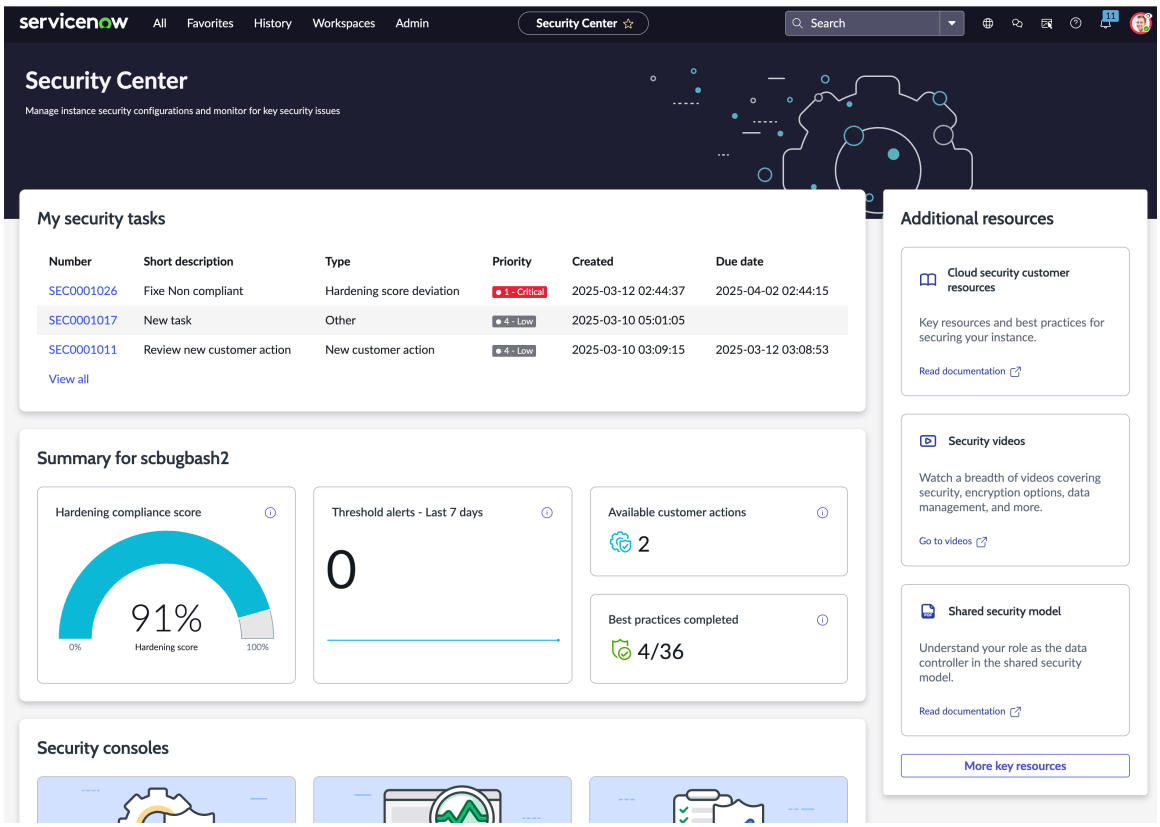
Security Center (SSC) のハードニング設定コンテンツの詳細な説明とコンプライアンス値を確認する

自動翻訳

仮想プライベートネットワーク (VPN) を使用して、インターネット経由でインスタンスを外部データソースと統合します。		

セキュリティセンター

ServiceNow セキュリティセンター は、組織が ServiceNow 展開のセキュリティを管理するのを支援するように設計された一連のツールで構成されるアプリケーションです。Security Center を使用して、シームレスなユーザーエクスペリエンスでセキュリティ体制を改善し、コンプライアンスレベルを強化できます。



Security Center は、アドミニストレーターが ServiceNow Store からダウンロードできる無料のアプリケーションです。Vancouver リリース以降、デフォルトでインストールされます。また、新機能の迅速なアダプションを促進するために、四半期ごと、ファミリーリリースの合間に ServiceNow Store でも入手可能になります。

重要:

インスタンスセキュリティセンター (ISC) は 2024 年 9 月の時点で販売が終了しており、サポートやアクティブ化は終了しています。

セキュリティセンター (SSC) が、今後推奨されるソリューションです。詳細については、「インスタンスセキュリティセンターから ServiceNow セキュリティセンターへの移行」を参照してください。

開始

セキュリティセンターのランディング

セキュリティセンターのランディングページを使用して、インスタンスを保護するために必要な情報とツールを見つけます。

セキュリティ構成コンソール

セキュリティ構成コンソールを使用すると、概要ページでインスタンスのセキュリティ体制の概要をすばやく把握できます。

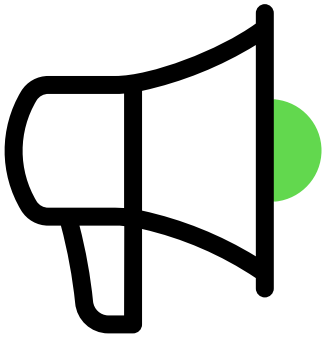
セキュリティモニタリングコンソール

セキュリティ通知とメトリクスを監視して、インスタンスの潜在的なセキュリティリスクについて常に最新情報を入手します。

セキュリティ体制コンソール

包括的な可視性と段階的な手順により、セキュリティ体制を向上させます。

自動翻訳

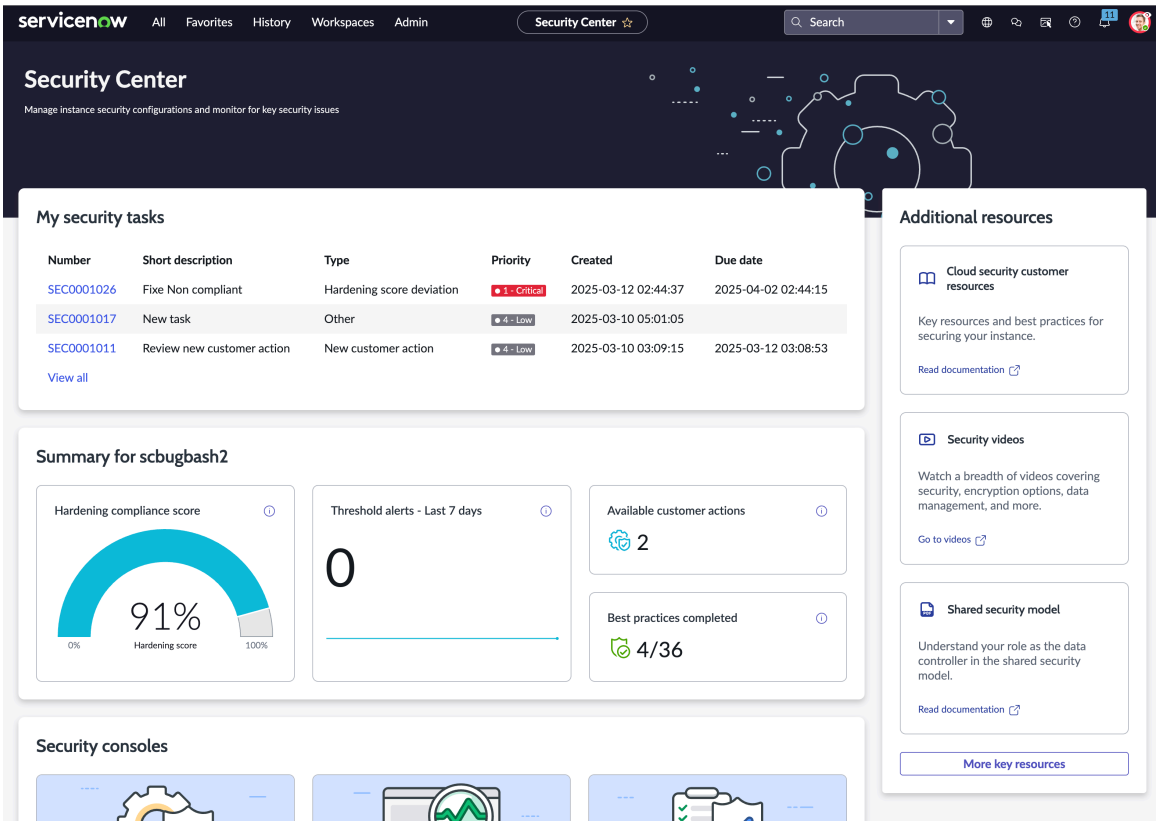
	セキュリティバナーのお知らせ  セキュリティ通知とメトリクスを監視して、インスタンスの潜在的なセキュリティリスクについて常に最新情報を入手します。	

トラブルシューティングとサポート

- について質問し、他のリソースを調べる セキュリティセンター の [Security Operations] セクション ServiceNow コミュニティ .
- 既知のエラーポータルで既知のエラー記事を検索する
- 連絡先 カスタマーサービス & サポート

セキュリティセンターのランディングページ

セキュリティセンターのランディングページを使用して、インスタンスを保護するために必要な情報とツールを見つけます。



自分のセキュリティタスク

[自分のセキュリティタスク] セクションには、自分にアサインされたセキュリティタスクのリストが表示されます。タスクの詳細を確認する番号を選択してタスクを開くか、[すべて表示] を選択すると、すべてのタスクが表示されるセキュリティタスクマネージャーにアクセスできます。セキュリティタスクの詳細については、「 セキュリティタスク」を参照してください。

インスタンスサマリー

インスタンスのサマリには、インスタンスのセキュリティステータスの概要が表示されます。カードで情報 (ⓘ) アイコンを選択すると、それぞれの意味の詳細が表示されます。任意のカードを選択して、セキュリティセンターの関連領域に移動することもできます。

ID とアクセスの管理

[Identity and Access Management (IAM)] セクションのツールを使用して、データを必要とするユーザーとプロセスのみがデータにアクセスできることを確認します。これらのツールの詳細については、「 ID とアクセスの管理」を参照してください。

セキュリティコンソール

セキュリティコンソールセクション内のカードを選択して、3 つのセキュリティコンソールのいずれかに移動します。

- セキュリティ構成コンソール
- セキュリティモニタリングコンソール
- セキュリティ体制コンソール

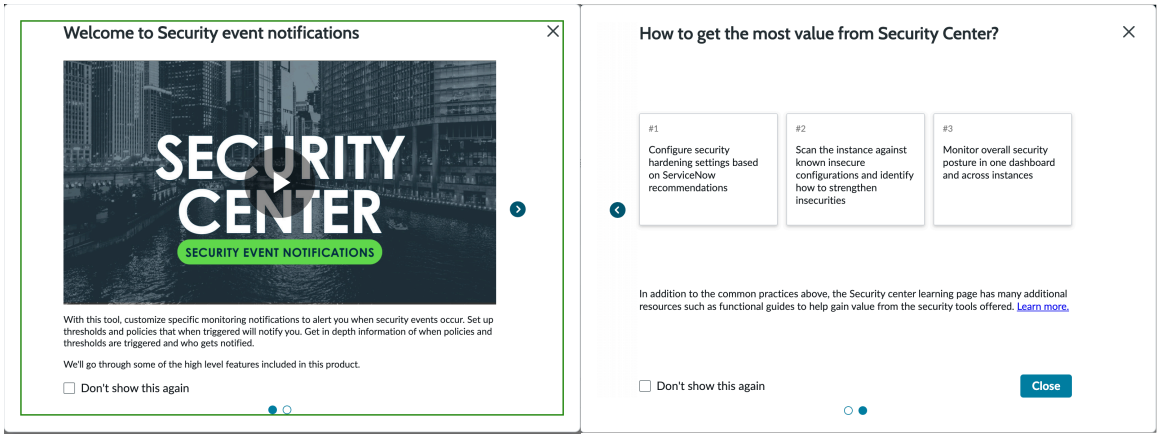
ツール

ツールセクションを使用して、インスタンスのセキュリティを管理するために必要なツールに移動します。これらのツールは、セクションの上部に表示される 3 つのタブに整理されています。カードを選択して、選択したツールに移動します。

その他のリソース

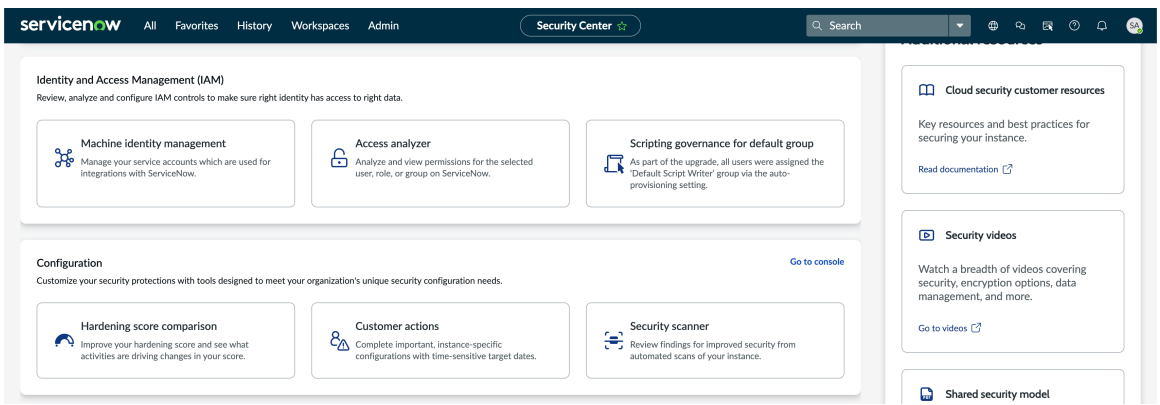
画面の端にある [追加リソース] セクションを使用して、Security Center の学習に関連するドキュメントとビデオ、およびインスタンスセキュリティの管理に関する情報に移動します。

セキュリティセンターは Zurich 年に更新され、ツールで何ができるか、その使用方法を理解するためのガイダンスが提供されました。この再設計は、セキュリティセンターの新規ユーザーと使用頻度の低いユーザーの両方を支援するように調整されています。セキュリティセンター内に新しいモーダルウィンドウが表示され、これらのツールがセキュリティ目標の達成にどのように役立つかを示します。



ID とアクセスの管理

[Identity and Access Management (IAM)] セクションのツールを使用して、データを必要とするユーザーとプロセスのみがデータにアクセスできることを確認します。



IAM は、インスタンス上のデータへのアクセスを管理するために使用できる 3 つのツールで構成されています。

マシン ID 管理

マシン ID は、証明書、キー、トークンなどのデジタル資格情報です。サーバー、アプリケーション、コンテナ、およびクラウドサービスは、これらの ID を相互に認証します。マシン **ID** コンソールを使用して、ServiceNowとの統合に使用されるマシン ID を管理します。

アクセスアナライザー

[アクセスと権限の分析 **(Analyze Access and Permissions)**] コンソールを使用して、選択したユーザー、ロール、グループの権限を表示したり、2 人のユーザー間でアクセスを比較したりします。

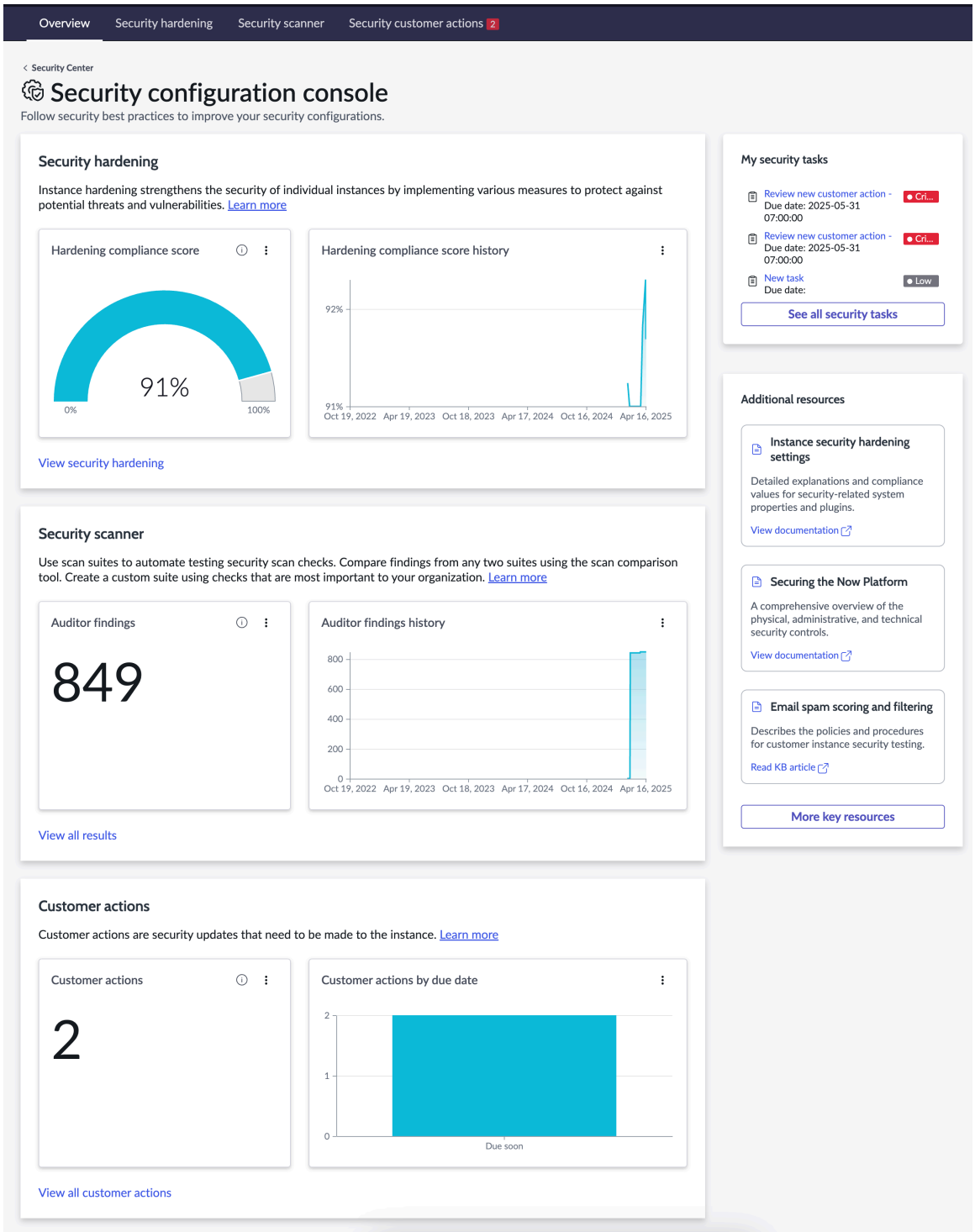
デフォルトグループのスクリプティングガバナンス

Scripting Governance ツールを使用して、条件付きスクリプトライターグループの設定を構成します。このグループのユーザーには *snc_required_script_writer_permission* ロールが割り当てられます。これにより、ユーザーはプラットフォーム全体でスクリプトやスクリプトに似たフィールドにアクセスできます。

このコンソールの設定では、このロールの自動割り当てをオンまたはオフにしたり、ユーザーをグループに手動で割り当てたりすることができます。アサインされているユーザーに関する情報を確認し、インスタンスをスキャンして特定の期間内にスクリプトを作成したユーザーを見つけることもできます。

セキュリティ構成コンソール

セキュリティ設定ページを使用して、インスタンスのセキュリティ体制の概要を把握します。ハードニングコンプライアンススコアを表示し、グラフィカルな傾向を検出し、上位の非準拠ハードニング設定を分析し、セキュリティスキャンの結果を確認します。



ナビゲーションバー

ページの上部にあるバーを使用して、このページと、セキュリティセンターのセキュリティ強化、セキュリティスキャナー、またはセキュリティ顧客アクションセクションの間を移動します。

セキュリティの強化

ハードニングコンプライアンスの詳細については、「セキュリティ強化」セクションを参照してください。このセクションのカードを選択して、[セキュリティの強化] ページに移動します。

ハードニングコンプライアンススコアは、セキュリティ設定が推奨設定にどの程度近いかを示すパーセンテージです。[セキュリティ強化] セクションには、このパーセンテージと、経時的なインスタンスのスコアの履歴が表示されます。

セキュリティスキャナー

セキュリティスキャナーセクションを使用して、自動セキュリティスキャンチェックの結果を確認します。これらのチェックは、セキュリティの脆弱性、非効率性、コンプライアンスの問題を見つけるのに役立ちます。

このセクションのカードを選択して、セキュリティスキャナーページに移動します。

顧客アクション

[顧客アクション] セクションで、インスタンスに対して実行できるセキュリティ更新を検索します。このセクションには、利用可能なアクションの数と、これらのアクションの期日を示すチャートが表示されます。このセクションのカードを選択して、[顧客アクション] ページに移動します。

自分のセキュリティタスク

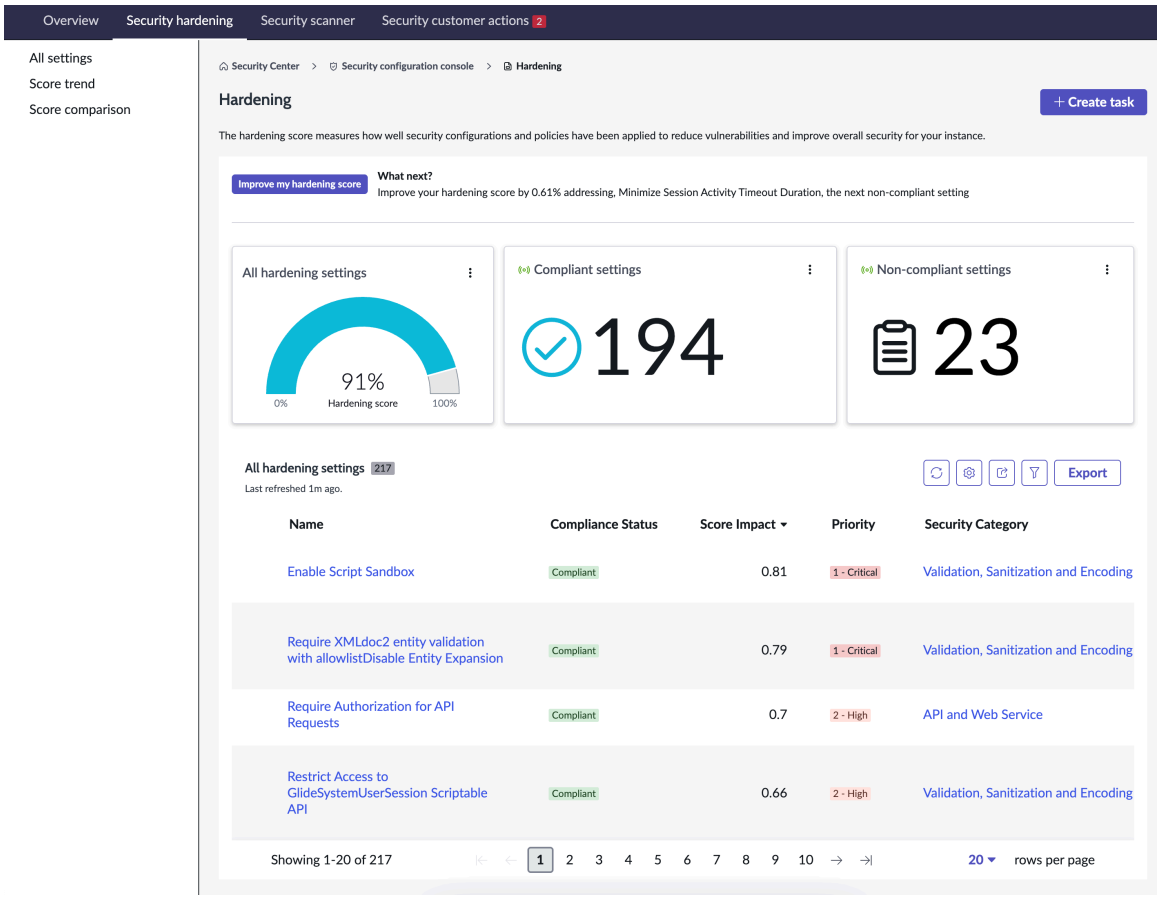
自分にアサインされた最も緊急性の高いセキュリティタスクを表示します。タスクを選択して詳細を表示するか、[すべてのセキュリティタスクを表示] を選択してセキュリティタスクの完全なリストを表示します。

その他のリソース

画面の端にある [追加リソース] セクションを使用して、Security Center の学習に関連するドキュメントとビデオ、およびインスタンスのセキュリティの管理に関する情報に移動します。

セキュリティの強化

セキュリティ強化ページで、ハードニングコンプライアンススコアを表示し、以前のスコアと比較し、コンプライアンススコアとセキュリティ体制を改善するための設定を変更します。



ハードニング設定ServiceNow AI Platformのセキュリティ関連プロパティとプラグインの推奨値を指定します。この強化ツールは、ハードニング設定のコンプライアンススコアをパーセンテージで計算します。この数値は、インスタンスが Security Center のハードニング設定にどの程度準拠しているかを示します。

ハードニングコンプライアンススコアの計算式：

- 各ハードニング設定のリスクスコアは 0 ～ 10 です。[すべての設定] セクションで、個々の設定の値を確認できます。
- スコアは、すべての準拠リスクスコアの合計をすべてのリスクスコアの合計で割った値と等しくなります。

たとえば、すべての準拠リスクスコアの合計が 25.4 で、すべてのリスクスコアの合計が 34.9 であるとします。この合計のコンプライアンススコアは (25.4 /34.9) × 100 で、これは 72.7 に相当します。この小数点は最も近い整数に切り上げられるため、73 になります。

この計算は、毎月 1 日、または ServiceNow Security Center のインストールまたは再インストール後に自動的に実行されます。このページの [スコアを更新] ボタンを使用すると、いつでも再計算をトリガーできます。

セキュリティの強化サブセクション

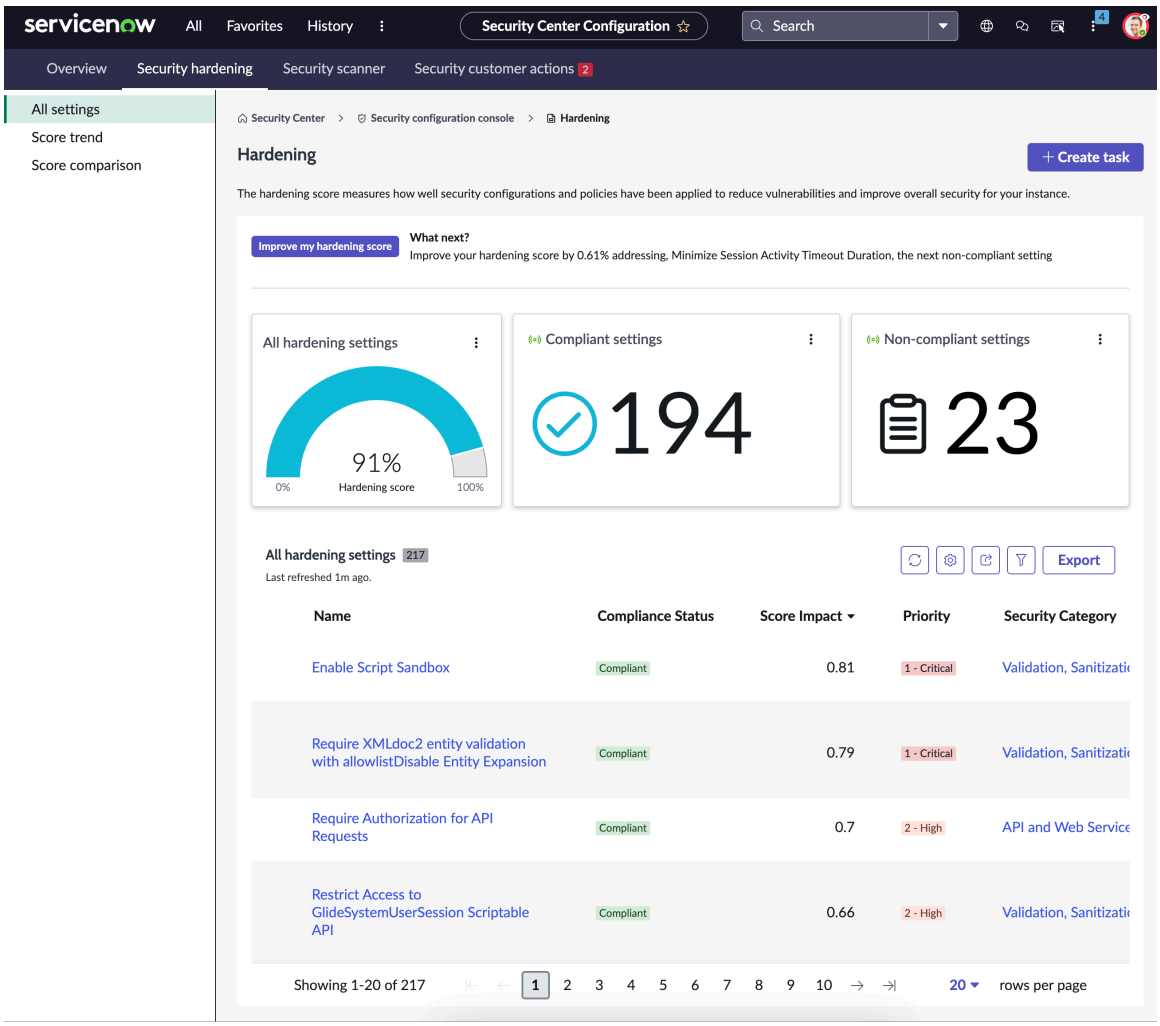
セキュリティ強化セクションには、画面の左端で選択できる 3 つのサブセクションが含まれています。

- すべての設定
- ハードニングコンプライアンススコアのトレンド
- ハードニングスコアの比較

すべての設定

単一のページで利用可能なすべてのインスタンスハードニング設定を確認します。

[すべての設定] ページには、すべてのインスタンスハードニング設定に関する情報が表示されます。右上のボタンを使用して、この情報を更新、フィルタリング、エクスポートします。\



[+ タスクを作成] ボタンを使用して、顧客アクションを完了するためのセキュリティタスクを作成します。このボタンは、[すべての設定] ページと個々のハードニング設定のページの両方に表示されます。詳細については、「セキュリティタスク」を参照してください。

ハードニング設定に関する以下の情報は、リストにあります。

名前

ハードニング設定の名前。

コンプライアンスステータス

システムが推奨に従って適切に構成されているか (準拠)、構成する必要があるか (非準拠)

スコアへの影響

このハードニング設定がセキュリティ体制に与える影響 (パーセンテージで表されます)。すべてのスコアへの影響の合計は 100% になります。

優先度

設定の重要度:重大、高、中、低。スコアが高いほど、影響度と優先度が高いことを示します。

セキュリティカテゴリ

プロパティのセキュリティカテゴリ。詳細を表示するカテゴリを選択します。

解決内容

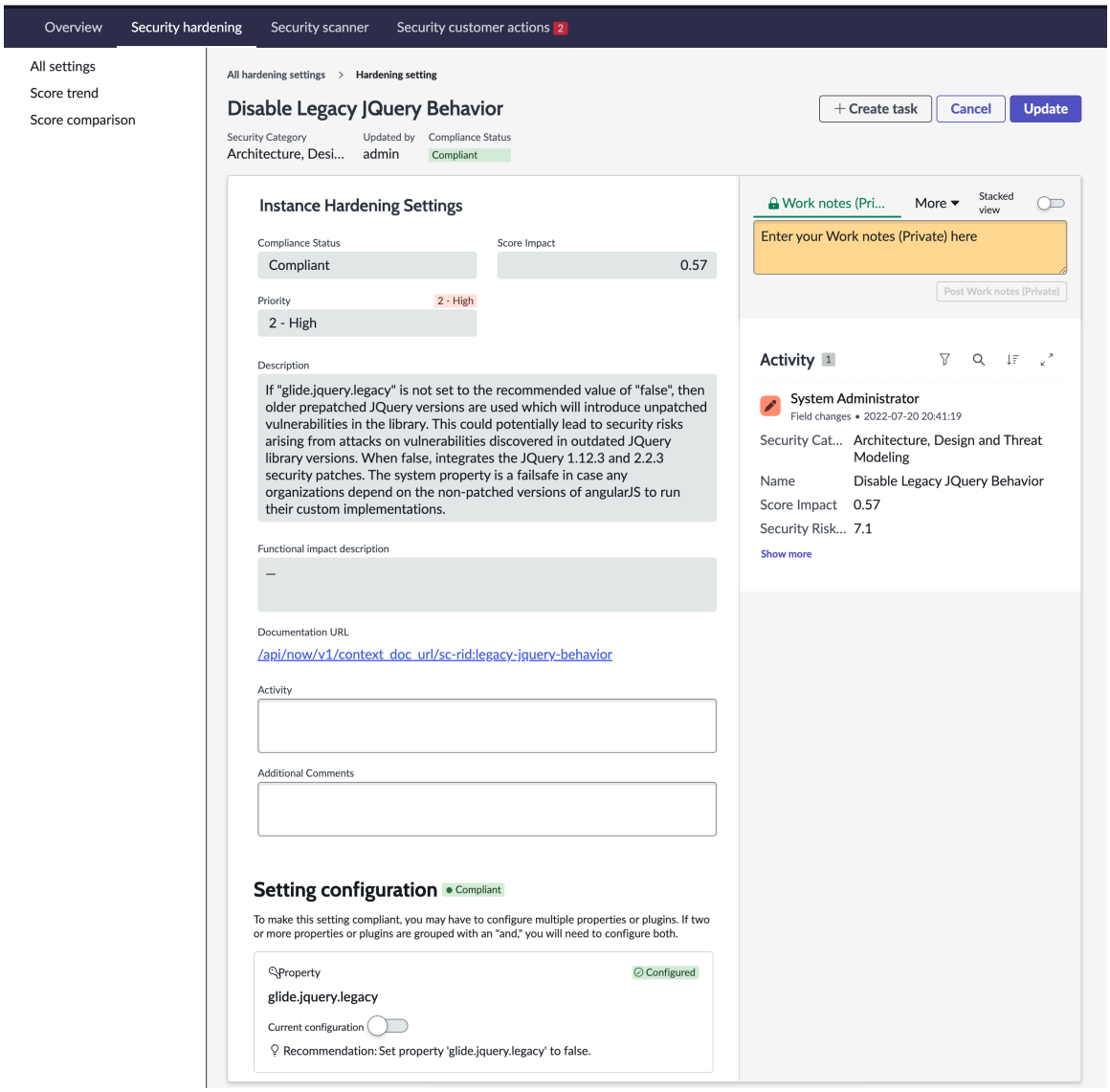
ハードニング設定のセキュリティ脆弱性を修復する手順の説明。

ハードニング設定プロパティの構成方法については、「ハードニングコンプライアンススコアの増加」を参照してください。

ハードニング設定の詳細

Security Center アプリ内のリンクを選択して、ハードニング設定の詳細を分析します。

移動先 ハードニング > すべての設定をクリックし、ハードニング設定を選択して、セキュリティ関連情報を表示するハードニング設定ツール内のページにリダイレクトします。



[+ タスクを作成] ボタンを使用して、顧客アクションを完了するためのセキュリティタスクを作成します。詳細については、「セキュリティタスク」を参照してください。

ハードニング設定構成の詳細

構成属性	説明
コンプライアンスのステータス	ハードニング設定が準拠しているかどうかを示します。
スコアへの影響	このハードニング設定がセキュリティ体制にどの程度影響するかを示すパーセンテージ。
優先度	1 ～ 4 の範囲内の数値。1 が最も高い重み付けで、このハードニング設定がセキュリティ体制に与える影響の強さを示します。
機能への影響度	このハードニング設定がインスタンスの操作に与える影響。
説明	ハードニング設定に関する一般的な概要。
ドキュメント URL	ハードニング設定のドキュメントへのリンク。
アクティビティ	ハードニング設定に関連する更新の通知。
設定構成	ハードニング設定のコンプライアンスステータスに関連する詳細と、それらを準拠させる方法の手順。 **注:** 一部のハードニング設定では、それらを準拠させるために複数のプロパティとプラグインの構成が必要になる場合があります。

ハードニング設定のフィルター

フィルターを使用してハードニングレビュープロセスを簡素化します。これらのフィルターは、レビュー用のハードニング設定の作業リストを作成し、後で使用するために復元して他のユーザーと共有することができます。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > システムセキュリティ > セキュリティセンター.
2. [セキュリティコンソール] セクションで、セキュリティ構成コンソールを選択します。
3. [セキュリティ強化] タブを選択します。
4. [**All settings**] を選択します。
5. [フィルターパネルを表示]ボタン( ])を選択します。
6. [フィルター] ウィンドウで、[詳細ビュー] を選択します。
7. 確認するハードニング設定のみを表示するフィルターを作成します。
8. フィルターが完了したら、[フィルターの保存] を選択します。

9. フィルターの名前を入力し、[保存] を選択します。

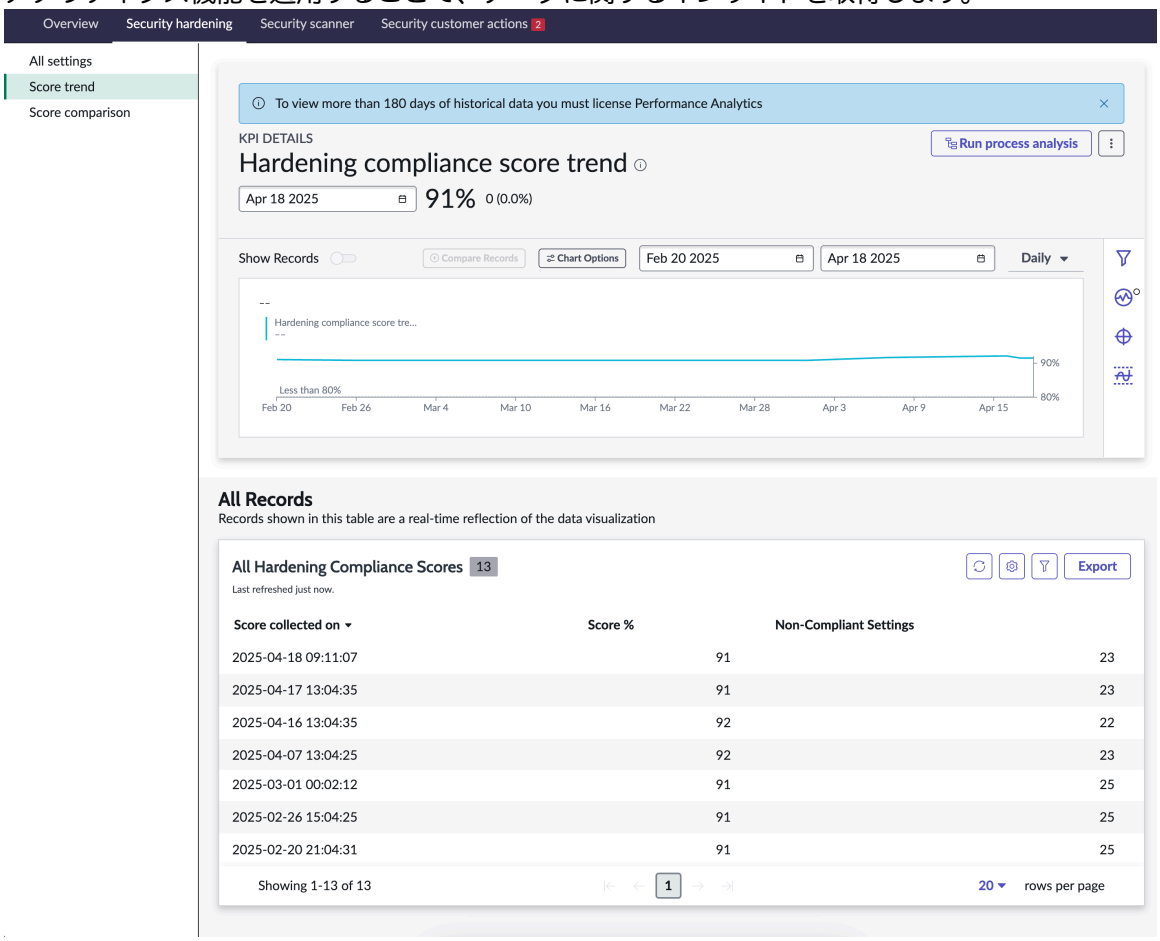
> **注:** また、アクセス許可を [すべてのユーザー] に設定したり、特定のグループと共有するグループ に設定したりして、他のユーザーがこのリストを使用できるようにすることもできます。これらのユーザーには、セキュリティセンターを使用する権限も必要です。

次のタスク
フィルターを保存した後、[既存のフィルターを使用] ボタンを使用して詳細ビューに読み込むことができます。

ハードニングコンプライアンススコアのトレンド

ハードニングコンプライアンススコアの経時的な傾向をチャートまたはテーブルで表示します。

コンプライアンススコアの経時的な傾向を確認します。日付ピッカーを使用して分析する時間範囲を選択し、ターゲット、しきい値、KPI などのパフォーマンスアナリティクス機能を適用することで、データに関するインサイトを取得します。



コンプライアンススコアチャート

[チャートオプション] ボタンを使用して、分析する情報とその情報の表示方法を指定します。

分析

オプションのリストから選択します。選択したチャートタイプによっては、一部のオプションが使用できない場合があります。

ターゲット

組織が達成したい目標です。「インジケーター ターゲット」を参照してください。

しきい値

インジケーターの通常のスコア範囲を定義し、特定のイベントが発生したときに警告します。「インジケーター しきい値」を参照してください。

予測

過去の動作に基づいて将来のスコアを予測する機能について説明します。「パフォーマンスアナリティクススコア予測」を参照してください。

傾向図

1 つ以上のアイテムの値が時間の経過とともにどのように変化するかを示します。

コメント

個々のデータポイントの注釈を表示します。

ラベル

可視化に関連するスコアを表示します。

統計情報

コンプライアンススコアに関連する統計情報を表示します。

時系列

グラフに表示するメトリクスを選択します。

スコア

重要業績評価指標 (KPI) のスコア。

変更

このインジケーターのスコアの変化。

変更割合

スコアに対する割合での変化。

グラフのタイプ

選択した情報の可視化方法を制御するチャートタイプを選択します。「さまざまな時系列可視化タイプのユースケース」を参照してください。

すべてのレコードテーブル

[すべてのレコード] テーブルには、コンプライアンススコアが収集された日付、スコアの割合、および非準拠設定の数が表示されます。このテーブルを別のオプションとして使用して、 時間の経過に伴うインスタンスのセキュリティ体制を分析できます。

ハードニングコンプライアンススコアの増加

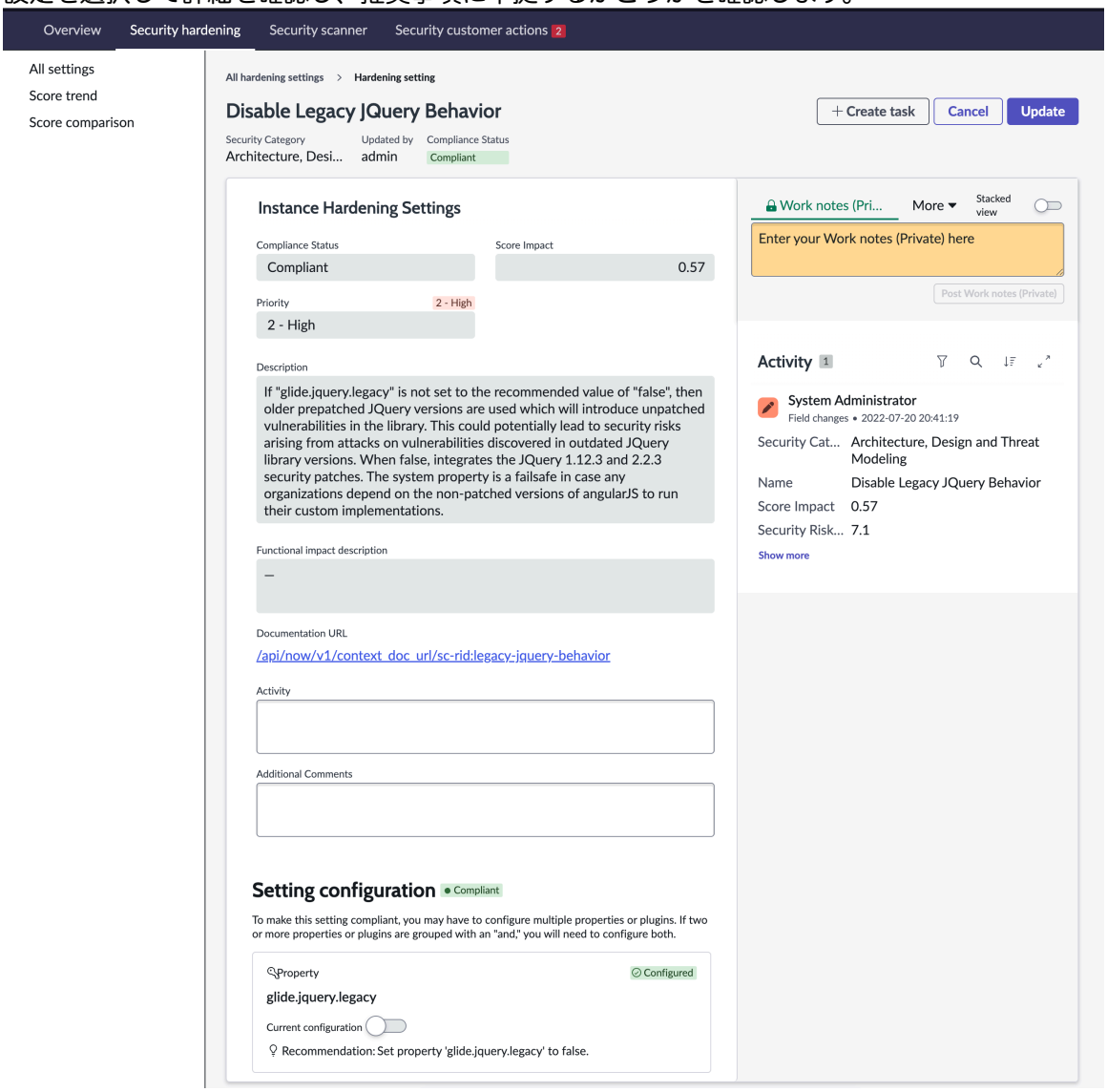
ハードニング設定がシステムの推奨事項に準拠していることを確認し、ハードニングコンプライアンススコアを増加します。

始める前に
必要なロール：admin

インスタンスでスコアへの影響が最も高い非準拠のハードニング設定を特定します。それらをレビューして、全体的なコンプライアンススコアを上げることができるように、システムの推奨事項に準拠できるかどうかを確認します。

手順

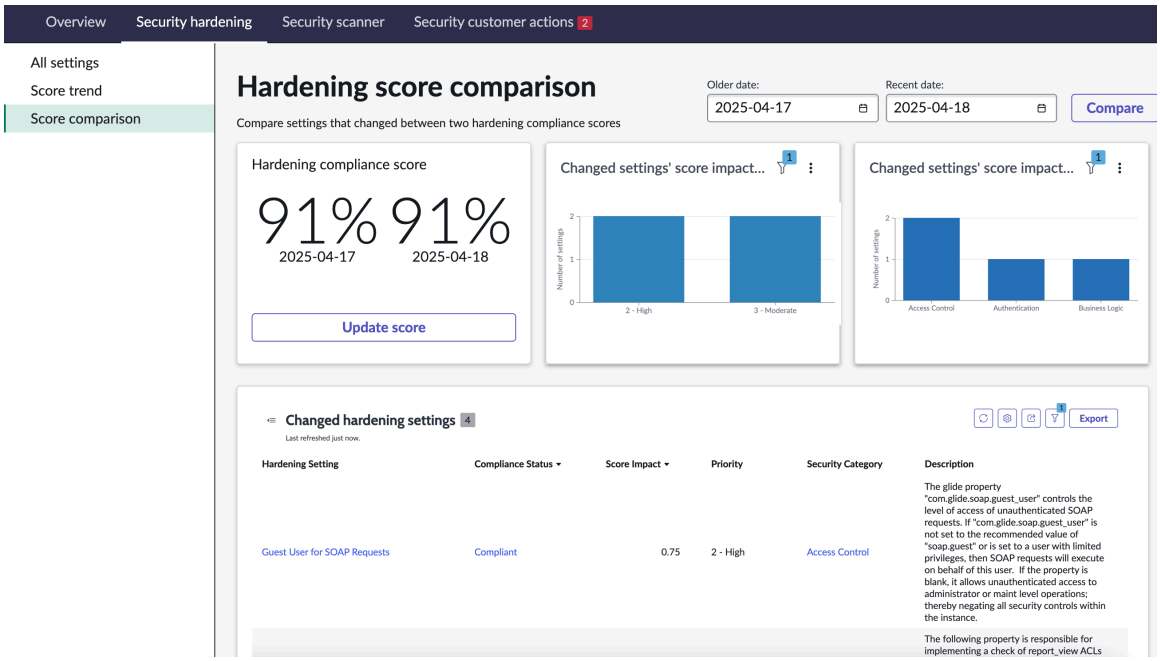
1. 移動先 ハードニング > すべての設定.
2. [コンプライアンスステータス] 列をフィルタリングして、非準拠ハードニング設定のみを表示します。
3. [スコアへの影響] を選択して、最大から最小の順に並べ替えます。
4. 設定を選択して詳細を確認し、推奨事項に準拠するかどうかを確認します。



5. ハードニング設定を準拠させます。
6. 非準拠のハードニングスコアを更新して準拠させる場合は、ホームページに移動し、[スコアを更新] を選択して最新のスコアを表示します。
   ハードニングスコアは切り上げられます。86.75% のスコアは 87 に切り上げられます。

ハードニングスコアの比較

ハードニング設定の健全性を可視化し、このデータを使用してインスタンスのセキュリティ体制を改善します。



このページには、ハードニングコンプライアンスの概要と、このスコアに影響を与えるインスタンスに加えられた変更が表示されます。

ハードニングスコアの比較ページには、[古い日付] フィールドと [最近の日付] フィールドで選択した日付の間のコンプライアンスステータスの変化を含むハードニング設定が表示されます。この情報を使用して、ハードニングコンプライアンススコアにどのような影響を与えたかを確認できます。

ハードニングコンプライアンススコア

カードは、選択した日付のインスタンスのコンプライアンススコアをパーセンテージで比較します。

優先度ごとの変更された設定のスコアへの影響

前回のスコア更新以降にコンプライアンスステータスが変更されたハードニング設定の数を優先度値別に表示します。

セキュリティエリアごとの変更された設定のスコアへの影響

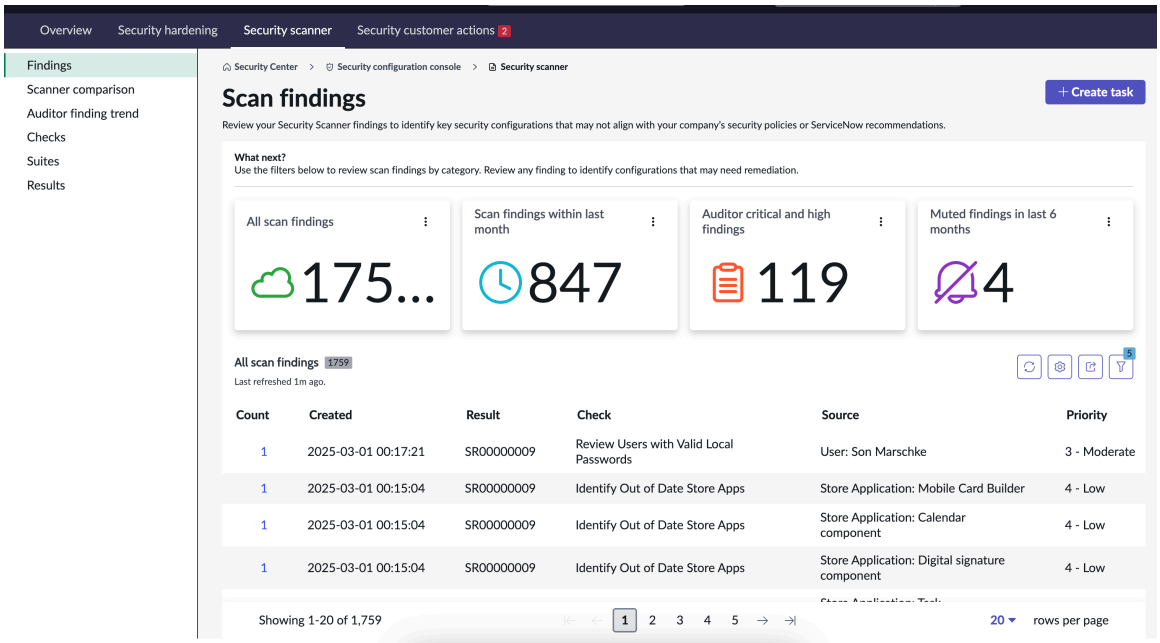
選択した 2 つの日付の間にコンプライアンスステータスを変更したハードニング設定の数を、セキュリティエリア別に表示します。

変更されたハードニング設定

選択した日付の間にコンプライアンスのステータスを変更したハードニング設定のリストを表示します。非準拠になり、ハードニング比較スコアが低下した設定を確認し、スコアを上げるためにそれらを準拠させる機会を探します。「ハードニングコンプライアンススコアの増加」を参照してください。

セキュリティスキャナー

一連のセキュリティチェックに対してインスタンスをスキャンし、構成ミスを特定します。スキャナーツールを使用すると、さまざまなユースケースに対してさまざまなチェックスイートを作成するプロセスが簡素化されるため、結果を経時的に分析できます。

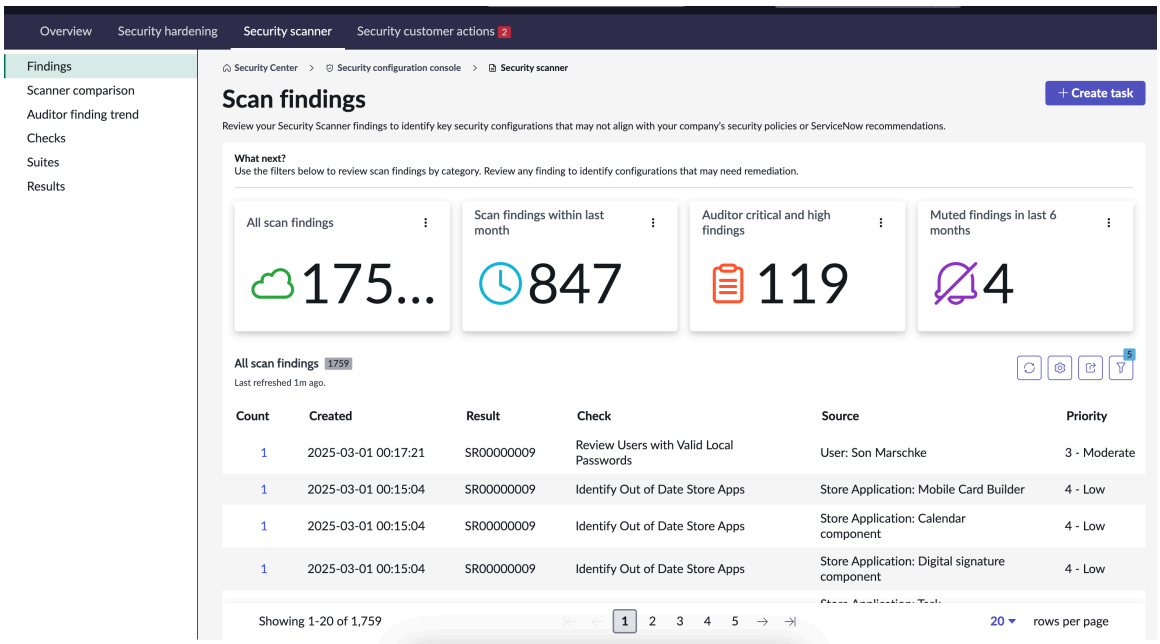


セキュリティスキャンは、インスタンスについてセキュリティの健全性の問題を示す構成を調べる方法です。この方法により、組織のセキュリティ推奨事項を実装する機会を特定できます。

スキャナーツールにアクセスする場合、スイートの少なくとも 2 つのスキャン結果と比較するスイートを選択するまで、比較することはできません。デフォルトのスイートとチェックを使用することも、独自のカスタムチェックとスイートを作成することもできます。

スキャン検出結果

検出結果は、インスタンスでチェックのルールに違反したレコードへの参照です。ソースレコードと、レコードが特定のチェックのルールをトリガーした回数を確認できます。



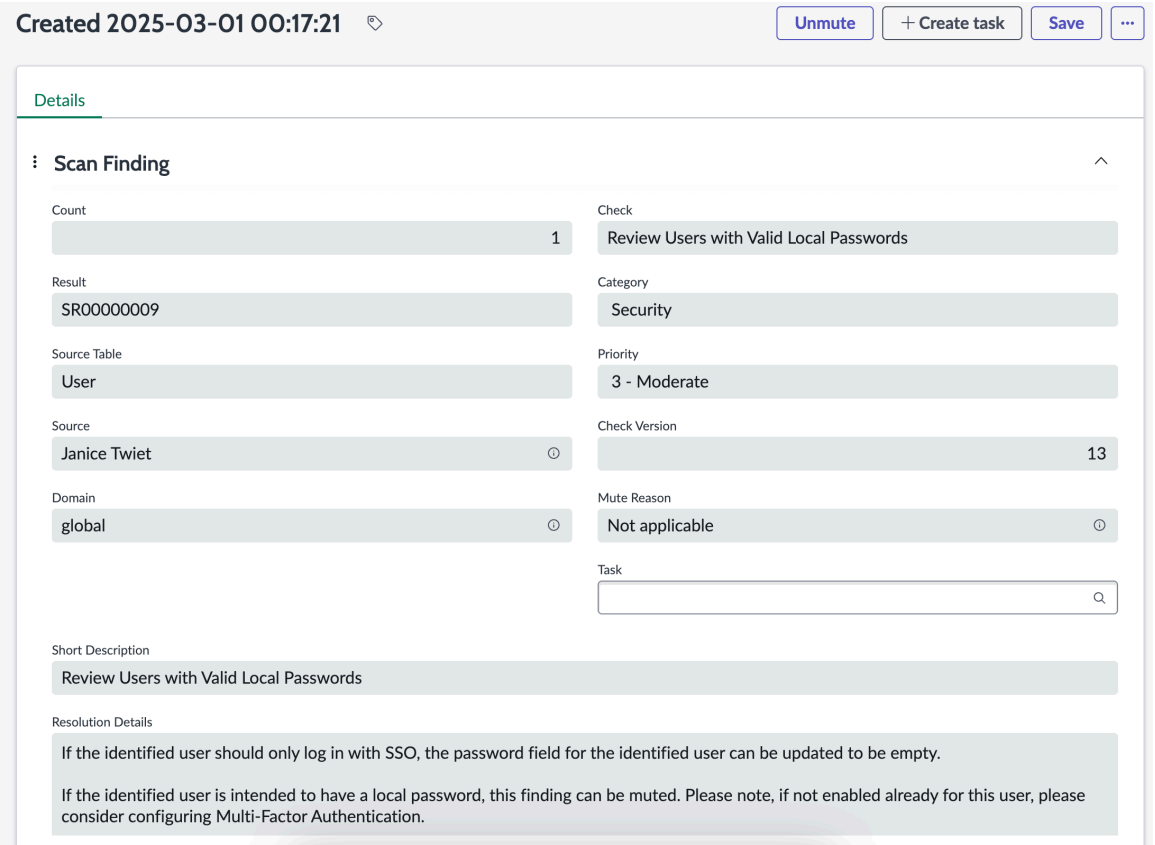
[検出結果] タブに移動して、スキャン検出結果をリストに表示します。リストの上にあるカードには、カードにリストされている特定の基準に一致する検出結果の数が表示されます。これらのカードのいずれかを選択してリストをフィルタリングし、基準に一致するカードのみを表示します。

自動翻訳

[+ タスクを作成] ボタンを選択して、検出結果を解決するセキュリティタスクを作成します。このボタンは、リストと結果レコード内の両方に表示されます。セキュリティタスクの詳細については、「 セキュリティタスク」を参照してください。

スキャン検知結果

[カウント] 列の下にあるリンクを選択すると、特定の結果に関連する詳細な結果レコードが表示されます。



チェック
: スキャンに関連付けられたチェックのリスト。

カテゴリ
: スキャンに関連付けられたセキュリティカテゴリ。たとえば、アクセス制御や悪意のあるコードなどです。

カウント
: レコードがチェックルールに違反した回数。

優先度
: セキュリティリスクの重大度:1 が最高の優先度、4 が最低です。

結果
: スキャンのステータスとタイプ。

チェックバージョン
: チェックのバージョン。

ソーステーブル
: チェックのルールに違反したレコード。

ミュートの理由

検出結果をミュートする理由。

ソース

検出結果が作成された日付。

タスク

スキャンに関連付けられたタスクレコード。レコードの検出結果からのタスクのアサインを容易にするために使用されます。

ドメイン

スキャンが適用されるドメイン。

簡単な説明

スキャンの簡単な説明。

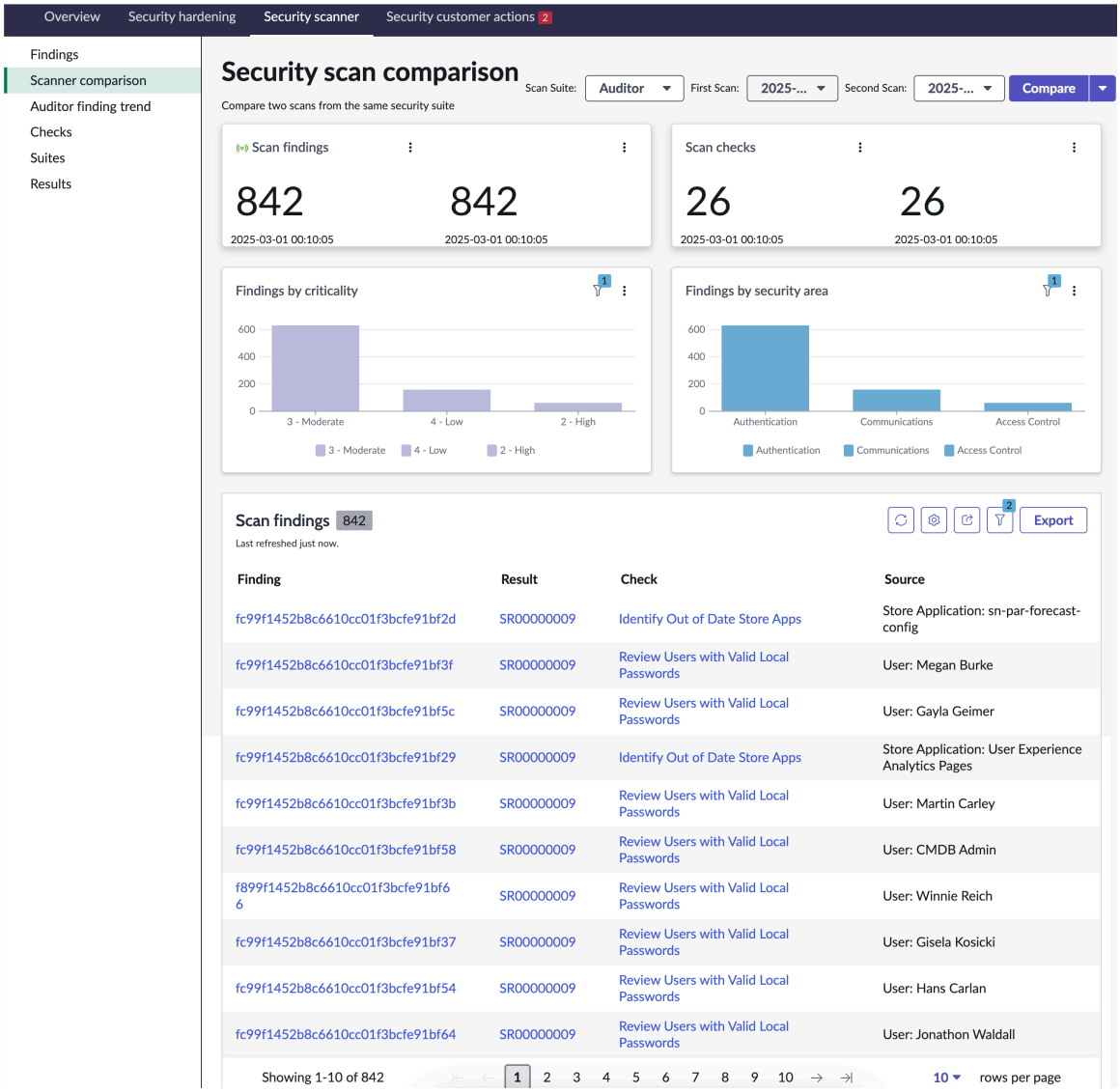
解決内容

スキャンによって報告された問題を解決する方法の説明。

検出結果は、[ミュート]/[ミュート解除]ボタンを選択することでミュートできます。スキャン結果をミュートすると、検出結果をミュートする理由を求められます。過去 6 か月間にミュートされた検出結果は、[スキャン検出結果] ページのミュート検出結果カードで確認できます。

セキュリティスキャンの比較

同じセキュリティスイートの 2 つのスキャンを比較して、ハードニング設定の健全性を可視化し、インスタンスのセキュリティ体制を改善します。



重要: 比較する同じスキャンのインスタンスが 2 つ以上ない場合、このページは空で表示されます。このページで比較を表示するには、スキャンを少なくとも 2 回実行してください。

[スキャンスイート] リストでスキャンを選択し、[最初のスキャン] と **[2** 番目のスキャン] を選択して比較を開始します。

セキュリティスキャンの比較ページには、1 回目と 2 回目のスキャンの間のハードニング設定のセキュリティ変更が表示されます。各カードの説明は次のとおりです。

スキャン検知結果

選択した各日付における、選択したスキャンの検出結果の数。

スキャンチェック

選択したスキャンで選択した日付ごとに実行されたスキャンチェックの数。

重要度別の変更された調査結果

重要度別に整理されたすべての結果を表示するグラフ。

セキュリティ領域別の変更された結果

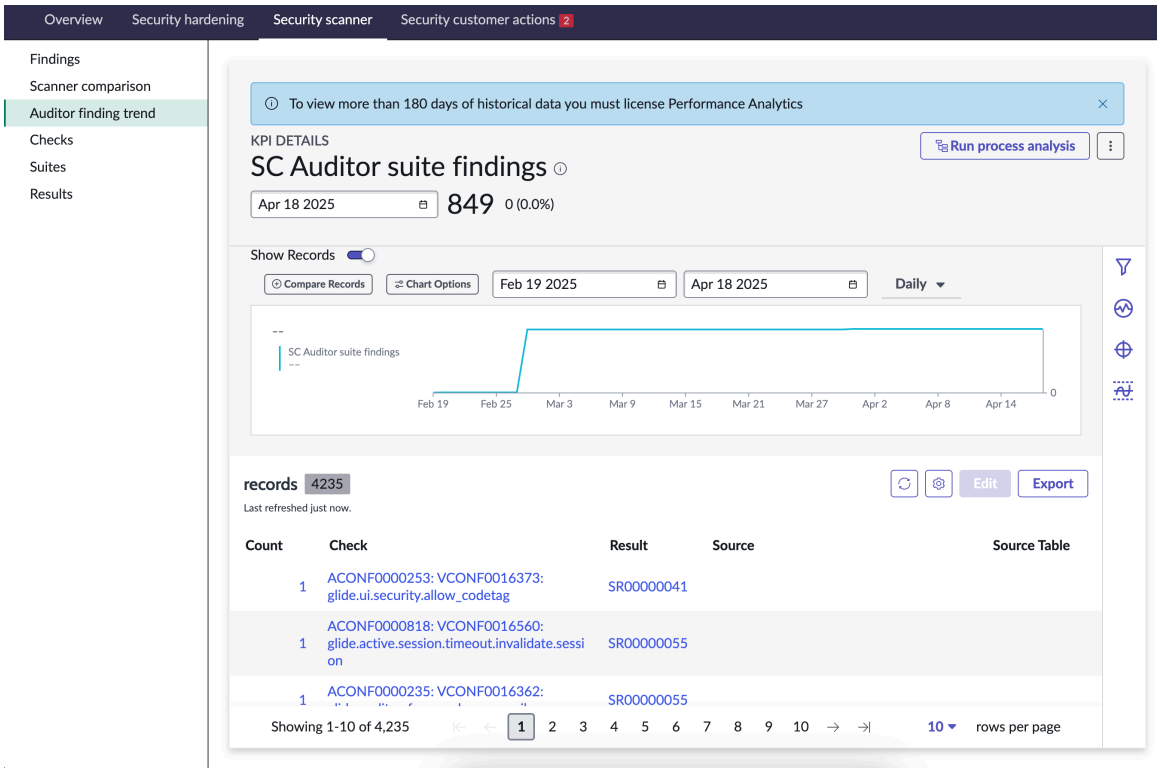
すべての検出結果が整理されたセキュリティエリアを表示するグラフ。

スキャン検知結果

すべてのスキャン結果のリスト。検出結果、スキャン結果を選択するか、チェックして関連するレコードを表示し、追加の詳細を確認します。右上の下部を使用して、リストをフィルタリング、更新、またはエクスポートします。

監査スイートの検知結果

セキュリティセンター監査スイートの結果を経時的にレビューします。



SC 監査スイートの結果チャート

このページには、選択した 2 つの日付間の SC 監査スイートの結果が比較されて表示されます。[チャートオプション] ボタンを使用して、分析する情報とその情報の表示方法を指定します。

分析

オプションのリストから選択します。選択したチャートタイプによっては、一部のオプションが使用できない場合があります。

ターゲット

組織が達成したい目標です。「インジケーター ターゲット」を参照してください。

しきい値

インジケーターの通常のスコア範囲を定義し、特定のイベントが発生したときに警告します。「インジケーター しきい値」を参照してください。

予測

過去の動作に基づいて将来のスコアを予測する機能について説明します。「パフォーマンスアナリティクススコア予測」を参照してください。

傾向図

1 つ以上のアイテムの値が時間の経過とともにどのように変化するかを示します。

コメント

自動翻訳

個々のデータポイントの注釈を表示します。

ラベル

可視化に関連するスコアを表示します。

統計情報

コンプライアンススコアに関連する統計情報を表示します。

時系列

グラフに表示するメトリクスを選択します。

スコア

重要業績評価指標 (KPI) のスコア。

変更

このインジケーターのスコアの変化。

変更割合

スコアに対する割合での変化。

グラフのタイプ

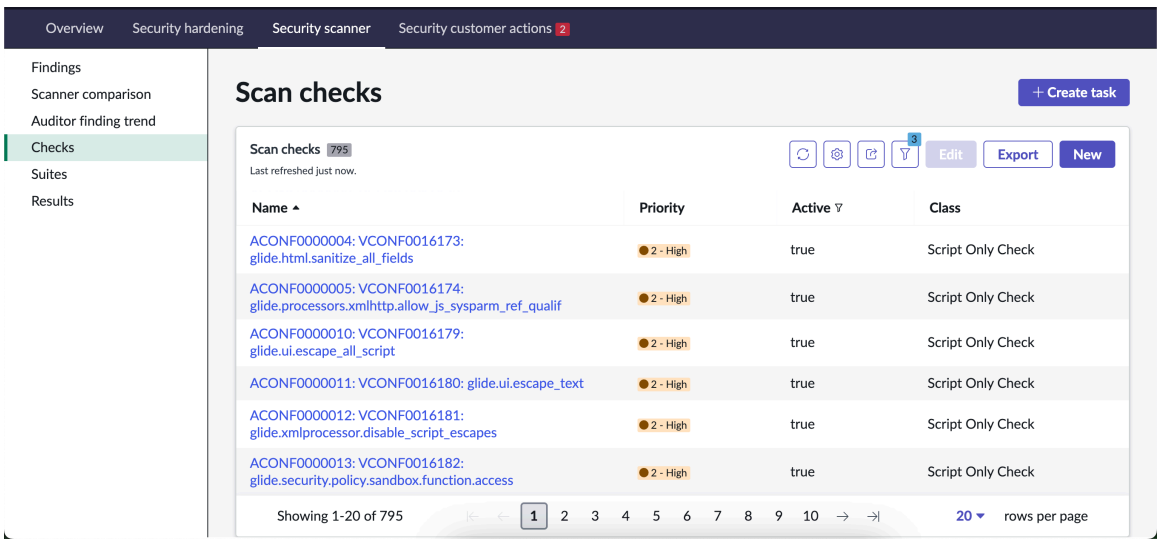
選択した情報の可視化方法を制御するチャートタイプを選択します。「さまざまな時系列可視化タイプのユースケース」を参照してください。

レコードリスト

スイートの一部として実行されるチェックの一覧を示します。

スキャンチェック

チェックを使用して、テーブル、レコード、またはメタデータに対して実行されているインスタンス内の例外を検出します。



チェックは、インスタンス内の異常を検出するために設計されたルールです。リストでチェックを選択すると、チェックで評価される内容や、チェックで検出結果が返された場合に問題を修正するための可能な手順などの詳細が表示されます。

[+ タスクを作成] ボタンを選択して、スキャンチェックに関連するセキュリティタスクを作成します。セキュリティタスクの詳細については、「セキュリティタスク」を参照してください。

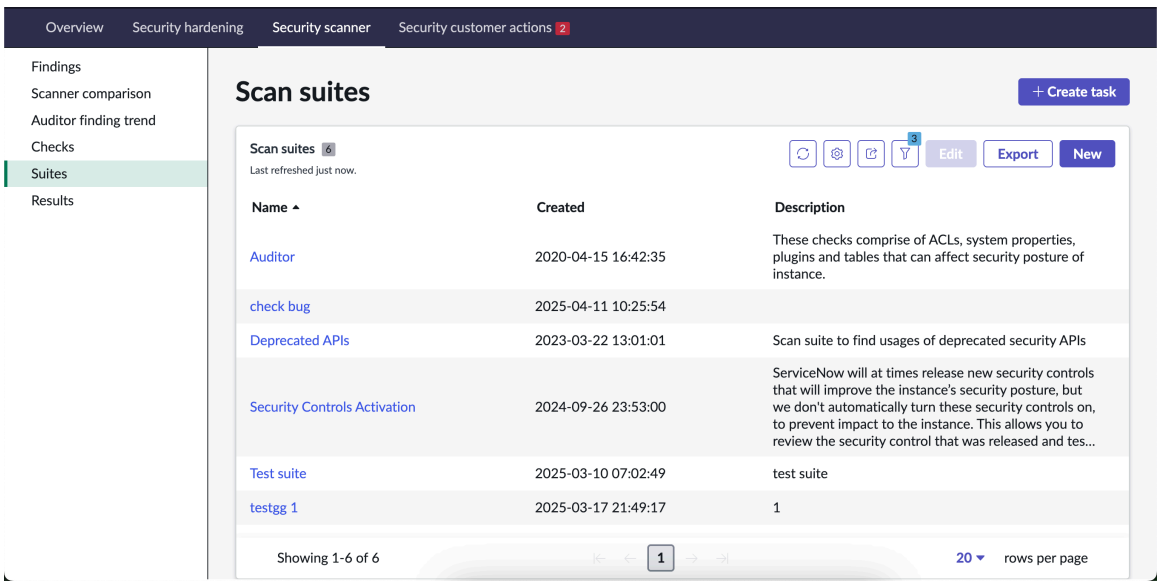
チェックは、問題を特定して組織のセキュリティ推奨事項を実装するために、テーブルチェック、列タイプチェック、スクリプトのみのチェック、リンターチェックの 4 つのクラスに分けられます。

クラスをチェック

クラスを確認	説明
テーブルチェック	テストする特定のテーブルとチェックがわかっている場合は、このチェッククラスを使用します。
列のタイプのチェック	このチェッククラスを使用して、ターゲット列フィールドタイプに一致するすべてのレコードを反復するために作成したルールを実装します。
チェックのみのスクリプト	このチェッククラスを使用して、メタデータや構成を検証し、独自のチェックを記述して複雑なチェックを実行します。
リンターチェック	このチェッククラスを使用して、スクリプトの問題を特定します。リンターチェックがレコードで実行されると、そのコードの抽象構文ツリーが生成され、それを使用して、コードの問題を分析できます。

スキャンスイート

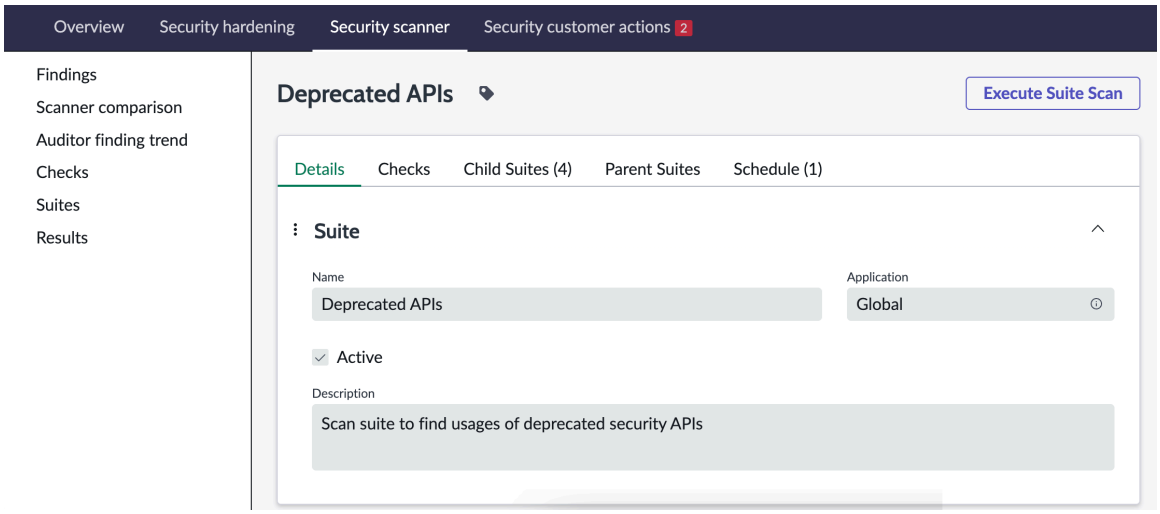
インスタンスで利用可能なスキャンスイートの詳細を確認します。



スキャンスイートは、一緒に実行されるセキュリティセンターのチェックのコレクションです。ベースシステムスイートを使用することも、既存のスイートをクローン作成し、クローンで行われたチェックを更新して独自のスイートを作成することもできます。詳細については、「スキャンスイートの作成」を参照してください。

[+ タスクを作成] ボタンを選択して、スキャンスイートに関連するセキュリティタスクを作成します。セキュリティタスクの詳細については、「 セキュリティタスク」を参照してください。

スキャンスイートの詳細



スイートの [名前] フィールドを選択して、スイートの詳細を表示します。このページでは、スキャンスイートの詳細がタブ付きインターフェイスに表示されます。タブ付きページでは、次の情報を確認できます。

詳細
: スイートの詳細には、名前、アプリケーション、および説明が含まれます。

チェック
: このスイートに含まれるチェックのリスト。個々のチェックの詳細は、 スキャンチェックに記載されているものと同じです。

子スイート
: このスイートに関連付けられている子スイートのリスト。このスイートを実行すると、すべての子スイートも実行されます。

親スイート
: このスイートが子スイートであるスイートのリスト。親スイートを実行すると、このスイートも実行されます。

スケジュール
: このスイートのスケジュールされた実行の詳細。

アクセス制御監査人のチェック

デフォルトのアクセス制御監査ツールスイートで利用可能なチェック、それらが評価する基準、およびそれらを使用してインスタンスのセキュリティを向上させる方法について説明します。

アクセス制御リストルール (ACL) は、まず要件のセットをユーザーに要求し、その後でユーザーとやり取りできるようにすることで、データへのアクセスを制限します。アクセス制御監査人のチェックは、次の表に示す 8 つの基準に従ってインスタンスを評価します。これらのチェックの結果を使用して、インスタンスのセキュリティを向上させます。

アクセス制御監査人のチェック

名前を確認	チェック基準	説明
タイプ - SCRIPT のすべてのプロセッサは CSRF トークンで保護する必要があります	CSRF トークンで保護されていない **SCRIPT** タイプのプロセッサーをチェックします。	**SCRIPT** タイプのすべてのプロセッサは、クロスサイトリクエストフォージェリ (CSRF) トークンで保護する必要があります。これらのプロセッサーで

アクセス制御監査人のチェック (続く)

名前を確認	チェック基準	説明
		は、CSRF オプションをオンにする必要があります。これにより、インスタンスが CSRF トークンを使用しない限りプロセッサーは実行できません。
ナレッジごとに定義する、寄稿可能/寄稿不可ユーザーのクライテリア	[寄稿可能] または [寄稿不可] ユーザー基準が定義されていないナレッジベースレコードをチェックします。	各ナレッジベースには 、[寄稿可能] または [寄稿不可] のユーザー基準を定義する必要があります。そうしないと、どのユーザーも寄稿基準が定義されていないナレッジベースにコンテンツを寄稿できてしまいます。
空の ACL	セキュリティ属性なし、ロールなし、または 公開 ロールを持たないアクセス制御リスト (ACL) レコードをチェックします。	ACL を空のままにするか 、public ロールを使用すると、この ACL で保護されているすべてのコンテンツにオープンアクセスが提供される可能性があります。
クライアントコール可能スクリプトインクルードのアクセス制御	ACL で保護されていないクライアント呼び出し可能スクリプトインクルードをチェックします。	すべてのクライアント呼び出し可能スクリプトインクルードは、必要なロールを使用して ACL で保護する必要があります。
UI ページのアクセス制御	ACL によって保護されていない UI ページをチェックする	UI ページへのアクセスを保護する ACL がないと、その UI ページにはログインしているすべての内部ユーザーがアクセスできます。制限がないと、ログインしているユーザーが不正な変更を行う可能性があります。
テーブルでのアクセス制御	ACL のないテーブルのチェック	テーブルは ACL で保護する必要があります。テーブルに格納されているデータへのアクセスは、それを必要とするユーザーのみに制限する必要があります。
ユーザーアカウントに内部ロールと外部ロールの両方を設定することはできません	内部ロールと外部ロールの両方がアサインされているユーザーレコードを確認します	内部ユーザーロールは、社内のユーザーを対象としています。外部ユーザーロールは、顧客やパートナーなどの外部担当者を対象としています。
誰でもアクセス可能なナレッジベースと記事	一般にアクセス可能なナレッジベースとナレッジベース記事をチェックします	公開されてアクセス可能なナレッジベースと記事は、インスタンス内のすべてのユーザーに表示されます。ナレッジベースと記事を、それらを必要とする特定の対象者に制限するこ

アクセス制御監査人のチェック (続く)

名前を確認	チェック基準	説明
		とで、セキュリティを強化します。

監査人チェック

監査ツールスイートを使用して SecureCheck を実行し、インスタンスのセキュリティ体制に影響を与える可能性のある構成ミスを検出します。

情報を確認

アクセス制御監査人のチェック

名前を確認	説明	スキャン結果タイプ
古いストアアプリを特定する	インスタンスでアクティブ化され、バージョンが更新されているアプリを識別します。 ストアアプリケーションの最新バージョンで実行していることを確認します。これには潜在的なセキュリティ問題の修正が含まれている可能性があります。	推奨される解決策
安全でない GlideRecord 呼び出し	エンドユーザーが直接呼び出すことができるスクリプトを識別します (クライアントコール可能なスクリプトインクルード、ウィジェット、プロセッサ、REST エンドポイントなど) これらのスクリプトは ACL を尊重し、GlideRecordSecure または GlideRecord を canRead、canWrite、canCreate、canDelete とともに使用する必要があります。	推奨される解決策
許可された JavaScript ライブラリを確認する	特定のサードパーティ JavaScript ライブラリを許可または拒否するために JavaScript コンテンツアクセス制御が使用されているスクリプトを識別します。 アクセスをブロックする前に、インスタンスのカスタマイズを確認して、ライブラリが使用されていないことを確認します。JavaScript コンテンツプロバイダーのアクセストラッキング [sys_js_content_provider_access_tracking] テーブルを確認して、ライブラリが最後にアクセスされた日付を確認できます。	推奨される解決策

アクセス制御監査人のチェック (続く)

名前を確認	説明	スキャン結果タイプ
	注: このチェックは、Tokyo 以降で最初にプロビジョニングされたインスタンスでは無視できます。関連付けられたテーブルのレコードには、デフォルトで拒否ルールが設定されています。Tokyo より前に最初にプロビジョニングされたインスタンスでは、JavaScript アクセス制御テーブルに許可ルールが存在する場合があります。	
対応する ACL がないクライアント呼び出し可能スクリプトインクルードを確認します	対応する ACL を持たないクライアント呼び出し可能スクリプトインクルードを識別します。これらのスクリプトは、デフォルトの ("*") クライアント呼び出し可能スクリプトインクルード ACL を使用します。 これらのスクリプトに対して、アクセスの適切な基準を定義する ACL を作成し、想定されるユーザーのみが提供された機能を操作できることを確認します。	推奨される解決策
レコードプロデューサーありでビジネスルールなしのカスタムテーブルをレビューする	サーバー側の追加検証を行わないレコードプロデューサーを識別します。このチェックでは、レコードプロデューサーはあるがビジネスルールが関連付けられていないカスタムテーブルを識別します。 これにより、ユーザーが予期しないデータを関連するテーブルに送信する可能性があります。	推奨される解決策
空の ACL をレビューする	スクリプト、条件、セキュリティ属性、ロールがない ACL レコード、または 公開 ロールを持つ ACL を識別します。 ACL を空のままにするか 、public ロールを使用すると、この ACL で保護されているすべてのコンテンツにオープンアクセスできます。	推奨される解決策
HTML サニタイズが無効になっているフィールドを確認する	HTML サニタイズが非アクティブになっている HTML フィールドを識別します。 HTML サニタイズは、HTML コード内の潜在的に有害な要素や属性を削除または置き換えます。サニタイズが非アクティブな HTML フィールドを確認して、この構成が必要かどうかを確認します。	推奨される解決策
非アクティブなセキュリティ機能プラグインをレビューする	追加の構成可能なセキュリティコントロールを提供する、アクティブ化されていないプラグインを特定します。このチェックによって生成さ	知らせる

アクセス制御監査人のチェック (続く)

名前を確認	説明	スキャン結果タイプ
	れた検出結果は、情報提供を目的として提供されています。 特定されたプラグインのいずれかを有効にする前に、プラグインがユースケースまたは要件を満たしていることを確認してください。識別されたユースケースがない場合は、これらの検出結果をミュートできます。	
許可された大きな IP アドレス範囲を確認する	多数の IP アドレスを含む IP アドレスアクセス制御範囲を識別します。 **注:** 誤検出が多数発生している場合は、特定のビジネスニーズに合わせて largestExpectedCIDRBlock 変数を調整することを検討してください。 クラスレスドメイン間ルーティング (CIDR) ブロックには、数が減少するにつれて大量の IP アドレスが含まれます。たとえば、CIDR ブロックサイズ 8 は CIDR ブロックサイズ 16 よりも大きい (IP アドレスが多い) 場合です。 現在の構成がビジネスニーズに一致していることを確認して確認します。	レビューと決定
パブリック GraphQL スキーマのレビュー	GraphQL API [sys_graphql_schema] テーブル内のパブリック GraphQL スキーマを識別します。 これらのスキーマは、認証なしで使用できるように構成できます。エンドポイントの機能によっては、認証されていないユーザーが予期しないアクションを実行したり、予期しないデータを操作したりする可能性があります。	レビューと決定
公開ナレッジベース記事のレビュー	非認証ユーザーがアクセスできるように設定されたナレッジベースとナレッジベース記事を識別します。 現在の構成がビジネスニーズに一致していることを確認して確認します。	レビューと決定
パブリック REST API エンドポイントのレビュー	認証なしで使用できるように設定されている、スクリプト化済み REST リソース [sys_ws_operation] テーブル内の REST API エンドポイントを識別します。	レビューと決定

アクセス制御監査人のチェック (続く)

名前を確認	説明	スキャン結果タイプ
	エンドポイントの機能によっては、認証されていないユーザーが予期しないアクションを実行したり、予期しないデータを操作したりする可能性があります。	
公開サービスポータルページのレビュー	公開されているサービスポータルページを識別します。サービスポータルページを非認証ユーザーが利用できるようにするには、[public] フィールドを [true] に設定します。 現在の構成がビジネスニーズに一致していることを確認して確認します。	レビューと決定
公開 UI ページのレビュー	公開される UI ページを識別します。UI ページは、[sys_public] ページを使用して非認証ユーザーが利用できるようにすることができます。 現在の構成がビジネスニーズに一致していることを確認して確認します。	レビューと決定
「アドミン」ロールを含むロールをレビューします	**admin** ロールを含むすべてのロール (ロール [sys_user_role] テーブル) を識別します。 **admin** ロールはユーザーに管理権限を付与します。必要な場合にのみ使用してください。現在の構成がビジネスニーズに一致していることを確認して確認します。これが意図的な構成である場合、このチェックはミュートできます。	レビューと決定
対応する ACL がない UI ページのレビュー	その UI ページの ACL を持たない UI ページを識別します。 特定の ACL を持たない UI ページは、デフォルトで汎用の UI ページ ACL に設定され 意図しないユーザーにアクセスが許可される可能性があります。	推奨される解決策
有効なローカルパスワードを持つユーザーをレビュー	ローカルに設定されたパスワードを持つユーザーを識別します。 ローカルパスワードを持つユーザーは、ローカルログインが許可されていない場合でも、ローカル認証情報を使用して API を介してインスタンスとやり取りできます。このパスワード設定は、統合ユーザーアカウントが正しく機能するために必要です。	レビューと決定

自動翻訳

アクセス制御監査人のチェック (続く)

名前を確認	説明	スキャン結果タイプ
	これらのユーザーアカウントを確認し、意図したユーザー (統合アカウントなど) のみがローカル認証で認証できることを確認します。	
古いハッシュアルゴリズムで保存されているパスワードをローテーションする	以前のバージョンの ServiceNow AI Platform で作成されたパスワードを持つユーザーアカウントを識別します。このパスワードは、現在レガシーまたは古いハッシュアルゴリズムと見なされているものを使用していた可能性があります。 パスワードをローテーションしていない古いプラットフォームバージョンで作成されたアカウントには、従来のハッシュアルゴリズムで保存されたパスワードが残っている可能性があります。作成された識別されたアカウントを確認し、パスワードのリセットを検討します。	推奨される解決策
レコードプロデューサーの保護	安全でないレコードプロデューサーを識別します。 適切なロールにアサインされていない場合、権限のないユーザーがそれらのロールにアクセスでき、機密情報が漏洩する可能性があります。レコードプロデューサーに適切なロールをアサインし、それらを必要とするユーザーのみがアクセスできるようにします。	推奨される解決策
UI アクションヴィジビリティ	テーブルへの読み取りアクセス権を持たないロールなしユーザーがアクセスできる UI アクションを特定します。 これらのユーザーは、これらの UI アクションを介して、アクセス権のないテーブルのデータを変更できる可能性があります。影響を受けるテーブルへのアクセス権を持つユーザーのみが UI アクションを使用できることを確認します。	推奨される解決策

スキャンスイートの作成

組織に対するインスタンスのセキュリティを分析できるように、カスタムスイートを作成してスケジュールします。

始める前に

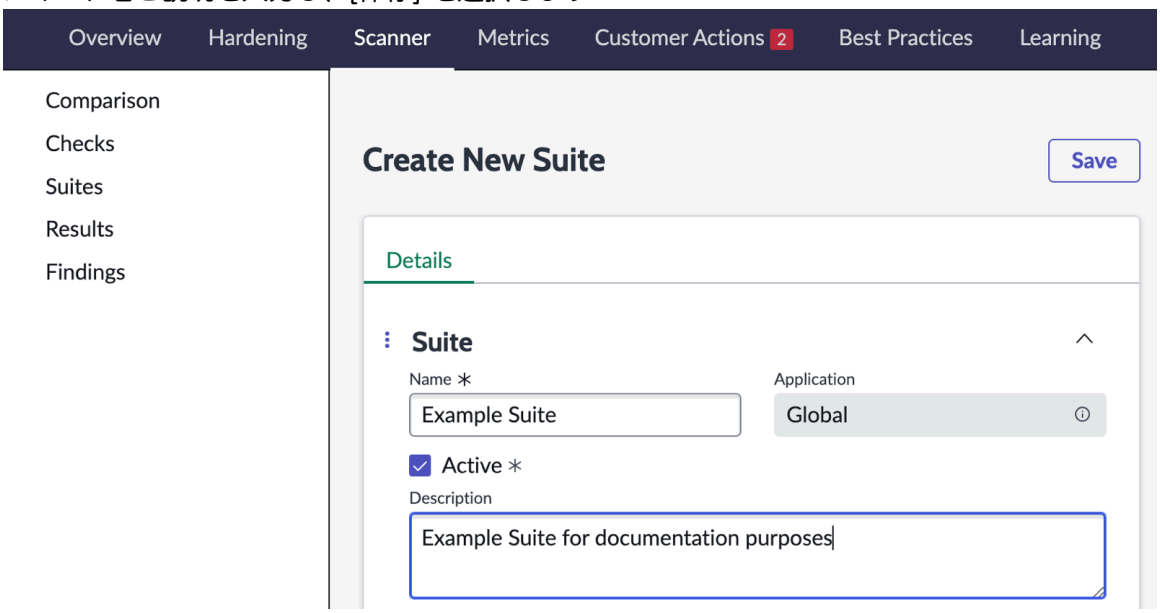
必要なロール：admin または sn_vsc.security_center_viewer

スイートは、スキャンに使用できるチェックのコレクションです。に移動すると、表形式で整理されたスキャンスイートのリストが表示されます。 スキャナー > スイート. 独自のスイートを作成するか、デフォルトのスイートである監査人を使用します。監査ツールは、セキュリティのベストプラクティスのチェックを含むデフォルトのベースシステムスイートです。これらのチェックは、インスタンスのセキュリティ体制に影響を与える可能性があるシステムプロパティ、プラグイン、およびテー

ブルで構成されています。次の手順は、スイートを作成する方法と、スイートを構成するために使用できるいくつかのオプションを示しています。

手順

1. Security Center で、[スキャナー] タブを選択し、画面の左側のパネルで [スイート] を選択します。
2. [スキャンスイート] ページで、[新規] ボタンを選択します。
3. スイート名と説明を入力し、[保存] を選択します



[保存] を選択すると、スイートの構成オプションがタブ付きインターフェイスに表示されます。

4. [チェック] タブを選択します。
   このタブを使用して、スイートにチェックを追加します。
   a. **[Edit** (編集)] を選択します。
   b. 追加するチェックを選択し、[追加] (**>**) を選択してスイートに配置します。
   c. **[Save** (保存)] を選択します。
5. [子スイート] タブを選択します。
   このタブを使用して、子スイートを追加します。子スイートとして追加されたスイートも実行されます。このスイートがスキャンで使用される場合。
   a. **[Edit** (編集)] を選択します。
   b. 追加する子スイートを選択し、[追加] (**>**) を選択して子スイートに配置します。
   c. **[Save** (保存)] を選択します。
6. [親スイート] タブを選択します。
   このタブを使用して、親スイートを追加します。スキャンで実行される親スイートも、このスイートを実行します。
   a. **[Edit** (編集)] を選択します。
   b. 追加する子スイートを選択し、[追加] (**>**) を選択して親スイートに配置します。
   c. **[Save** (保存)] を選択します。
7. [**Schedule** (スケジュール)] タブをクリックします。
   このタブを使用して、スイートの実行時間を設定します。

a. **[New** (新規)] を選択します。

b. スケジュールスキャンの詳細を入力します。
時間フィールドの形式は、時:分:秒です。

c. [保存] を選択します。

アクセス制御監査ツールスイートのクローンの作成

インスタンスのデフォルトのアクセス制御監査ツールスイートをクローンしてカスタマイズし、組織のセキュリティプラクティスに合わせて調整された新しいスイートを作成します。

始める前に
必要なロール：admin

このタスクについて
インスタンスで提供されるデフォルトのアクセス制御監査ツールスイートは変更できません。ただし、組織のセキュリティプラクティスに合わせてチェックを追加、削除、または編集したい場合は、アクセス制御監査ツールスイートを複製できます。アクセス制御監査ツールスイートをコピーすると、それをカスタマイズして、デフォルトのスイートに基づく新しいスイートを作成できます。アクセス制御監査ツールスイートには、インスタンスのセキュリティ体制に影響を与えるシステムプロパティ、プラグイン、およびテーブルを対象としたセキュリティのベストプラクティスに関連するチェックが含まれています。次の手順では、デフォルトのアクセス制御監査ツールスイートを複製して、組織の要件に合わせて調整する方法を示します。

手順

1. 移動先 すべて > インスタンススキャン > スイート.
2. [新規] を選択し、スイートの名前とオプションの説明を入力します。
3. フォームヘッダーを右クリックし、[保存] を選択します。
4. スキャンに必要なチェックを追加します。

   a. [チェック] タブで、[編集] を選択します。

   b. 条件を追加します。
   たとえば、スキャンチェックを追加するには、次のフィールド、演算子、値、および条件を適用します。

   [カテゴリ][次の値に等しい][セキュリティ]AND [アプリケーション][次の値に等しい][グローバル]

   たとえば、Category is Security And Application is Global となります。

5. [フィルターを実行] を選択します。
6. コレクションリストからチェックリストに追加するスキャンチェックを選択し、追加 (>) ボタンを選択します。
7. [保存] を選択します。
カスタムチェックが追加されたスイートが作成されました。
8. [スイートスキャンの実行] を選択します。

アクセス制御監査ツールスイートの表示

デフォルトのアクセス制御監査ツールスイートで使用可能なチェックを表示して、このスイートを実行したときに、どのチェックが実行されるのかを把握します。

始める前に
必要なロール：admin

このタスクについて
ServiceNow インスタンス内のデフォルトのアクセス制御監査ツールスイートにアクセスするために実行する必要があるステップです。

手順

1. セキュリティセンターで、[スキャナー] タブを選択し、左側のパネルから [スイート] を選択します。
2. [スキャンスイート] ページで、リストから [監査人] を選択します。
3. スイートチェックを確認するには、[チェック] タブを選択します。
   スイートで使用可能な 8 つのチェックのリストが表示されます。
4. 分析するチェックの名前を選択します。
5. チェックに関連付けられたフィールドを分析します。

名前	説明
名前	チェックの名前。
アプリケーション	チェックが属するアプリケーション (セキュリティセンター)。
カテゴリ	チェックに関連付けられているカテゴリ。
優先度	緊急性のレベル。
バージョン	チェックのバージョン番号。
アクティブ	チェックのステータス (アクティブ、非アクティブなど)。
簡単な説明	チェックの簡単なサマリー。
説明	チェックのより包括的なサマリー。
解決内容	緩和された潜在的なセキュリティインシデント。
ドキュメント URL	製品ドキュメントまたはナレッジベース (KB) 記事の関連ドキュメントへのリンク。
実行条件	チェックの開始をトリガーする条件。
テーブル	チェックが属するテーブル。
条件	チェックに適用される条件付きロジック。
詳細	詳細設定オプションです。

スキャンスイートの再スケジュール

ニーズに合わせてスキャンスイートのスケジュールを変更します。

始める前に
必要なロール:admin、sn_vsc.security_center_viewer

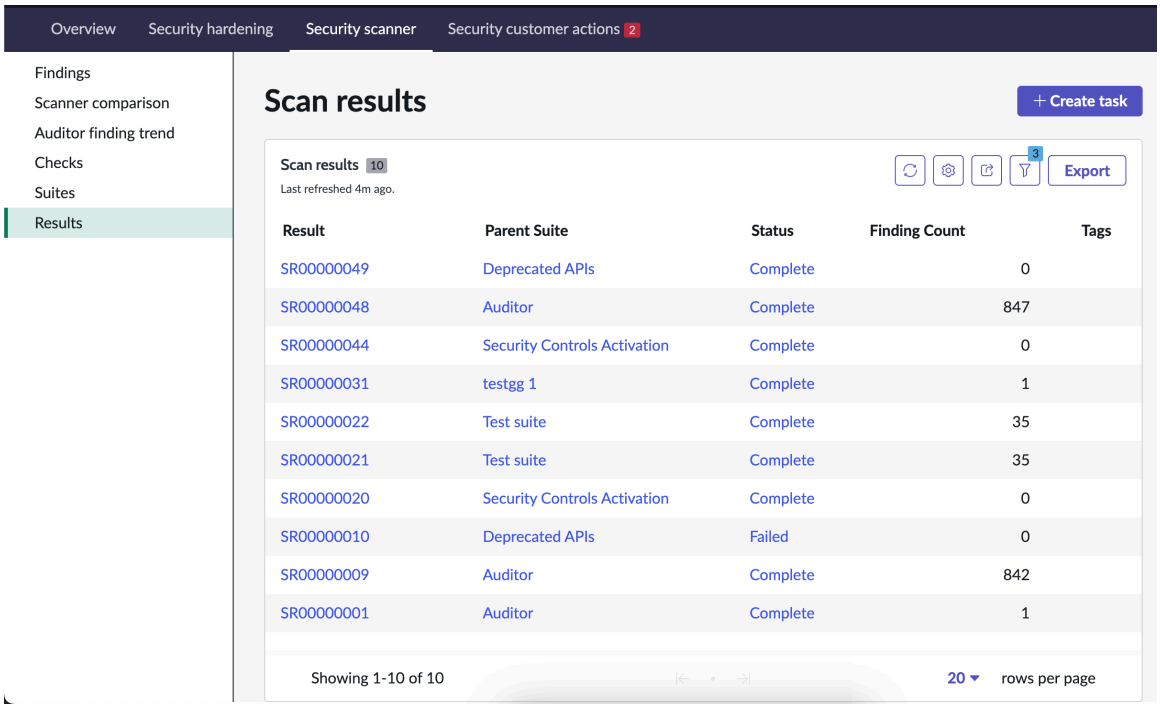
手順

1. 移動先 すべて > システムセキュリティ > セキュリティセンター.
2. [ツール] セクションで、[セキュリティ スキャナー] を選択します。
3. 画面の左端にあるメニューで [スイート] を選択します。
4. 再スケジュールするスイートを開きます。
5. [**Schedule** (スケジュール)] タブをクリックします。
6. 既存のスケジュールを選択するか、[新規] を選択して新しいスケジュールを作成します。
7. 必要に応じて、[スケジュール済みスキャン] フォームのフィールドに入力します。

フィールド	説明
名前	スキャンスケジュールの名前
実行	スイートの実行頻度を選択します。 注: スケジュールの定義に役立つ追加フィールドが表示されます。これらのフィールドは、選択内容によって異なります。
アクティブ	スケジュールがアクティブかどうか。非アクティブなスケジュールは実行されません。
条件付き	スケジュールのスクリプト化された条件を定義できる [条件] フィールドを有効にする場合に選択します。
条件	スケジュールのスクリプト化された条件。スイートは、スクリプトが `true` と評価された場合にのみ実行されます。
実行時のタイムゾーン	スケジュールの実行タイミングを決定するときに使用するタイムゾーン。

スキャン結果

スキャン結果に関連するデータを単一のビューから表示します。



スキャン結果は、スキャンのステータスとタイプをレポートします。スキャンの一部として実行されたすべてのチェックと、エラーやスキャンログなどのスキャンに関連するその他すべての情報を表示することもできます。

スキャン結果ダッシュボードを見つけるには、セキュリティセンターの [スキャナー] タブを選択し、左側のパネルで [結果] を選択します。スキャン結果は [スキャン結果] リストの下に一覧表示されます。結果を選択してドリルインし、タブ付きセクションに分けて詳細を表示します。

詳細
: ステータス、タイプ、実行時間 (秒単位) を含むスキャン結果の概要。

スキャン検知結果
: チェックの実行中に検出された検出結果。

スイート
: このスキャンの一部として実行されたスイートを表示します。

チェック
: このスキャンの一部として実行されたチェックを表示します。

失敗 (Failures)
: スキャン中に失敗したチェックとその失敗の理由がエラーメッセージの形式で表示されます。

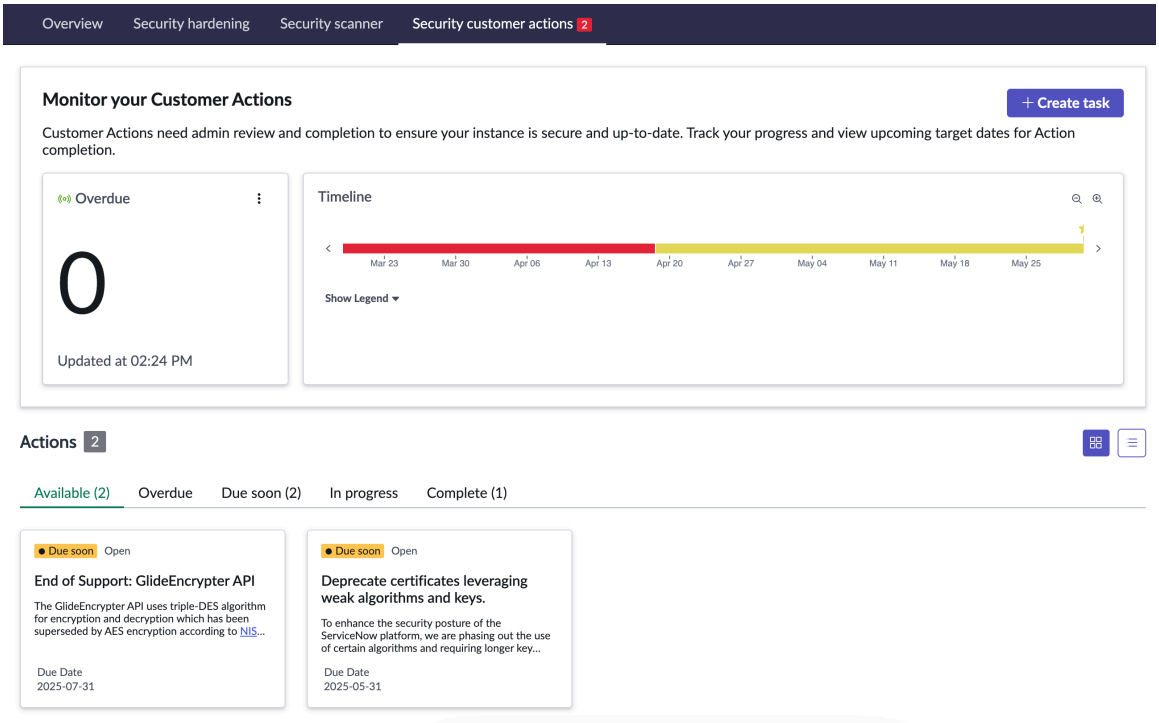
スキャンログ
: スキャン中に出力されたメッセージを表示します。

スキャン統計情報
: スキャンに関連する統計情報を表示します。

ターゲット
: チェックが実行されたすべてのターゲットを表示します。

顧客アクション

顧客アクションツールを使用して、インスタンスとプラグインの構成に基づいて重要なセキュリティ更新を実装します。



顧客アクションは、セキュリティ関連の変更の重要性の理解を助けることに重点を置いた、インスタンスでの推奨される変更です。手動で実装された顧客アクションは、インスタンスでレンダリングされる、ファミリーリリースやパッチを使用した ServiceNow コードや機能の自動更新とは異なります。顧客アクションを実装して、インスタンスのカスタマイズが妨げられるのを防ぎます。

顧客アクションのホームページでは、顧客アクションのタイムラインを表示できるため、手順を実装するタイミングに優先順位を付けることができます。上記の例のタイムラインには、期限切れの顧客アクションと、まもなく期限が来る顧客アクションが表示されます。さらに、[更新] ラベルで顧客アクションの進行状況を追跡できます。タブを選択すると、利用可能、期限切れ、期限が近い、進行中、または完了している顧客アクションを確認できます。

[+ タスクを作成] ボタンを使用して、顧客アクションを完了するためのセキュリティタスクを作成します。詳細については、「セキュリティタスク」を参照してください。

開始するには：

- このツールにアクセスするには、セキュリティセンター内の [顧客アクション] に移動します。
- 顧客アクションは、Washington DC 以降のセキュリティセンターアプリに含まれています。

顧客アクションを実装

インスタンスに顧客アクションを実装してセキュリティ体制を強化する方法について説明します。

始める前に

必要なロール：admin

このタスクについて

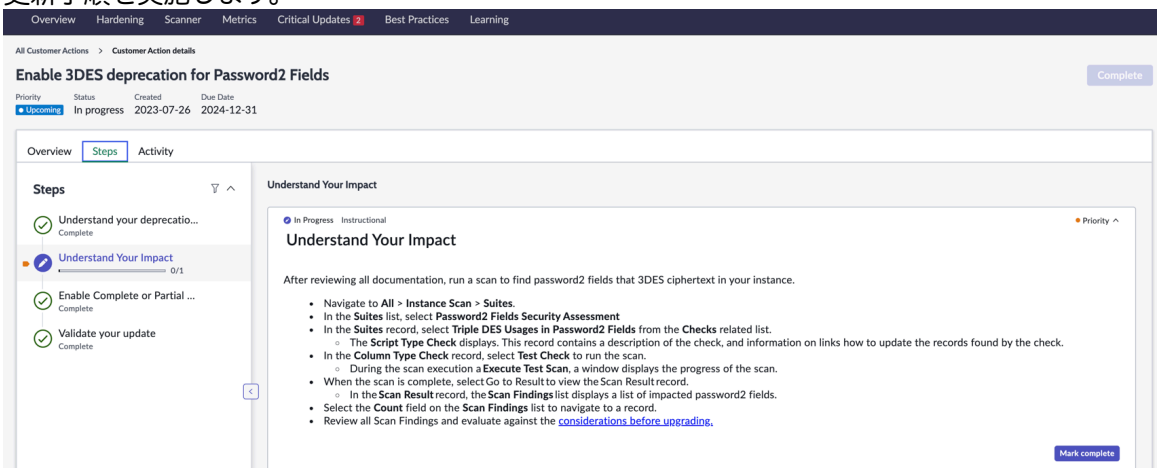
顧客アクションでは、インスタンスに変更を実装してセキュリティ体制を強化する方法を順を追って説明します。顧客アクションは、製品ドキュメントから関連コンテンツを取得して、実装プロセスの各手順を示します。顧客アクションを実装する方法については、以下の手順に従ってください。

手順

1. 顧客アクションアプリに移動します。
2. 実装する顧客アクションを選択します。

[更新] セクションの [利用可能] タブには、インスタンスのすべての顧客アクションが一覧表示されます。このタブの [顧客アクション] をクリックすると、指示が表示されます。

3. 概要を読んで、カスタマーアクションが必要な理由を確認し、手順を実装する準備をします。[次のステップに移動] をクリックして、実装の詳細に進みます。
4. 更新手順を実施します。



   a. 各グループの指示を読み、インスタンス内で実施します。
   b. 一連の指示を完了したら、[完了としてマーク] をクリックします。
   c. すべての更新手順を完了するまで、手順 a、b を繰り返します。
   d. [完了] をクリックします。
   顧客アクションが [**Complete**] タブに表示されます。

顧客アクションの表示

顧客アクションに関連するすべてのアクティビティの詳細を表示します。

始める前に

必要なロール：admin

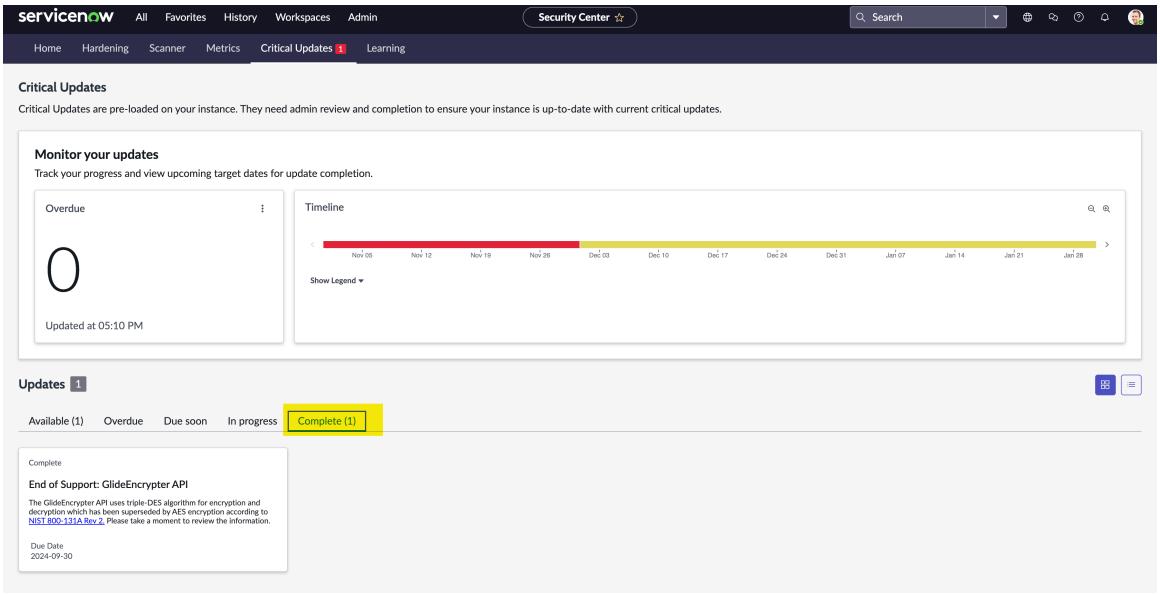
このタスクについて

アクティビティは新しいものから古いものの順にリストされるため、最新のアクティビティを最初に分析できます。アクティビティに対する各更新にはタイムスタンプが付き、検索とフィルターを使用して特定の情報をクエリできます。

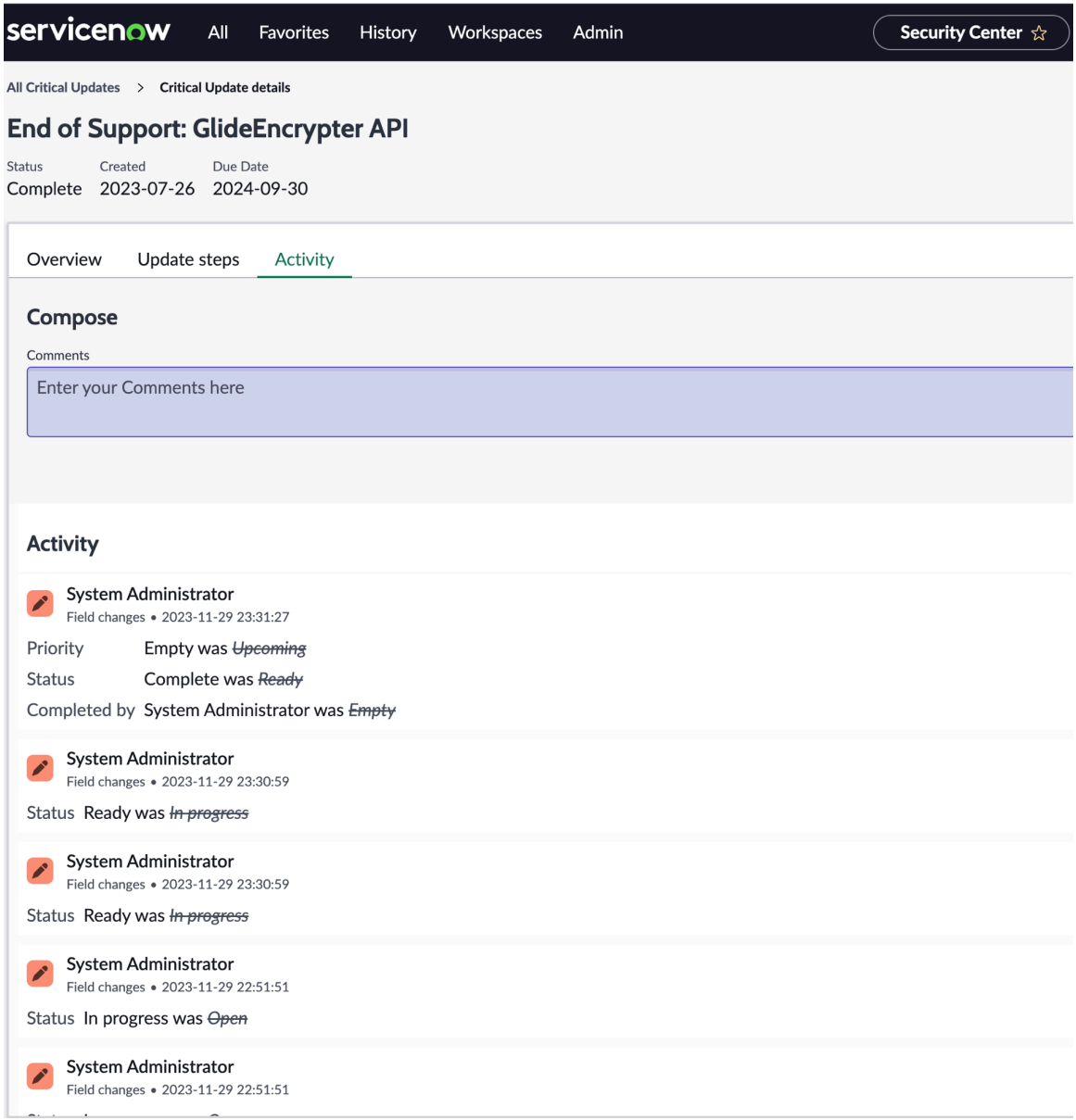
手順

1. セキュリティセンター内で [顧客アクション] アプリに移動します。
2. 顧客アクションの [更新] ラベルに移動します。
3. ステータスをクリックし、そのステータスから顧客アクションを選択します。

   以下の例では、[完了] ステータスが選択されてから、[サポート終了:GlideEncrypter **API**] の [カスタマーアクション] が選択されています。



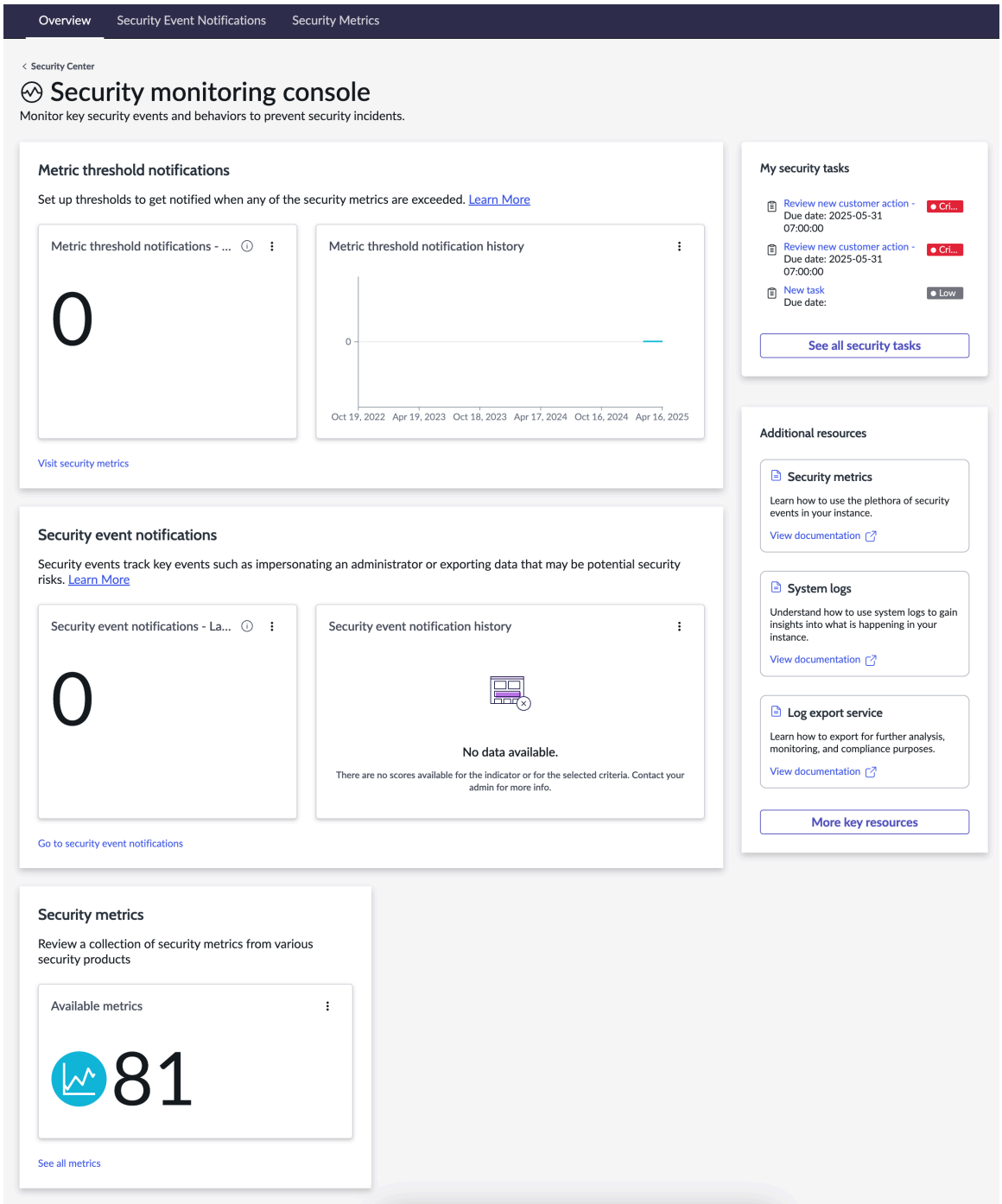
4. カスタマーアクションのアクティビティを表示します。



顧客アクションに関連するすべてのアクティビティは自動的に記録されます。さらに、アクティビティに追加のコメントを追加することもできます。

セキュリティモニタリングコンソール

セキュリティ通知とメトリクスを監視して、インスタンスの潜在的なセキュリティリスクについて常に最新情報を入手します。



セキュリティモニタリングコンソールは複数のセクションに編成されており、各セクションにはセキュリティモニタリングの特定の側面に関する情報を含むカードがあります。カードで情報 (ⓘ) アイコンを選択すると、それぞれの意味の詳細が表示されます。

ナビゲーションバー

ページの上部にあるバーを使用して、[概要]、[セキュリティイベント通知]、および [セキュリティメトリクス] セクション間を移動します。

メトリクスしきい値の通知

[メトリクスしきい値通知] パネルで、インスタンスのメトリクスしきい値通知について説明します。しきい値は、インジケーターの通常のスコアの範囲を定義し、スコアが過去最高値に達したときな

ど、特定のイベントが発生したときに警告します。このパネルには、過去 7 日間のしきい値通知の数と、経時的なしきい値数を示すグラフが表示されます。

いずれかのカードを選択して [セキュリティメトリクス] タブを開き、これらの通知の詳細を表示します。

セキュリティイベント通知

[セキュリティイベント通知] セクションで、代理操作やデータエクスポートなどのセキュリティイベントを追跡します。このパネルには、過去 7 日間のセキュリティイベントの数と、その数の経時的なグラフが表示されます。

いずれかのカードを選択して [セキュリティメトリクス] タブを開き、これらの通知の詳細を表示します。

セキュリティ測定基準

このセクションには、インスタンスで利用可能なセキュリティメトリクスの数が表示されます。さまざまなセキュリティ製品から得られる一連のセキュリティメトリクスをレビューします。

自分のセキュリティタスク

自分にアサインされた最も緊急性の高いセキュリティタスクを表示します。タスクを選択して詳細を表示するか、[すべてのセキュリティタスクを表示] を選択してセキュリティタスクの完全なリストを表示します。セキュリティタスクの詳細については、「 セキュリティタスク」を参照してください。

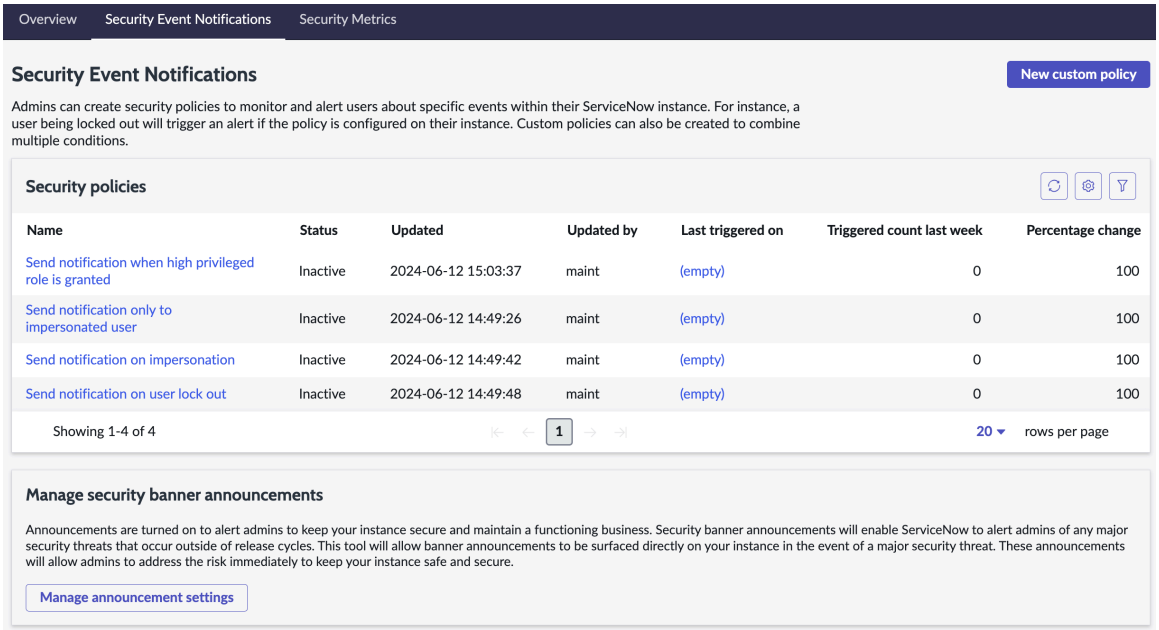
その他のリソース

画面の端にある [追加リソース] セクションを使用して、Security Center の学習に関連するドキュメントとビデオ、およびインスタンスのセキュリティの管理に関する情報に移動します。

セキュリティイベント通知

ServiceNow インスタンスのデフォルトのセキュリティイベント通知ポリシーを表示、管理、分析したり、機能にアクセスしてカスタムポリシーを作成したりできます。

セキュリティセンターのホームページからセキュリティイベント通知にアクセスするには、[セキュリティコンソール] セクションで [セキュリティモニタリングコンソール] を開き、[セキュリティイベント通知] タブを選択します。



安全でない可能性がある、またはアドミンの監視が必要なアクションをユーザーが ServiceNow AI Platform で実行したときにメール通知をトリガーする通知ポリシーをカスタマイズして作成します。

アドミニストレーターは、デフォルトポリシーの値をカスタマイズしたり、デフォルトポリシーのクローンを作成して編集したり、独自のカスタムポリシーを作成したりできます。

セキュリティポリシー

セキュリティイベント通知コンソールには、インスタンスのセキュリティポリシーに関する重要な情報が表示されます。このテーブルには、以下を表示する列が含まれています。

ラベル	説明
名前	セキュリティイベント通知ポリシーの名前。
ステータス	ポリシーのステータスは、アクティブまたは非アクティブです。
更新日時	ポリシーが最後に更新された日時。
更新者	ポリシーを更新したユーザー。
前回のトリガー日	ポリシーが最後に実行された日時。
先週のトリガー回数 (Triggered count last week)	ポリシーが過去 7 日間に生成したトリガーの数。
変更の割合	週次トリガーの変化率 (パーセンテージで計算)。 **注:** たとえば、このポリシーが週に平均 5 件の通知をトリガーし、先週 10 回トリガーした場合、変化率の増加は 100% になります。

テーブルでポリシーを選択すると、概要、設定、通知履歴などの詳細が表示されます。[新しいカスタムポリシー] ボタンを選択して、組織のユースケースに合わせたカスタムポリシーを作成します。

セキュリティバナーのお知らせを管理

[お知らせ設定を管理] を選択して、アドミニストレーターがこのインスタンスでバナー通知を受信するかどうかを制御します。これらのお知らせの詳細については、「 セキュリティバナーのお知らせ」を参照してください。

カスタムセキュリティイベント通知ポリシーの作成

組織のニーズに固有のカスタムセキュリティイベント通知を作成する方法について説明します。これにより、インスタンスでユーザーとグループによって実行されるアクションを監視し、潜在的なセキュリティリスクに関する通知を生成できます。

始める前に
必要なロール：admin

手順

1. セキュリティセンターアプリに移動し、 通知.
2. [新規カスタムポリシー **(New custom policy)]** を選択します。
3. ポリシーの開始をトリガーする 3 種のイベントのいずれかを選択し、[作成] を選択します。
4. ポリシーを構成します。
   新しいカスタムセキュリティイベント通知ポリシーを構成する方法については、「セキュリティイベント通知ポリシーの設定の構成」を参照してください。

セキュリティイベント通知ポリシーの変更

セキュリティイベント通知ポリシーの設定を変更する方法について説明します。

始める前に
必要なロール：admin

手順

1. セキュリティイベント [通知] ホームページに移動し、変更するポリシーのいずれかを選択します。

   注: デフォルトポリシーのトリガーまたは条件ロジックを変更することはできません。これらの設定を変更するには、まずポリシーを複製する必要があります。これを行うには、[更新] の横にある下矢印を選択し、次に [複製] を選択します。

2. [ポリシー設定 **(policy settings)]** を選択して、設定を変更します。
3. [更新] を選択して、変更を保存します。
4. オプション: 通知のトリガーを開始する準備ができるまで、[非アクティブ化 **(deactivate)]** を選択します。
5. オプション: 通知のトリガーを開始するには、[アクティブ化 **(Activate)]** を選択します。
   新しいカスタムセキュリティイベント通知ポリシーを構成する方法については、「セキュリティイベント通知ポリシーの設定の構成」を参照してください。

セキュリティイベント通知ポリシーの設定の構成

組織固有のニーズに合わせて、セキュリティセンターでセキュリティイベント通知ポリシーをカスタマイズする方法について説明します。

始める前に
必要なロール：admin

このタスクについて

[ポリシー設定 **(Policy settings)]** ページでは、セキュリティ イベント通知ポリシーの設定をカスタマイズできます。ここでは、いつポリシーを実行するか、条件付きロジック、および送信する通知に関連するオプションを調整できます。

手順

1. セキュリティセンターで、[通知] を選択します。
2. [セキュリティポリシー] テーブルで、構成するセキュリティイベント通知ポリシーを選択します。たとえば、[高特権ロールが付与されたときに通知を送信する **(Send notification when high privileged role is granted)]** です。
3. [ポリシー設定 **(Policy settings)]** を選択します。
4. ポリシー設定を構成します。

ポリシー設定の構成オプション

ラベル	構成の内容
実行タイミング	トリガー：ポリシーを開始するイベント。
条件	ポリシーに適用する条件付きロジックと条件。 ○ 条件のロジック **(Condition logic)**: 2 つのオプション ▪ このポリシーをトリガーするには、すべての条件を満たす必要がある **(All conditions must be met to trigger this policy)**：通知を送信するには、ポリシーのすべての条件が有効である必要があります。 ▪ このポリシーをトリガーするには、いずれかの条件を満たす必要がある **(Any of the conditions must be met to trigger this policy)**：条件のいずれかが有効な場合に通知が送信されます。 ○ 条件を追加：[条件を追加] を選択し、適切な値を指定します。 [条件を削除] を選択すると、条件を削除できます。
通知	メール本文とメール受信者： ○ 通知：事前定義された通知テンプレートのいずれかを選択します。詳細については、「セキュリティイベント通知のカスタムメールの作成」を参照してください。 ○ 通知を追加：[通知を追加] を選択します。 ○ グループ：通知を受信するグループを選択します。 ○ ユーザー：通知を受信するユーザーを選択します。

ラベル	構成の内容
	[通知を削除] を選択すると、通知を削除できます。

注: ポリシーを非アクティブにするには、[非アクティブ化] を選択します。ポリシーを複製するには、[更新] (

) ボタンの横にある下矢印を選択します。

5. [更新] を選択して設定を保存します。

セキュリティイベント通知のカスタムメールの作成

新しい通知の構成、トリガーの設定、受信者の定義、動的イベントフィールドを使用したメールコンテンツの作成などによって、セキュリティイベント通知用のカスタムメールを作成する方法について説明します。

始める前に
必要なロール：admin

手順

1. 選択 システム通知 > 通知.

2. [新規] を選択します。

3. 通知 フィールドに一意の名前を入力します。

4. テーブルの横にあるフィールドを選択し、セキュリティポリシー通知 (*sn_vsc_security_policy*) を入力します。

5. 新しいレコードを構成します。
レコードには、構成が必要なタブが、[送信時]、[受信者]、[内容] と 3 つあります。

タブ	説明
送信時	[送信条件] ラベルの横のフィールドを選択し、[トリガー] を選択します。
受信者	○ [フィールド内のユーザーまたはグループ] の横のローカルアイコンをクリックし、[ユーザー] と [グループ] の両方を選択します。[アイテムを追加] (>) アイコンを選択して、フィールドを [選択済み] テキストフィールドに移動します。 ○ レコードヘッダーバナーを右クリックして、[保存] を選択します。
内容	メール テンプレートを作成します。 詳細については、「メール テンプレートを作成する」を参照してください。 イベントのフィールドを [メッセージ **HTML]** に入力します。これを行うに

タブ	説明
	は、${event_id.FIELD_NAME} を使用します。以下に例を示します。 Role: ${event_id.role.URI_REF} Granted to: ${event_id.granted_to.URI_REF} Granted by: ${event_id.user.URI_REF} Logged at: ${event_id.sys_created_on} Security Center Notification: ${execution.policy.name}

6. [送信] を選択します。
7. セキュリティイベント通知のメールテンプレートにカスタム通知を追加します。
   a. セキュリティセンターで、次を選択します: 通知 > ポリシー設定.
   b. [通知] ラベルに移動します。
   c. 作成したカスタム通知の名前を [通知] フィールドに入力します。

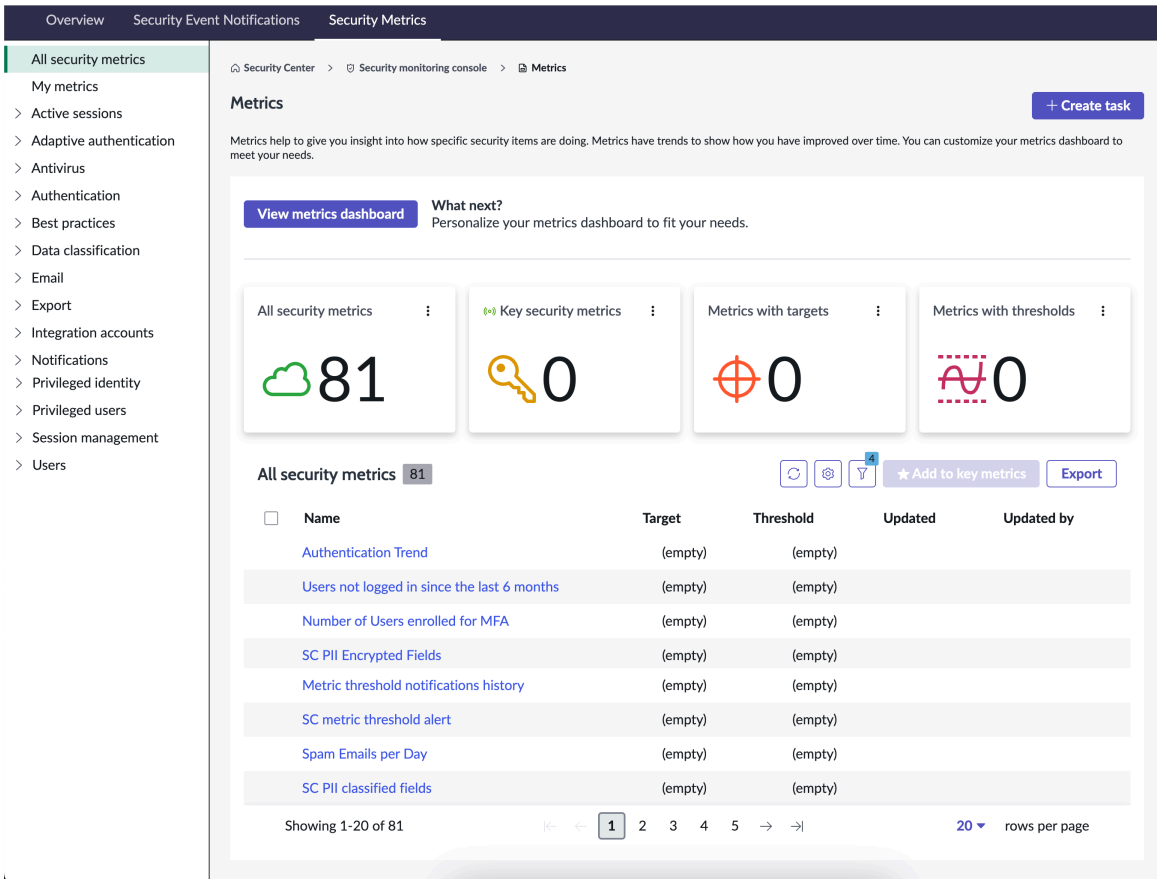
セキュリティイベント通知履歴

インスタンスのセキュリティイベント通知の完全な履歴について説明します。

[通知] ページには、インスタンスのすべてのセキュリティイベント通知ポリシーの通知履歴の概要が表示されます。データはチャートで表示したり、グラフィカルに表示したり、ターゲット、しきい値、KPI シグナルなどのパフォーマンスアナリティクス機能を適用したりできます。または、データをテーブル形式で表示して、検索、ソート、フィルター、クエリなどの標準機能を利用することもできます。

セキュリティ測定基準

50 を超えるさまざまなセキュリティメトリクスを監視して、潜在的なセキュリティの脅威や安全でない動作を特定します。メール通知のしきい値を設定し、さまざまな方法でデータを可視化および分析します。データをエクスポートするか、組織にとって最も重要なメトリクスを使用してダッシュボードを作成します。



Security Center 内の Security Metrics にアクセスするには、ツールセクションで **[Security Monitoring**] を選択します。次に、[セキュリティメトリクス] カードを選択します。

[すべてのセキュリティメトリクス] ダッシュボード

このダッシュボードを使用して、ログイン失敗、エクスポート、代理操作などのメトリクスを表示します。リストの上にあるカードには、カードにリストされている特定の基準に一致するメトリクスの数が表示されます。これらのカードのいずれかを選択してリストをフィルタリングし、基準に一致するメトリクスのみを表示します。

[+ タスクを作成] ボタンを選択して、メトリクスに関連するセキュリティタスクを作成します。セキュリティタスクの詳細については、「 セキュリティタスク」を参照してください。

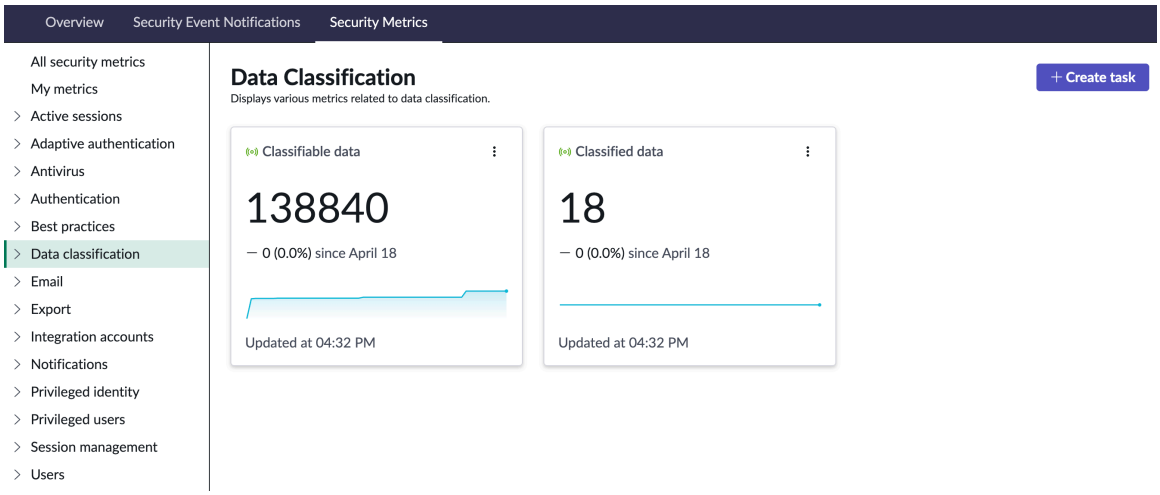
[自分のメトリクス] ダッシュボード

これは、最も関心のあるメトリクスに関する情報を表示するカスタマイズ可能なダッシュボードです。[編集] を選択して、可視化、フィルター、見出し、画像、リッチテキスト、およびリストを追加してダッシュボードをカスタマイズします。ダッシュボードをカスタマイズするさまざまな方法については、「のダッシュボード プラットフォームアナリティクス 」を参照してください。

メトリクスナビゲーションペイン

画面の端にあるナビゲーションペインには、インスタンスメトリクスが表示されます。すべてのメトリクスを単一のリストに表示するには、[すべてのセキュリティメトリクス] を選択します。[すべてのセキュリティメトリクス] エントリの下にあるいずれかのアイテムを選択して、カテゴリ別に整理されたメトリクスを参照します。

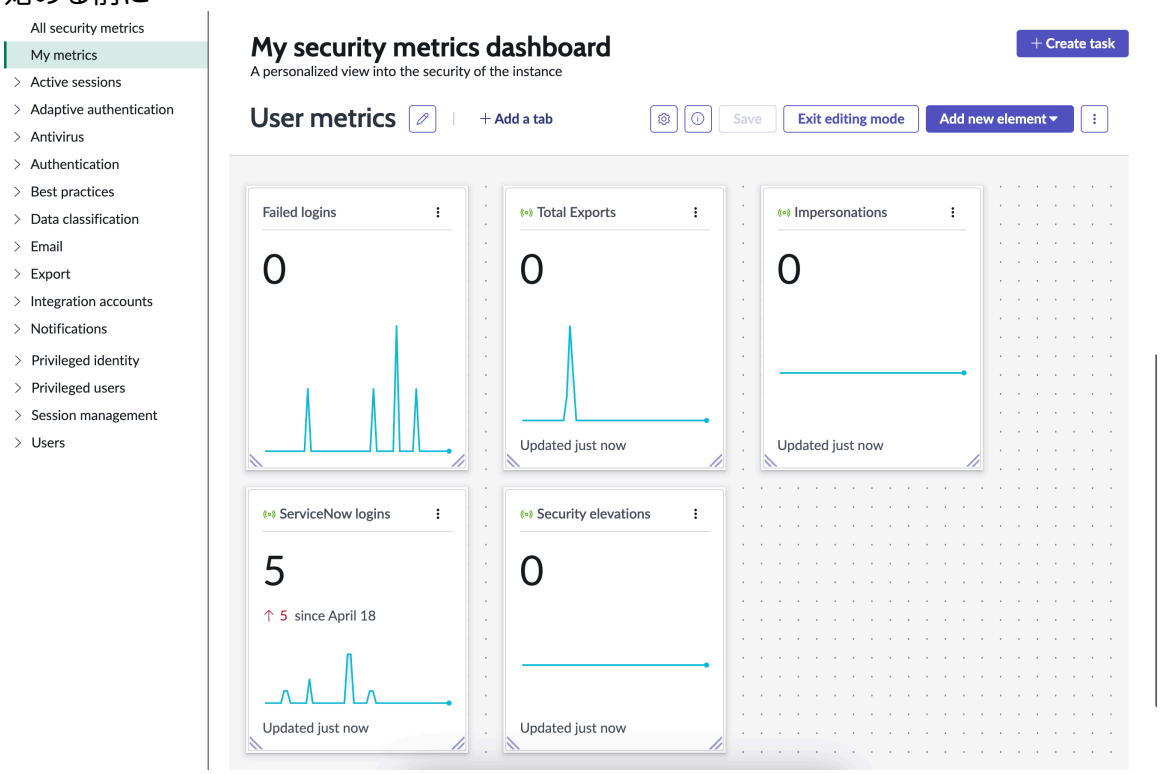
各カテゴリには、セキュリティの [自分のメトリクス] ダッシュボードと同様のダッシュボードと、そのカテゴリのメトリクスのカードが表示されます。



[自分のセキュリティメトリクス] ダッシュボードのカスタマイズ

グラフやチャートなどのさまざまなソースのメトリクスでカスタマイズできる [自分のセキュリティメトリクス] ダッシュボードの柔軟性について説明します。組織の特定の要件に合わせてダッシュボードをカスタマイズします。

始める前に



セキュリティ測定基準ダッシュボードをカスタマイズするには、セキュリティセンターのアプリケーションスコープ内にいることを確認してください。このスコープに切り替えるには、次を選択しますアプリケーションピッカー > アプリケーションスコープをクリックし、「セキュリティセンター」と入力します。

必要なロール：admin

このタスクについて

[自分のセキュリティ測定基準] ダッシュボードを実装およびカスタマイズします。レポートから生成されたグラフやチャートなど、さまざまなソースの測定基準およびコンテンツをダッシュボードに追

加します。組織のニーズに合わせてタブをカスタマイズできるように、ダッシュボードにタブを追加するために必要な手順について説明します。

手順

1. Security Center 内の Security Metrics にアクセスするには、ツールセクションで **[Security Monitoring**] を選択し、 **[Security metrics**] カードを選択します。
2. [編集] を選択して編集モードに入ります。
3. [+タブを追加] を選択してタブを作成します。

   注: 既存のタブの横にあるプラス **(+)** アイコンを選択して、新しいタブを追加することもできます。

4. 鉛筆アイコンを選択して名前を入力し、新しく作成したタブのタイトルを編集します。
5. [新しい要素の追加] を選択して、タブに要素を追加します。
6. **[Save** (保存)] を選択します。
7. オプション: 組織内のユーザー、グループ、およびロールとダッシュボードを共有するには、次を選択します その他のアクションメニュー > 共有.
   a. アクセス権を付与するユーザーの名前を入力します。
   b. [**Confirm** (確認)] を選択します。

しきい値がトリガーされたときにメールを送信するようにセキュリティメトリクスを設定する

しきい値がトリガーされたときにインスタンスがメール通知を生成するように、セキュリティメトリクスを構成する方法について説明します。

始める前に
必要なロール：admin

手順

1. セキュリティセンターで、[メトリクス] に移動します。
2. 組織のしきい値を設定する測定基準を選択します。

   Example
   失敗したログインを管理するために、しきい値を設定できます。失敗したログインの数がこのしきい値に達すると、組織のアドミニストレーターまたはセキュリティエキスパートに通知されます。この場合にターゲットとする適切なセキュリティメトリクスは、 ユーザー > ログイン失敗.
3. しきい値アイコンを選択します。
4. しきい値パネルの プラス **(+)** アイコンを選択して、フォームを構成します。
   使用可能なオプションは、[ヴィジビリティ]、[しきい値タイプ]、[しきい値] の設定です。
   a. 可視性については、全員に通知を受信させたい場合は [全員用] を、自分だけ受信したい場合は [自分] を選択します。
   b. [しきい値タイプ] で、常に高い値を選択します。
      このフィールドはしきい値通知をトリガーします。

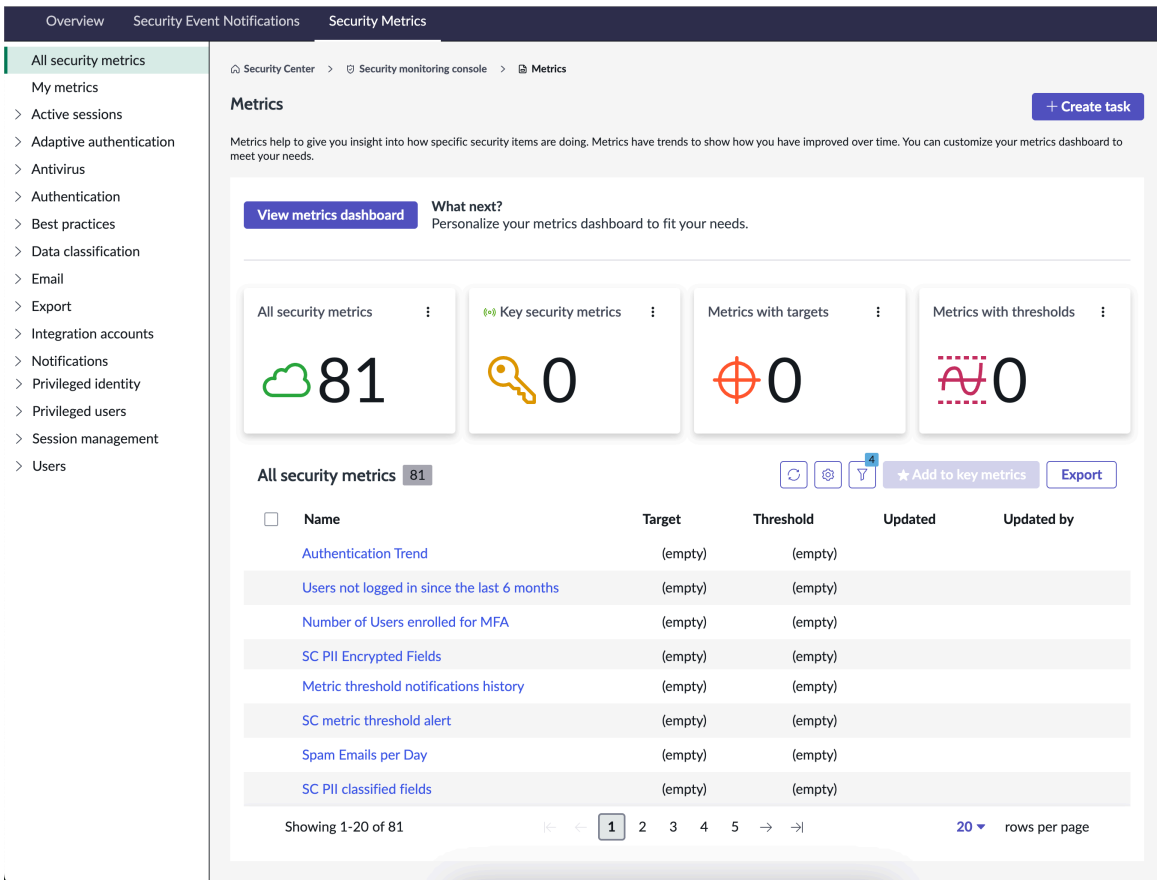
      注: しきい値は、[しきい値タイプ] が [次の値より小さい] または [より大きい] に設定されている場合にのみ入力する必要があります。

5. [保存] をクリックしてしきい値を作成します。
   次に、通知を設定する必要があります。
6. [すべて] を選択し、システム通知と入力します。
7. 移動先 メール > 通知.
8. [名前] 列に、*pa thre と入力します。
   しきい値通知はイベントベースで、[PA しきい値通知] テーブルで設定されます。
9. **[PA** しきい値通知] を選択します。
   結果には次の 3 つのタブがあります。
   - 送信時：メール通知を送信するには、何が起こる必要があるのかを指定します。この例では、上で設定したしきい値に達すると、メールがトリガーされます。
   - 受信者: 通知を受信するユーザーを指定します。ユーザーとグループを追加できます。
   - 内容：ユーザーとグループが受信するメッセージをカスタマイズするためにユーザーが変更できるスクリプトアクションを指定します。スクリプトアクションの作成の詳細については、「スクリプトアクション」を参照してください。

すべてのセキュリティメトリクス

[すべてのセキュリティメトリクス] に移動して、インスタンスのセキュリティメトリクスに関連するデータを含むテーブルを表示します。

セキュリティメトリクスダッシュボードにアクセスし、左側のリストから [すべてのセキュリティメトリクス] オプションを選択して、フィルタリング可能なリストのすべてのセキュリティメトリクスのリストにアクセスします。



リストの任意のアイテムを選択すると、各メトリクスの詳細が表示されます。

[+ タスクを作成] ボタンを選択して、メトリクスに関連するセキュリティタスクを作成します。セキュリティタスクの詳細については、「 セキュリティタスク」を参照してください。

アクティブセッション

ServiceNow AI Platform のアクティブユーザーのトレンドラインを表示します。

このページには、アクティブセッションに関連するメトリクスに関する情報を含むカードが表示されます。各カードには、次のメトリクスのトレンドラインが表示されます。

- ユーザーセッション：インスタンスのアクティブなユーザーセッションに関連するメトリクス
- 特権ユーザーセッション (Privileged user sessions)：特権ユーザーまたは追加のロールがアサインされたユーザーに関連するメトリクス

カードを選択すると、個々のメトリクスページが表示され、追加の詳細が表示されます。

[+ タスクを作成] ボタンを選択して、メトリクスに関連するセキュリティタスクを作成します。セキュリティタスクの詳細については、「 セキュリティタスク」を参照してください。

適応認証セキュリティメトリクス

認証ポリシーを使用して認証要求を評価し、指定されたポリシー条件に基づいてインスタンスへのアクセスを拒否または許可します。

適応認証メトリクスを使用すると、インスタンスでの適応認証の使用状況を監視できます。適応認証に関するすべてのメトリクスのサマリーと、ポリシー結果評価や拒否された IP アドレスなどの個々のメトリクスを表示します。このページでは、適応認証用の Adaptive Authentication (*com.snc.adaptive_authentication*) プラグインをインスタンスで使用できるようにする必要があります。メトリクスを表示するには、認証ポリシーを有効にする必要があります。詳細については、 適応認証のアクティブ化 および 適応認証プロパティの設定 を参照してください。

注: この機能はバージョン 1.2 でリリースされました。

- ポリシー結果評価：成功および失敗したすべての適応認証イベント
- イベント失敗の分布：イベントタイプごとの失敗したすべてのイベント
- イベント成功の分布：イベントタイプごとに関連付けられた成功イベント
- 拒否された IP アドレス：インスタンスによってブロックされた IP アドレスの数とその関連データ
- 認証済みのユーザーログイン：イベントタイプごとにカウントされたイベントの数 (ログイン前のイベントを除く)
- API ユーザーログイン：イベントタイプごとの API 認証ポリシーに関連付けられたイベントの数
- 認証傾向：記録されたイベントの合計数
- 認証ユーザー：イベントタイプごとにカウントされたユーザーの数 (ログイン前のイベントを除く)

カードを選択すると、個々のメトリクスページが表示され、追加の詳細が表示されます。

[+ タスクを作成] ボタンを選択して、メトリクスに関連するセキュリティタスクを作成します。セキュリティタスクの詳細については、「 セキュリティタスク」を参照してください。

ウィルス対策

感染した可能性があるファイルでイベントが発生したときの傾向を表示します。検出、隔離、復元、または削除がいつ行われたかを確認します。

このページには、インスタンスのウイルス対策アクティビティに関連するメトリクスに関する情報を含むカードが表示されます。各カードには、次のメトリクスのトレンドラインが表示されます。

- 隔離されたファイル：マルウェアが含まれている可能性があるファイルの数
- ダウンロードされたファイル (Downloaded files)：ダウンロードされたファイルの数
- 復元されたファイル (Restored files)：復元されたファイルの数
- 削除されたファイル：削除されたファイルの数

カードを選択すると、個々のメトリクスページが表示され、追加の詳細が表示されます。

[+ タスクを作成] ボタンを選択して、メトリクスに関連するセキュリティタスクを作成します。セキュリティタスクの詳細については、「 セキュリティタスク」を参照してください。

認証

マルチファクター認証 (MFA) の使用、Web サービスアカウント、生体認証スキャナーの使用状況など、認証スキームに関連するメトリクスの傾向を表示します。

このページには、認証に関連するメトリクスに関する情報を含むカードが表示されます。各カードには、次のメトリクスのトレンドラインが表示されます。

- MFA に登録されているユーザー
- MFA を使用しているユーザー
- 高特権非 MFA ユーザー
- アクティブな MFA ユーザー
- ロックアウトされた MFA ユーザー
- Web サービスアカウントのみ
- 失効まで 30 日の X509 証明書
- 生体認証スキャナー/ハードウェアキーユーザー
- アクセスポリシーのない REST API
- 「Web サービスへのアクセスのみ」フラグが有効になっていない統合アカウント

カードを選択すると、個々のメトリクスページが表示され、追加の詳細が表示されます。

[+ タスクを作成] ボタンを選択して、メトリクスに関連するセキュリティタスクを作成します。セキュリティタスクの詳細については、「 セキュリティタスク」を参照してください。

データ分類

ServiceNow インスタンスのデータ分類のセキュリティメトリクスにアクセスします。

データ分類を使用すると、任意のテーブルの既存の辞書エントリにデータ分類ラベルを適用できます。これらのラベルは、インスタンスでホストされているデータのタイプを可視化するのに役立ちます。これらの分類を使用して、プライバシー法に準拠し、業界の規制要件を満たします。

詳細については、「データ分類の概要 」を参照してください。

- 分類可能なデータ：分類可能なテーブルまたは列。
- 分類済みデータ：分類済みの辞書エントリ、テーブル、または列。

データはカスタマイズ可能なチャートにグラフィカルに表示され、ターゲット、しきい値、傾向、統計、予測の設定などの詳細な分析を提供します。また、データはテーブル内のレコードとして整理されます。詳細については、「アナリティクスハブ 」を参照してください。

カードを選択すると、個々のメトリクスページが表示され、追加の詳細が表示されます。

[+ タスクを作成] ボタンを選択して、メトリクスに関連するセキュリティタスクを作成します。セキュリティタスクの詳細については、「 セキュリティタスク」を参照してください。

認証メトリクス

ダッシュボードからインスタンスの認証に関連するメトリクスを表示します。

このページには、認証に関連するメトリクスに関する情報を含むカードが表示されます。各カードには、次のメトリクスのトレンドラインが表示されます。

- MFA に登録されているユーザー：MFA の使用に登録されているユーザーの合計数
- MFA バイパスを使用しているユーザー (Users using MFA bypass)：マルチファクター認証を回避しているユーザーの合計数
- 高権限非 MFA ユーザー:MFA を使用していない高権限ユーザーの合計数
- アクティブな MFA ユーザー：インスタンスでアクティブな MFA ユーザーの合計数
- ロックアウトされた MFA ユーザー：インスタンスでロックアウトされた MFA ユーザーの合計数
- Web サービスアカウントユーザー(Web service account user)：Web サービスアカウントのみを持つユーザーの合計数
- 期限が切れる X509 証明書 (X509 certificates expiring)：30 日後に期限が切れる X509 証明書の合計数
- 生体認証スキャナー/ハードウェアキーユーザー:生体認証スキャナーまたはハードウェアキーを使用してログインしたユーザーの合計数。
- アクセスポリシーのない REST API:REST API のアクセスポリシーなしでログインしたユーザーの合計数。
- Web サービスへのアクセスのみフラグが有効になっていない統合アカウント (Integration Accounts without web service access only flag enabled):統合アカウントの Web サービスへのアクセスのみのフラグが有効になっていないユーザーの合計数。

カードを選択すると、個々のメトリクスページが表示され、追加の詳細が表示されます。

[+ タスクを作成] ボタンを選択して、メトリクスに関連するセキュリティタスクを作成します。セキュリティタスクの詳細については、「 セキュリティタスク」を参照してください。

メール

外部で受信されているスパムメールに関連するデータを表示します。

ページには、1 日に受信した迷惑メールの数を表す、スパムメールの傾向ラインを示すカードが表示されます。

カードを選択すると、追加の詳細が表示されます。

[+ タスクを作成] ボタンを選択して、メトリクスに関連するセキュリティタスクを作成します。セキュリティタスクの詳細については、「 セキュリティタスク」を参照してください。

エクスポート

一般的にエクスポートされるデータと、エクスポートを実行するユーザーを検出します。

このページには、エクスポートに関連するメトリクスに関する情報を含むカードが表示されます。各カードには、次のメトリクスのトレンドラインが表示されます。

- 合計エクスポート数 (Total exports)：ユーザーによってエクスポートされたテーブルレコードの合計数
- 分類されたエクスポート (Classified exports)：アサインされたデータ分類別に要約されている、エクスポートされたテーブルレコードの合計数

カードを選択すると、個々のメトリクスページが表示され、追加の詳細が表示されます。

[+ タスクを作成] ボタンを選択して、メトリクスに関連するセキュリティタスクを作成します。セキュリティタスクの詳細については、「 セキュリティタスク」を参照してください。

統合アカウント

ServiceNow AI Platform で作成された統合アカウントに関する傾向を表示します。

このページには、 ServiceNow AI Platformで作成された統合アカウントに関連するメトリクスに関する情報を含むカードが表示されます。各カードには、次のメトリクスのトレンドラインが表示されます。

- 合計統合アカウント (Total integration accounts)：インスタンスの統合アカウントに関連する傾向
- アクティブ統合アカウント (Active integration accounts)：インスタンスの統合アカウントに関連する傾向
- 非アクティブ統合アカウント (Inactive integration accounts)：インスタンスで非アクティブな統合アカウントの合計数の傾向

カードを選択すると、個々のメトリクスページが表示され、追加の詳細が表示されます。

[+ タスクを作成] ボタンを選択して、メトリクスに関連するセキュリティタスクを作成します。セキュリティタスクの詳細については、「 セキュリティタスク」を参照してください。

特権 **ID**

特権 ID を持つユーザーのメトリクスに関連するデータを分析します。

このページには、特権 ID ユーザーとそのインスタンスでのアクティビティに関連するメトリクスに関する情報を含むカードが表示されます。各カードには、次のメトリクスのトレンドラインが表示されます。

- admin ログイン：admin ロールのユーザーによるログインの合計数
- 代理操作：impersonator ロールを持つユーザーにより行われた代理操作の合計数
- 昇格 (Elevation)：security_admin ロールを持つユーザーにより行われたセキュリティ昇格の合計数
- ServiceNow ログイン：ServiceNow 従業員によるログインの合計数
- 追加されたアドミンユーザー：admin ロールを付与されたユーザーの合計数

カードを選択すると、個々のメトリクスページが表示され、追加の詳細が表示されます。

[+ タスクを作成] ボタンを選択して、メトリクスに関連するセキュリティタスクを作成します。セキュリティタスクの詳細については、「 セキュリティタスク」を参照してください。

特権ユーザー

ServiceNow AI Platform での特権ユーザー (アクティブおよび非アクティブ) とそのアクティビティのトレンドラインを表示します。

[特権ユーザー概要 (Privileged users overview)] セクションに、ServiceNow AI Platform での特権ユーザー (アクティブおよび非アクティブ) とそのアクティビティのトレンドラインが表示されます。特権ユーザーは、高セキュリティ設定、インポート、ポータルユーザーなどの機能にアクセスするために、アドミニストレーターによって追加のロールがアサインされたユーザーです。

特権ユーザーの説明は次のとおりです。

- 合計ユーザー：インスタンスのユーザーの合計数
- アクティブユーザー：インスタンスでセッションを開始したユーザーの合計数
- 非アクティブなユーザー：最近インスタンスにログインしていないユーザー
- ロックアウトされていない非アクティブなユーザー (Inactive users who are not locked out)：最近ログインしていないが、まだ自分のアカウントにアクセスできるユーザー
- ロックアウトされたユーザー：アカウントへの認証が許可されていないユーザー
- 新規ユーザー：インスタンスに最近追加されたユーザー
- 成功したログイン (Successful logins)：正常にログインしたユーザー
- 失敗したログイン：失敗したログイン試行
- MFA で保護されていないローカルログイン (Local logins not protected by MFA)：MFA なしでログインしたユーザー
- ログインしていないユーザー (Users never logged in)：インスタンスにログインしたことのないユーザー
- 先月からログインしていないユーザー (Users not logged in since last month)：過去 30 日間ログインしていないユーザー
- 6 か月前からログインしていないユーザー (Users not logged in since the last 6 months)：過去 6 か月間ログインしていないユーザー
- 1 年前からログインしていないユーザー (Users not logged in since the last 1 year)：過去 1 年間ログインしていないユーザー
- パスワードのリセットが必要 (Need to reset password)：パスワードをリセットする必要があるユーザー
- ユーザーのパスワードリセットの失敗 (Password reset failures of the users：ユーザーあたりのパスワードの失敗数

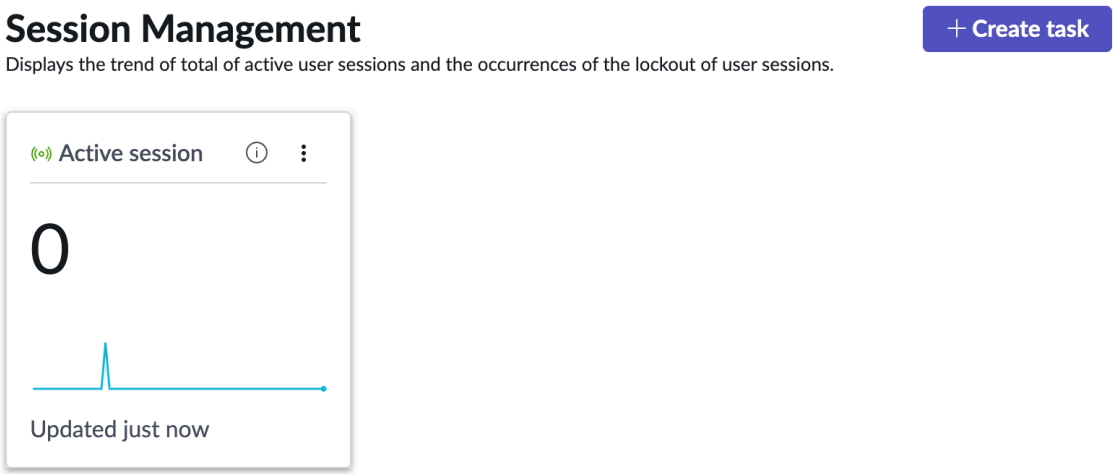
カードを選択すると、個々のメトリクスページが表示され、追加の詳細が表示されます。

[+ タスクを作成] ボタンを選択して、メトリクスに関連するセキュリティタスクを作成します。セキュリティタスクの詳細については、「 セキュリティタスク」を参照してください。

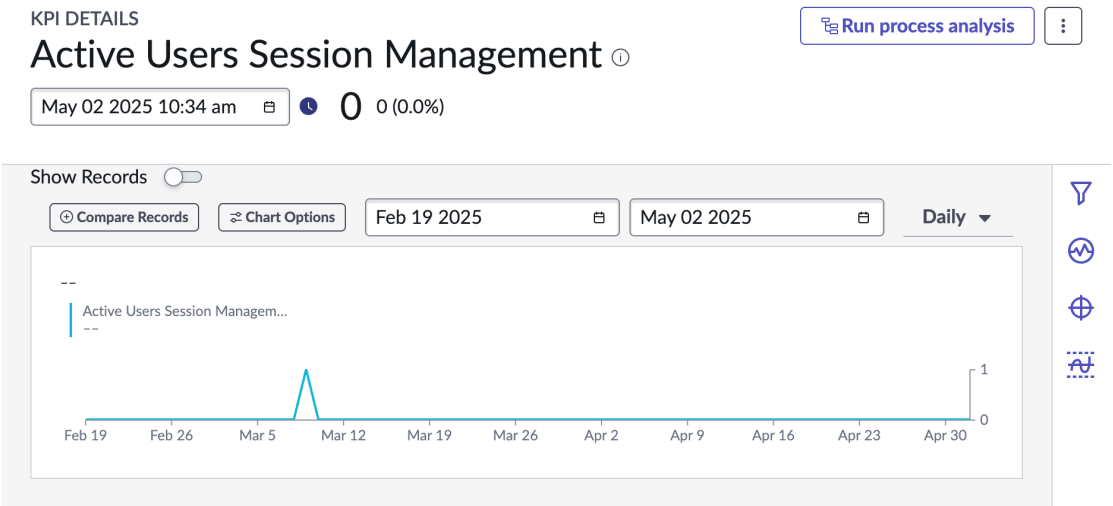
セッション管理

ユーザーセッションに関連するメトリクスとセッションのロックアウトの頻度を表示します。

このページには、インスタンス上のアクティブなセッションの数を表す、アクティブなセッションの傾向線がカードに表示されます。



カードを選択すると、追加の詳細が表示されます。



詳細ページから、画面の端にあるボタンを使用して、フィルター、KPI シグナル、ターゲット、およびしきい値を構成できます。

[+ タスクを作成] ボタンを選択して、メトリクスに関連するセキュリティタスクを作成します。セキュリティタスクの詳細については、「 セキュリティタスク」を参照してください。

ユーザー

ServiceNow AI Platform で合計ユーザー (アクティブおよび非アクティブ) とそのアクティビティのトレンドラインを表示します。

[ユーザー概要 (Users overview)] セクションに、ServiceNow AI Platform での合計ユーザー (アクティブおよび非アクティブ) とそのアクティビティのトレンドラインが表示されます。

ユーザーのタイプの説明は次のとおりです。

- 合計ユーザー：インスタンスのユーザーの合計数
- アクティブユーザー：インスタンスでセッションを開始したユーザーの合計数
- 非アクティブなユーザー：最近インスタンスにログインしていないユーザー
- ロックアウトされていない非アクティブなユーザー (Inactive users who are not locked out)：最近ログインしていないが、まだ自分のアカウントにアクセスできるユーザー

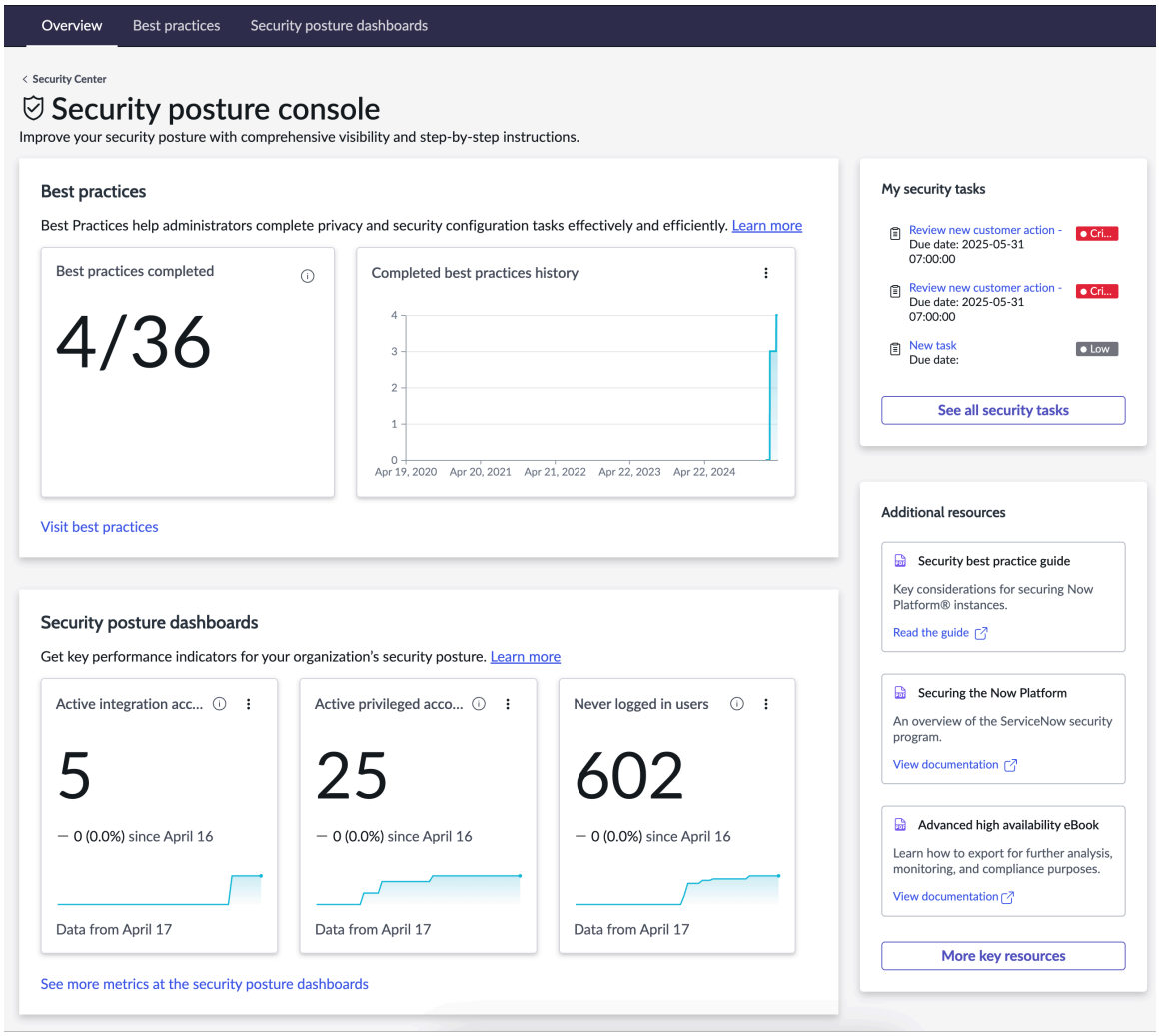
- 新規ユーザー：インスタンスに最近追加されたユーザー
- 成功したログイン (Successful logins)：正常にログインしたユーザー
- 失敗したログイン：失敗したログイン試行
- 外部ログイン：サードパーティを使用する認証
- MFA で保護されていないローカルログイン (Local logins not protected by MFA)：MFA を使用しないログイン
- 先月からログインしていないユーザー (Users not logged in since last month)：過去 30 日間ログインしていないユーザー
- 6 か月前からログインしていないユーザー (Users not logged in since the last 6 months)：過去 6 か月間ログインしていないユーザー
- 1 年前からログインしていないユーザー (Users not logged in since the last 1 year)：過去 1 年間ログインしていないユーザー
- パスワードのリセットが必要 (Need to reset password)：パスワードをリセットする必要があるユーザー
- ユーザーのパスワードリセットの失敗 (Password reset failures of the users：ユーザーあたりのパスワードの失敗数

カードを選択すると、追加の詳細が表示されます。

[+ タスクを作成] ボタンを選択して、メトリクスに関連するセキュリティタスクを作成します。セキュリティタスクの詳細については、「 セキュリティタスク」を参照してください。

セキュリティ体制コンソール

包括的な可視化と段階的な手順により、セキュリティ脅威を特定、対応、復旧する能力を向上させます。



セキュリティ体制コンソールは、ベストプラクティスとセキュリティ体制ダッシュボードに関する情報を提供するセクションに分かれています。これらのセクションでいずれかのカードを選択して、詳細を確認します。ページ上部のバーを使用して、このページと Security Center のベストプラクティスおよびセキュリティ体制ダッシュボードセクション間を移動することもできます。

ベストプラクティス

このセクションには、ベストプラクティスの数と、インスタンスに適用した数が表示されます。完了したベストプラクティスを経時的に示すチャートを表示できます。

セキュリティポスチャダッシュボード

このセクションには、アクティブな統合アカウント、アクティブな特権アカウント、および一度もログインしていないユーザーの数が表示されます。

アクティブな統合アカウント

統合アカウントは、インスタンスとサードパーティアプリケーション間の統合を管理するユーザーアカウントです。

アクティブな特権アカウント

特権ユーザーは、高セキュリティ設定、インポート、ポータルユーザーなどの機能にアクセスするために、アドミニストレーターによって追加のロールがアサインされたユーザーです。

一度もログインしていないユーザー

インスタンスにログインしたことのないユーザーアカウント。

自分のセキュリティタスク

自分にアサインされた最も緊急性の高いセキュリティタスクを表示します。タスクを選択して詳細を表示するか、[すべてのセキュリティタスクを表示] を選択してセキュリティタスクの完全なリストを表示します。

その他のリソース

画面の端にある [追加リソース] セクションを使用して、Security Center の学習に関連するドキュメントとビデオ、およびインスタンスのセキュリティの管理に関する情報に移動します。

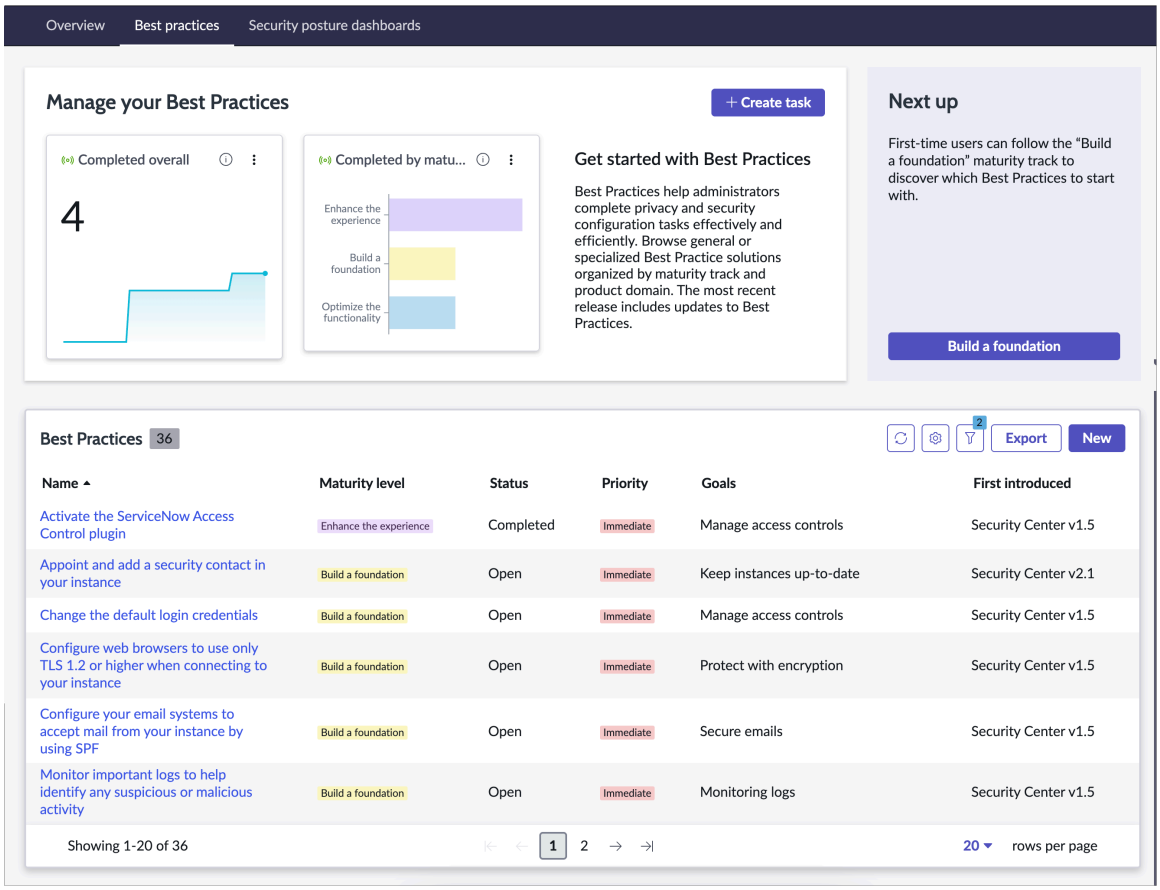
セキュリティのベストプラクティス

セキュリティのベストプラクティスを使用して、 ServiceNow インスタンスにプライバシーとセキュリティの構成タスクを実装します。

セキュリティ体制を改善するためのベストプラクティスを特定し、その実装方法に関するステップバイステップの指示に従います。セキュリティのベストプラクティスでは、次の情報が提供されます。

- ホームページには、セキュリティのベストプラクティスの実装の進捗状況の概要が表示されます。また、組織の目標に応じて、セキュリティのベストプラクティスのリストを整理し、管理することもできます。
- 概要ページには、各セキュリティのベストプラクティスの詳細、それらを実装する手順、およびすべてのアクティビティとコメントの記録が表示されます。
- タスクステップページには、セキュリティのベストプラクティスを実装する手順が記載されています。
- アクティビティページでは、セキュリティのベストプラクティスに関連するユーザーおよびシステムアクションの履歴を追跡します。

セキュリティのベストプラクティスのホームページ



ホームページには、[ベストプラクティスの管理] セクションが表示されます。グラフには、進捗状況の概要を示すグラフが含まれています。

全体で完了

完了したベストプラクティスの数と傾向ラインを表示します。カードを選択すると、[完了した全体的な メトリクス] ページが セキュリティ測定基準で表示されます。

成熟度レベルごとに完了

成熟度レベル別に整理された完了したベストプラクティスのチャートを表示します (成熟度レベルの説明については、次の表を参照してください)。カードを選択すると、セキュリティ測定基準で [成熟度レベル別の完了] メトリクスページが表示されます。

基盤を構築

[基盤のビルド] ボタンを選択して、このページのテーブルをフィルタリングし、 基盤のビルド 成熟度レベルのベストプラクティスのみを表示します。これらは、インスタンスセキュリティの改善を開始するために行うことができる、影響の少ない変更です。

タスクを作成

[+ タスクを作成] ボタンを使用して、ベストプラクティスの作業を追跡または委任するセキュリティタスクを作成します。セキュリティタスクの詳細については、「 セキュリティタスク」を参照してください。

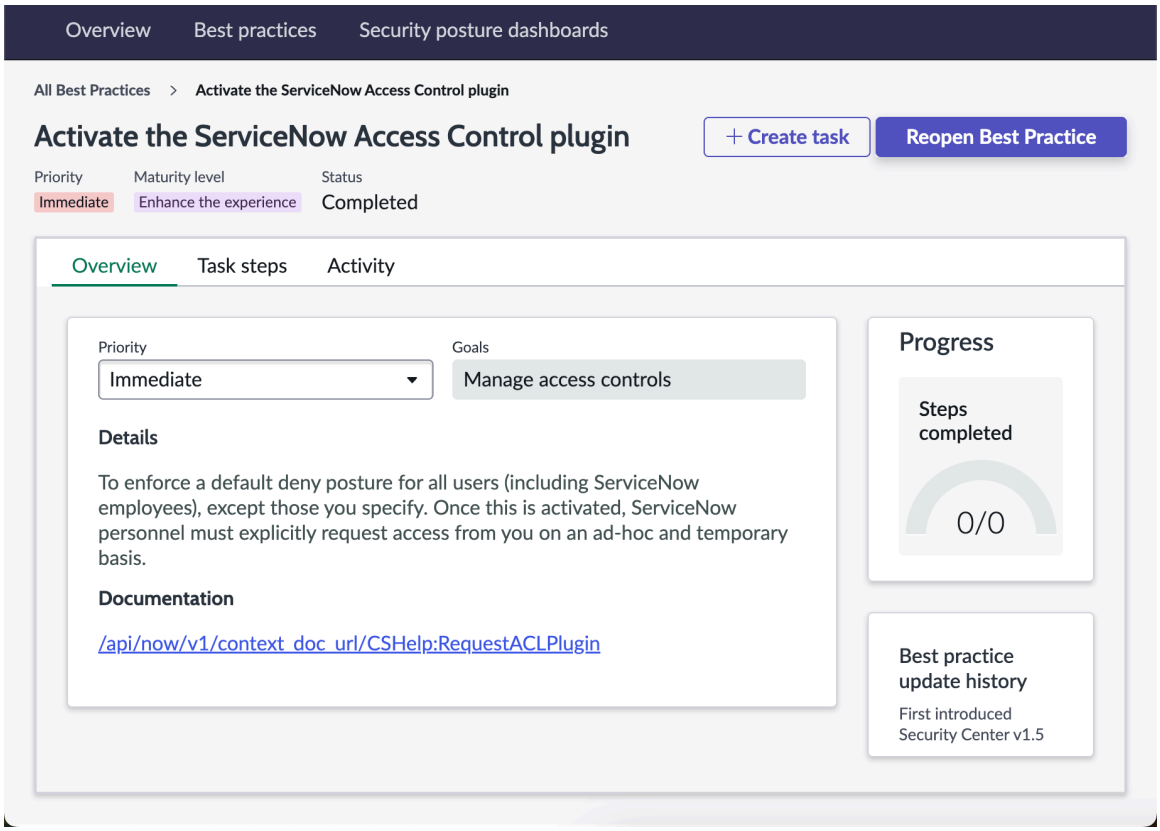
このテーブルを使用すると、フィルターを適用して絞り込まれたリストをソートして保存し、さまざまなユースケースやロールの作業リストとして使用できます。詳細については、「フィルター済みリストの保存」を参照してください。

以下にセキュリティベストプラクティステーブルに関連するフィールドについて説明します。

セキュリティベストプラクティステーブルで使用されるフィールドのまとめ

名前	説明
名前	セキュリティベストプラクティスの識別に使用する名前。
成熟度レベル	測定可能な結果が得られるように影響の大きさ順に並べられたアプリケーションと機能。成熟度レベルの値は次のとおりです。 • 基盤の構築 • エクスペリエンスを向上 • 機能の最適化 • 高度な機能を追加 これらは、クロール、ウォーク、ラン、フライの各段階と考えることもできます。
ステータス	ベストプラクティスの現在のステータス: • オープン • 進行中 • 完了
優先度	組織でベストプラクティスを実装するための重要度の順番: • 即時 • 後で • 適用外
目標	ベストプラクティスが対処するセキュリティカテゴリ: • 初期のセキュリティ構成に対応 • 安全なメール • 監視ログ • アクセスコントロールの管理 • 暗号化による保護 • インスタンスを最新の状態に保つ
最初の導入	ベストプラクティスが導入されたセキュリティセンターのバージョン。
変更済み	ベストプラクティスが変更されたセキュリティセンターのバージョン。
削除	ベストプラクティスが削除されたセキュリティセンターのバージョン。

セキュリティのベストプラクティスの詳細ページ



テーブルからベストプラクティスを選択して、そのページを表示します。詳細ページの上部には、優先度、成熟度レベル、ステータスなど、セキュリティのベストプラクティスに関する一般的な情報が表示されます。[ベストプラクティスの完了] ボタンを使用してプラクティスを完了としてマークするか、[ベストプラクティスを再オープン] ボタンを使用してプラクティスを完了としてマークします。**[+** タスクを作成] を使用して、このタスクを追跡および委任するセキュリティタスクを作成します。セキュリティタスクの詳細については、「 セキュリティタスク」を参照してください。

このページでは、ベストプラクティスに関する詳細情報をタブに分割して提供します。

概要

このタブには [優先度] ドロップダウンメニューがあり、現時点で重要で適用できないセキュリティのベストプラクティスを指定できます。

詳細セクションには、セキュリティベストプラクティスに関連する機能に関するコンテンツが表示され、ドキュメントセクションには追加情報を参照できる 1 つ以上のリンクがあります。

右側の [進捗状況] カードには、完了済みのステップ数と含まれるステップの合計数が表示されます。[次のステップに移動] を選択すると、次の未完了のステップに移動します。

[ベストプラクティス更新履歴] カードには、ベストプラクティスのリリース情報のスナップショットが表示されます。セキュリティベストプラクティスがリリースされた ServiceNow セキュリティセンターのバージョンと、その後最後に更新されたバージョンを追跡できます。

タスクのステップ

このタブでは、このセキュリティのベストプラクティスを実装する方法を順を追って説明します。詳細については、「セキュリティのベストプラクティスの完了」を参照してください。

アクティビティ

このタブには、タイムスタンプ付きのアクティビティが新しいものから古いものの順に一覧表示されます。検索とフィルターを使用して情報をクエリします。詳細については、「セキュリティベストプラクティステーブルへのフィルターの適用」を参照してください。

セキュリティのベストプラクティスの完了

ServiceNow インスタンスでセキュリティのベストプラクティスを完了する方法について説明します。

始める前に
必要なロール：admin

このタスクについて
ServiceNow インスタンスにセキュリティのベストプラクティスを実装するには、次の手順を実行します。

手順

1. セキュリティのベストプラクティスにアクセスするには、ServiceNow インスタンスで [すべて] を選択し、「ベストプラクティス」と入力します。
2. フィルターを適用して、実装するセキュリティのベストプラクティスを表示します。
   a. UI の [ベストプラクティス] テーブルに移動します。
   b. ケバブメニュー、または [ステータス] 列で縦に並んだ 3 つの点 (⋮) の項目を選択します。
   c. [オープン] および [進行中] オプションを選択し、[適用] を選択します。
3. 実装するセキュリティベストプラクティスの名前を選択します。
4. [次のステップに移動] を選択します。
5. [タスクのステップ] リストからタスクステップを選択します。
6. 下向きキャレット (v) アイコンを選択すると、そのステップの実装手順が表示されます。
7. 手順を読み、ServiceNow インスタンスに実装します。
8. タスクが完了したら、[ステップを完了としてマーク] を選択します。

   注: 後で戻ってくる場合や、組織の目標に当てはまらない場合は、タスクをスキップできます。さらに、タスクステップを選択し、[ステップを再起動] を選択することで、完了したタスクステップを開き直せます。

9. [ベストプラクティスを完了] を選択します。

   セキュリティのベストプラクティスはいつでも完了できます。すべてのステップを完了する必要はなく、スキップしても問題ありません。ただし、監査参照のすべてのステップを完了していないのにセキュリティのベストプラクティスを完了する場合は、[アクティビティ] サブタブにコメントを挿入することを検討してください。

ベストプラクティスのアクティビティの表示

ServiceNow インスタンスで実行しているセキュリティのベストプラクティスに関連する、時系列順のタイムスタンプ付きの履歴と、アクティビティを開始したユーザーを追跡します。

始める前に
必要なロール：admin

手順

1. セキュリティベストプラクティスマネージャーにアクセスするには、ServiceNow インスタンスで [すべて] を選択し、「ベストプラクティス」と入力します。
2. [ベストプラクティス] テーブルの [名前] 列から、アクティビティを表示するセキュリティベストプラクティスを選択します。
3. [アクティビティ] タブを選択します。

   表示できるアクティビティの例にはフィールドの変化があります。セキュリティベストプラクティスのステータスが [オープン] から [完了] に変わった場合や、タスクステップが完了またはスキップされた場合などです。また、テキストボックスにコメントを入力することで、アクティビティに関連する追加情報を入力することもできます。

完了したベストプラクティスのデータの表示

完了したベストプラクティスの合計数を、視覚的にまたは成熟度レベル別にセグメント化して表示します。

始める前に
必要なロール：admin

このタスクについて
完了したセキュリティのベストプラクティスの視覚的なトレンドを全体的および成熟度レベルごとに示すチャートを表示するための手順について説明します。

手順

1. セキュリティセンターアプリにアクセスし、[ベストプラクティス] タブに移動します。
2. [ベストプラクティスの管理] カードで、[全体で完了] チャートを選択します。
3. チャートを表示して、完了したセキュリティベストプラクティスの構成に関するトレンドを確認します。
4. [成熟度レベルごとに完了] を選択すると、完了したベストプラクティスのトレンドを成熟度レベル別に表示するチャートが表示されます。

セキュリティベストプラクティステーブルをフィルターします

セキュリティのベストプラクティスに関連性の高い結果が返ってくるようにフィルターを適用します。

始める前に
必要なロール：admin

手順

1. セキュリティベストプラクティスマネージャーにアクセスするには、ServiceNow インスタンスで [すべて] を選択し、「ベストプラクティス」と入力します。
2. ベストプラクティステーブルに移動し、ケバブメニュー、または縦に並んだ 3 つの点があるフィールド (⋮) を選択して、フィルターを適用するフィールドを選択します。

フィールド	フィルター
名前	条件演算子とテキスト文字列を組み合わせ、入力した名前に一致するセキュリティベストプラクティスを返します。

フィールド	フィルター
成熟度レベル	フィルターを適用して、次の成熟度レベルの 1 つ以上に一致するセキュリティのベストプラクティスを返します。 ○ 基盤の構築 ○ エクスペリエンスを向上 ○ 機能の最適化 ○ 高度な機能を追加
ステータス	フィルターを適用して、次のステータスの 1 つ以上に一致するセキュリティベストプラクティスを返します。 ○ 空 ○ オープン ○ 進行中 ○ 完了
優先度	フィルターを適用して、次の優先度に一致するセキュリティベストプラクティスを返します。 ○ 空 ○ 即時 ○ 後で ○ 適用外
ゴール	フィルターを適用して、次の目標の 1 つ以上に一致するセキュリティベストプラクティスを返します。 ○ 初期のセキュリティ構成に対応 ○ インスタンスを最新の状態に保つ ○ アクセスコントロールの管理 ○ ログのモニタリング ○ 暗号化による保護 ○ 安全なメール 注：詳細フィルターはデフォルトで目標に適用されます。変更するには、次を選択します: 変更を加える > 詳細ビュー > フィールド目標を選択 をクリックし、残りの条件を作成して [更新] を選択します。
最初の導入	条件演算子とテキスト文字列を組み合わせて、入力したバージョンで最初に導入されたベストプラクティスを返します。

フィールド	フィルター
変更済み	条件演算子とテキスト文字列を組み合わせて、入力したバージョンで変更されたセキュリティベストプラクティスを返します。
削除	条件演算子とテキスト文字列を組み合わせて、入力したバージョンで削除されたセキュリティベストプラクティスを返します。

後で使用するためのフィルターの保存

後で再利用できるように、セキュリティベストプラクティスのフィルターを保存する手順について説明します。

始める前に

必要なロール：なし

このタスクについて

セキュリティベストプラクティステーブルでフィルターを作成した後、それを保存して、次回インスタンスへ認証するときに使用できます。作成できるフィルターのタイプに柔軟性を持たせたい場合は、詳細フィルターを適用します。

手順

1. [フィルターを表示] ( ) パネルアイコンを選択します。
2. [フィルター] ダイアログパネルで、下にスクロールして [詳細表示] を選択します。
3. フィールド、演算子、および値を含む条件を追加して、フィルターをビルドします。
4. [フィルターの保存] を選択します。
5. フィルター名を入力します。
6. 権限を選択します。
7. [保存] を選択します。
8. [更新] を選択して、保存したフィルターをセキュリティベストプラクティステーブルに適用します。

保存済みフィルターの使用

セキュリティベストプラクティスの保存済みフィルターを使用する手順について説明します。

始める前に

必要なロール：なし

このタスクについて

インスタンスからログアウトすると、フィルターはデフォルト設定にリセットされます。次の手順は、保存済みのフィルターを取得して、セキュリティベストプラクティステーブルに再適用する方法を示しています。

手順

1. [フィルターパネルを表示] ( ) アイコンを選択します。
2. [フィルター] ダイアログボックスで、下にスクロールして [詳細表示] を選択します。

3. [既存のフィルターを使用] を選択し、ドロップダウンリストから保存済みフィルターを選択します。
4. [更新] を選択して、保存済みフィルターを適用します。

ベストプラクティス

セキュリティ体制コンソールのベストプラクティスの詳細について説明します。

ベストプラクティス	説明
ServiceNow Access Control プラグインを有効にする	ServiceNow Access Control プラグインを使用して、どの従業員がいつインスタンスにアクセスできるかを制御します。指定したユーザーを除くすべてのユーザーに対してデフォルトの拒否体制を適用します。これには、 ServiceNow 人の従業員を含めることができます。このプラグインを使用すると、インスタンスへの不要なアクセスを防ぐことができます。 注: ServiceNowアクセス制御がアクティブ化された後、担当者はアドホックかつ一時的に、お客様からのアクセスを明示的に要求する必要があります。 このプラグインの詳細については、「ServiceNow アクセス制御」を参照してください。
インスタンスでセキュリティ連絡先を任命して追加します	セキュリティチームからセキュリティ関連情報を受け取る組織内の情報セキュリティ連絡先を選択します。この連絡先に加えて、これらの更新を受け取るアドミンも含まれます。 この情報は、セキュリティの問題、セキュリティ アラート、または重要なソフトウェア更新プログラムに関する詳細である可能性があります。 セキュリティ連絡先の追加の詳細については、「KB0621516」を参照してください。
デフォルトのログイン認証情報を変更	アドミン、ITIL、従業員などのインスタンスのビルトインユーザーアカウントのパスワードを変更します。これらのアカウントは、インスタンスに固有のデフォルトのパスワードでプロビジョニングされますが、できるだけ早く変更する必要があります。 インスタンスのユーザーアカウントのパスワードを変更する方法の詳細については、「 ユーザーのパスワードの設定」を参照してください。
インスタンスへの接続時に TLS 1.2 以降のみを使用するように Web ブラウザを設定する	インスタンスに接続するブラウザーで、より安全なトランスポートレイヤーセキュリティ (TLS) 1.2 を使用していることを確認します。この変更は、ブラウザーで行うか、Web プロキシまたは他のゲートウェイによって強制することができます。 TLS 1.2 のみを使用するようにこれらの製品を設定する手順については、ブラウザ、Web プロキシ、またはゲートウェイのドキュメントを参照してください。

ベストプラクティス	説明
SPF を使用してインスタンスからのメールを受け入れるようにメールシステムを設定する	組織で Sender Policy Framework (SPF) を使用してスパム対策テクノロジーで受信メールを制御している場合は、インスタンスから送信されたメールを受け入れるように設定する必要があります。SPF レコードを動的にクエリするように SPF を設定します。 SPF を選択できない場合は、メール サーバーの IP アドレスを許可リストに追加するという方法もあります。アドレスは変更される可能性があるため、この設定を監視する必要があります。 これらのソリューションの手順と詳細については、「KB0535456 」を参照してください
ファイルの添付、アップロード、ダウンロードを制限することを検討する	添付ファイルのアップロードをロール、ファイル拡張子、MIME タイプ、またはサイズで制限して、悪意のある可能性のあるファイルが保存されてインスタンスから配信されるのを防ぎます。また、ダウンロードできるファイルの種類 (MIME の種類など) を制御したり、認証されていないユーザーによる画像アクセスを防止したりすることもできます。 これらの添付ファイル制限は、インスタンスのシステムプロパティによって制御されます。構成の詳細については、「 添付ファイルのシステムプロパティの構成 」を参照してください。
ブラウザの SQL メッセージを無効にする	SQL エラーメッセージが Web ブラウザに表示されないようにします。これらのメッセージは、ユーザーや開発者にとって有用ですが、攻撃者がシステムに関する情報を取得したり、データへのアクセスを誘導したりするために使用される可能性があります。これらのメッセージは、システムプロパティを使用してオフにすることができます。 このシステムプロパティの詳細については、「 SQL エラーメッセージを無効にする (Security Center 1.3 および 1.5 で更新)」を参照してください。
パスワードを使わない認証を無効にする	可能な場合はパスワードなしの認証を無効にすることで、強力な認証を確保します。パスワードなしの認証を無効にしないと、潜在的な攻撃者はユーザー名 (firstname.lastname やロールタイトルなど) を正しく推測してインスタンスにアクセスできる可能性があります。 システムプロパティを使用して、インスタンスでパスワードなしの認証を無効にすることができます。このプロパティの詳細については、「パスワードなしの認証の無効化」を参照してください。
重要または機密データのテーブル監査を有効にする	テーブル監査を使用して、データの変更を追跡します。監査は、有効になっているテーブル内のすべてのレコードの作成、更新、削除を追跡し、アドミンが重要なデータや機密データの変更を追跡できるようにします。アドミニストレーターは、監査対象のテーブル内の特定のフィールドを選択して、より的を絞った結果

自動翻訳

ベストプラクティス	説明
	を表示したり、パフォーマンスへの影響を軽減したりすることもできます。 インスタンスでの監査の詳細については、「 監査」を参照してください。 テーブルの監査を有効にする具体的な手順については、「 テーブルの監査の構成」を参照してください。
インスタンス内に置かれているデータを暗号化する	データを暗号化して、機密性と完全性を維持します。インスタンスのデータはデータベース内に置くことができます。バックエンドでデータボリュームを透過的に暗号化する機能をサブスクライブすることもできます。インスタンスが実行されている物理ディスク全体を暗号化して、紛失や盗難の場合にデータを保護することもできます。 ユースケースと軽減するリスクに応じて、インスタンスに保存されているデータに対して異なる暗号化方法を同時に使用できます。たとえば、ほとんどのテーブルでデータベース暗号化を使用し、データボリューム全体でクラウド暗号化を使用して、保存データを透過的に暗号化することを選択できます。また、フルディスクハードウェア暗号化を使用することもできますが、これにはドライブやサーバーの盗難から保護するための専用環境も必要です。 in キー管理フレームワーク で利用可能な暗号化オプションを確認します。
強力なパスフレーズの使用を強制する	パスワードポリシーを使用して、インスタンスのネイティブアカウントとローカルアカウントの長さ、複雑さ、有効期限、一意性、ロックアウトなどを適用します。これらのポリシーを使用して、セキュリティを最大化し、長いパスフレーズの採用を促進し、単純なパスワードの使用を排除します。 LDAP や SAML など、統合した外部認証サービスの既存のポリシーを保持できます。 パスワードポリシーの設定の詳細については、「 パスワードポリシーの設定」を参照してください。
アカウントの自動作成を徹底する	この機能を使用して、メールでユーザーアカウントを動的に作成します。この機能は、ビジネスニーズに必要な場合にのみ、アカウントを作成できる信頼できるドメインのリストを定義した後にのみアクティブ化してください。この方法で作成された新しいアカウントにパスワードを割り当てる方法を制御することもできます。 ユーザーの自動作成の詳細については、「 ユーザーの自動作成の有効化 」を参照してください。
ナレッジベースへの安全なアクセスの確保	ナレッジベースと記事へのアクセスを管理して、安全で効率的な情報共有を実現します。寄稿および読み取りアクセスを制御することで、特定のユーザーまたはユーザーのカテゴリがナレッジ

ベストプラクティス	説明
	ベースおよびナレッジ記事にアクセスできるかどうかを決定できます。 具体的な構成は、ビジネスニーズによって異なります。ナレッジアクセス権を設定するためのオプションについては、「ナレッジベースとナレッジ記事へのアクセスの管理」を参照してください。
高いセキュリティプラグインがインストールされ、アクティブになっていることを確認する	High Security プラグイン (HSP) を使用して、セキュリティ管理を強化し、適切な設定を適用します。高セキュリティ設定は、セキュリティ設定の中心的な場所を提供し、個別のセキュリティアドミニストレーターロール、デフォルトの拒否プロパティ、およびその他の重要なセキュリティ機能を作成します。 HSP は、すべての新しいインスタンスにインストールされ、デフォルトで有効になっています。古いバージョンからアップグレードされたインスタンスを含む、古いインスタンスの HSP アクティベーションを要求できます。HSPの有効化は、アクティブ化によっていくつかの基本的なプロパティと動作が変更されるため、非本番環境で慎重にテストした後にのみ行う必要があります。 High Security プラグインの詳細については、「高セキュリティプラグインを有効にする (Security Center 1.3 で更新)」を参照してください。
NOW セキュリティリソースについて理解します	セキュリティ情報は常に進化しているため、情報セキュリティを強化するために、セキュリティリソースを常に最新の状態に保つことが重要です。 次のリソースを使用して、セキュリティリソースに関する最新情報を常に入手してください。 • CORE ディレクトリ:ServiceNow CORE コンプライアンスポータル • Securing the ServiceNow AI Platform: How ServiceNow protects customer data (ServiceNow AI プラットフォームの保護: ServiceNow による顧客データの保護) • インスタンスの保護
インスタンスの強化	セキュリティセンター強化ツールを使用して、悪用される可能性のある弱点を制限することでリスクを軽減し、インスタンスのセキュリティを強化するための推奨設定を実装します。 セキュリティセンターの詳細については、 セキュリティセンターを参照してください。 ハードニング設定で利用可能なハードニング設定を確認します。
パッチをできるだけ早くインストールする	パッチとプラットフォームの更新をできるだけ早くインストールすることで、インスタンスと他の顧客のインスタンスの両方に最高レベルのセキュリティを確保できます。更新を最新の状態に保

自動翻訳

ベストプラクティス	説明
	つことで、EOL ポリシーに準拠することで継続的なサポートを維持することもできます。アップグレードセンターを使用してプロセスを管理します。 Now Platform のセキュリティ修正は、製品の機能更新に伴うパッチとホットフィックスを通じて定期的にリリースされます。新しいパッチとホットフィックスが利用可能になったときにアップグレードすると、潜在的な脆弱性のリスクを軽減するのに役立ちます。 Now Platform のリリース、パッチ、ホットフィックスに関する情報は、製品ドキュメントの「リリースノート」セクションに記載されています。詳細については、「フェーズ 1 - 」を参照してください。
MFA との連携	サードパーティのマルチファクター認証 (MFA) を既存の SAML IdP と統合して、ログインセキュリティを強化します。認証には複数の認証要素が必要なため、MFA は高レベルのセキュリティを提供します。ユーザーが知っているもの (パスワード) とユーザーが所有しているもの (ワンタイムコード、携帯電話、または指紋などの生体認証属性) です。 MFA 統合の詳細については、「 多要素認証」を参照してください。
承認されたメール送信者ドメインを制限する	システムアドレスフィルターを使用して、インスタンスがメールで通信できるドメインとユーザーを制御します。これらのフィルターは、要件に合わせてカスタマイズできます。 信頼できないメールドメインと信頼できるメールドメインの指定で信頼できるドメインを構成する方法について説明します。
重要なログを監視して、疑わしいアクティビティや悪意のあるアクティビティを特定するのに役立てる	システムログモジュールは、インスタンス内で発生するトランザクションやイベントのトラブルシューティングとデバッグに使用できるさまざまなログを提供します。 イベントログ イベントログは、ログインイベント (成功またはそれ以外) や権限エスカレーションなど、システムアクティビティに関する多くの情報を明らかにします。 システムログ システムログには、構成の変更、システムエラー、ワークフロー、送受信データ接続など、一般的なアクティビティに関する広範な情報が含まれています。 注: イベントログとシステムログを使用して、担当者によるアクティビティの監査証跡を提供することもできます。 トランザクションログ トランザクションログは、インスタンスのすべての Web ブラウザ関連のアクティビティを記録し、行

自動翻訳

ベストプラクティス	説明
	われたすべての要求の詳細を提供できます。トランザクションログは、異常なアクティビティや悪意のあるアクティビティを識別するのに役立ちます。 テーブル監査とレコード履歴 データベーステーブルの監査を有効にします。レコード履歴は永続的であり、作成以降にデータに加えられた変更の詳細を追跡して表示できます。デフォルトでは、インシデント、問題、および変更テーブルのみが追跡されます。その他のテーブルについては、監査を手動で有効にする必要があります。 インポートログ インポートログを確認することで、インスタンスへのデータインポートアクティビティに関連する詳細情報を表示できます。これらのログには、ソースとステータス、時刻などに関する情報が含まれます アウトバウンド Web サービスログ これらは REST および SOAP 要求アクティビティを示し、外部サービスへの接続のボリュームと宛先を追跡するのに役立ちます。 システムログの詳細については、 システムログを参照してください。
ログインの失敗率を監視し、アラートを作成する	特に短期間での、ログイン失敗の多さなどの異常なアクティビティを監視します。定義したしきい値を超えたときにメールを送信するアラートを作成できます。 これらのしきい値を インジケーター しきい値 で設定する方法について説明します。
セキュリティイベントのモニタリング	[自分のセキュリティメトリクス] ダッシュボードを確認して、インスタンスで利用可能なセキュリティメトリクスを確認し、注目アクティビティに関するメール通知を生成するためのしきい値を設定します。注目すべきアクティビティの例には、次のようなものがあります。 権限エスカレーション アドミン、ITIL_Admin、またはより高い権限を持つその他のロールなどの特権ロールに予期しない変更が加えられた場合、不審なアクションを示す可能性があります。 ログイン失敗 ログイン失敗の数やパターンが異常であれば、総当たり攻撃やパスワードスプレー攻撃の可能性が明らかになります。 アドミンと高権限ユーザーが追加されました 新しい admin アカウントの作成の有効性をチェックして、不正な特権アクセスの試行を防ぐ必要があります。

自動翻訳

ベストプラクティス	説明
インスタンスのハードニングコンプライアンスレベルの監視	Security Center の強化ツールを使用して、インスタンスが最新のセキュリティ強化メトリクスに準拠していることを確認します。非本番インスタンスでこのツールにアクセスして、環境への影響を評価します。理想的には、製品の機能に影響を与えずに、スコアをできるだけ 100% に近づけ、最低スコアを 83% にする必要があります。 Security Center のセキュリティ強化設定ツールの詳細については、 ハードニング設定を参照してください。
開発者にセキュアコーディングガイドを参照するよう勧める	安全なコーディングプラクティスを使用して、インスタンスの安全性を確保し、不正アクセスに対する耐性を可能な限り確保します。インスタンス開発者向けの ServiceNow セキュアコーディングガイドでは、ServiceNow が提供するアプリケーションセキュリティ関連の GlideScriptable クラスとメソッドの概要を説明しています。このガイドは、開発者がターゲットインスタンスでコードを作成および変更する際に支援および教育することを目的としています。インスタンス開発者向けの ServiceNow セキュアコーディングガイド のガイドを確認してください。
[記憶する] チェックボックスを削除する	[記憶する] 機能を非アクティブ化することで、インスタンスへの不要なアクセスを防ぐことができます。この機能を有効にすると、Cookie がユーザーのコンピューターに保存され、その後のアクセス時にユーザーが自動的に認証されます。これにより、ユーザーが共有コンピューターなどの安全でないエンドポイントからインスタンスにアクセスした場合、セキュリティ上の問題が発生する可能性があります。 この機能の詳細と、 記憶する でこの機能を非アクティブ化する方法について説明します。
不明な IP アドレスからのインスタンスへのアクセスを制限する	組織とは無関係の IP アドレスからのアクセスを制限することで、インスタンスへの不正アクセスを防止します。許可されていない IP アドレスからインスタンスにアクセスしようとすると、拒否されます。このアプローチを使用する場合は、ゲートウェイまたは Web プロキシの外部アドレス、およびユーザーがインスタンスにアクセスするアドレス (リモート ユーザーを含む) のみを許可することを検討してください。IP アドレスによって送信アクセスと受信アクセスの両方を制限できます。 IP アドレスでインスタンスへのアクセスを制限する方法については、 特定の IP 範囲プラグインへのアクセスを制限する (Security Center 1.3 で更新)を参照してください。
パスワードスプレー攻撃に関する ServiceNowのガイダンスを確認する	パスワードスプレー攻撃からインスタンスを保護します。これらの攻撃は、一般的に使用されるパスワードを複数のアカウントに対して連続してテストすることにより、アクセスを試みます。 スプレー攻撃の詳細と、それらからインスタンスを保護する方法については、「 パスワードスプレー攻撃の緩和戦略 」を参照してください。

自動翻訳

ベストプラクティス	説明
共有セキュリティモデルの確認	責任共有モデルを確認して、インスタンスのセキュリティを維持する上での顧客としての共通のロールを理解します。責任共有モデルでは、顧客との間に特定の責任があるパートナーシップを定義します。 詳細については、「 ServiceNow 責任共有モデル 」を参照してください。
アーカイブおよび参照のためのインスタンスからのログデータの転送	ログデータをアーカイブして、デフォルトの 21 日間のログローテーション期間を超えて保持します。このアーカイブは、Web サービス要求、データエクスポート機能、MID サーバー、または Vault パッケージのログエクスポートサービスを使用して実行できます。 これらの方法の詳細については、次のリソースを使用します。 • Web サービス • ログエクスポートサービスの詳細 (LES)
RBAC で暗号化モジュールを使用して、データアクセス制御をさらに強化する	キー管理フレームワーク (KMF) を使用し、ロールベースのアクセス制御 (RBAC) を使用してインスタンス上のデータを保護する方法について説明します。KMF では暗号化モジュールを使用します。これにより、インスタンス上のどのデータを暗号化するか、および使用する暗号化方法を定義できます。複数のモジュールを使用して、インスタンスのさまざまな領域を異なる仕様で暗号化できます。 KMF とそのコンポーネントを使用して キー管理フレームワークの詳細 でデータを暗号化する方法について説明します。 暗号化モジュールの概要 の暗号化モジュールについて説明します。
連携プロバイダーでの証明書ベースの認証の使用	証明書ベースの認証を使用するように REST/SOAP 接続を使用して統合プロバイダーへのトラフィックを構成します。Secure Socket Layer (SSL) 証明書認証は、転送中のデータを暗号化し、送信時に読み取られないようにします。 この構成の詳細については、「 相互認証の構成」を参照してください。
SAML 認証を使用	サードパーティのマルチファクター認証 (MFA) を既存の SAML IdP と統合して、ログインセキュリティを強化します。認証には複数の認証要素が必要なため、MFA は高レベルのセキュリティを提供します。ユーザーが知っているもの (パスワード) とユーザーが所有しているもの (MFA トークンや携帯電話によって生成されたワンタイムコード、または指紋などの生体認証属性)。 ServiceNow は、ローカルアカウント、LDAP、SAML を使用した SSO、OIDC、またはダイジェストとの直接 MFA 統合をサポートしています。

ベストプラクティス	説明
	適応認証は、MFA を使用した SSO の前提条件です。 MFA は、指定したユーザーおよび指定したロールに対して有効にし、使いやすさを考慮して構成できます。たとえば、認識されたデバイスを数時間免除することができます。 セキュリティセンターで MFA を使用するためのメトリクスを表示できます。 SAML 認証の詳細については、次のリソースを参照してください。 • SAML 2.0 のコンセプト • マルチプロバイダー SSO を使用した SAML 2.0 構成
メールフィルター機能セットを使用して、疑わしい受信メッセージを処理する	メールフィルターを作成して、 ServiceNow ウイルス対策保護によって不審とマークされたメッセージを除外します。ウイルス対策保護は、ウイルス対策に加えて、電子メールのマルウェアとスパムを分析し、スコアリングとその結果により、これらの情報を x ヘッダーのメッセージに追加します。必要に応じて、これらのヘッダーをメールフィルタープラグインの基準として使用できます。 ServiceNowのウイルス対策機能の詳細については、アンチウイルススキャンをご覧ください。 インスタンスでメールフィルターを構成する方法については、「メールフィルター」を参照してください。
Syslog プローブを使用して SIEM にログを送信する	ServiceNow syslog プローブを使用して、インスタンスからセキュリティ情報およびイベントマネージャー (SIEM) にログメッセージを送信します。SIEM は、アクティビティの監視とセキュリティイベントの識別に使用できるサードパーティのソフトウェアまたはサービスです。 syslog プローブ構成の詳細についてはServiceNowSyslog プローブ を参照してください。
自分の成熟したメールセキュリティ環境を使用	独自の (またはサードパーティの) インフラストラクチャを使用してインスタンス関連のメールを送受信し、より正確な境界メール制御のメリットを活用することを検討してください。 独自の SMTP、POP3、または IMAP サーバーを使用することで、インスタンスに送信する前にメールをフィルタリングして受信する方法を制御できます。 注: 独自のメールインフラストラクチャの構成は高度なメール構成と見なされ、オプションで OAuth 2.0 メール認証を介してサードパーティのメールインフラストラクチャを使用できます。詳細については、各自のメールベンダーのドキュメントと手順を参照してください。

ベストプラクティス	説明
アクセスアナライザーを使用してアクセスを検証する	ServiceNowアクセスアナライザーツールを使用して、選択したユーザー、ロール、またはグループの権限を比較および分析できます。この情報を使用して、アクセスの問題をトラブルシューティングし、機密データにアクセスできるユーザーを特定し、インスタンス上のユーザーの適切なアクセスレベルを決定できます。 アクセスアナライザーのメリットの詳細については、 アクセスアナライザーの概要をご覧ください。

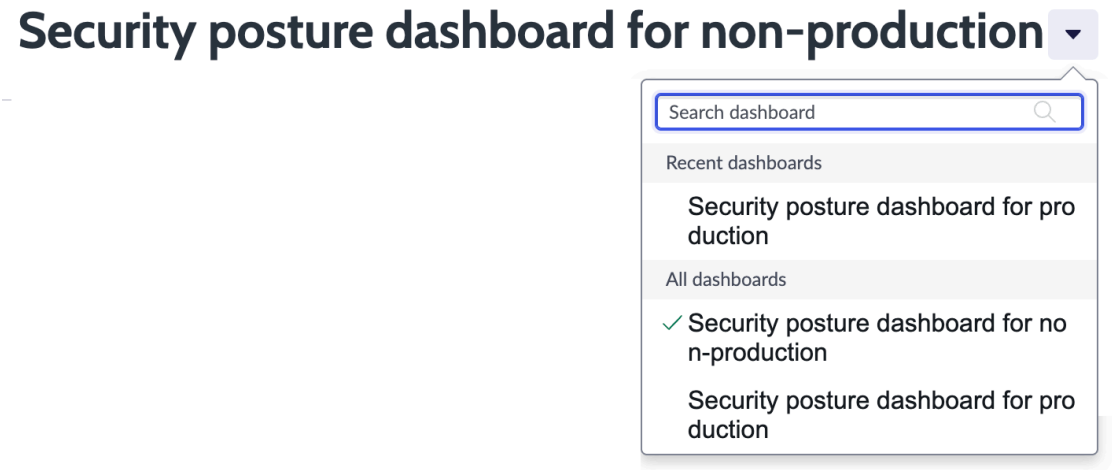
セキュリティ体制ダッシュボード

カスタマイズ可能な単一インスタンスおよびマルチインスタンスのセキュリティ体制ダッシュボードを使用して、セキュリティ KPI を監視します。これらのダッシュボードは、インスタンスのセキュリティに関する重要な情報を 1 か所に集約し、ベースシステムのダッシュボードウィジェットがいくつか含まれています。

セキュリティ体制ダッシュボードへのアクセス

セキュリティ体制ダッシュボードにアクセスするには、次に移動して Security Center を開きます: すべて > セキュリティセンター.[セキュリティコンソール] セクションで [セキュリティ体制コンソール] を選択します。[セキュリティ体制コンソール] ページで、上部にある [セキュリティ体制ダッシュボード] を選択します。

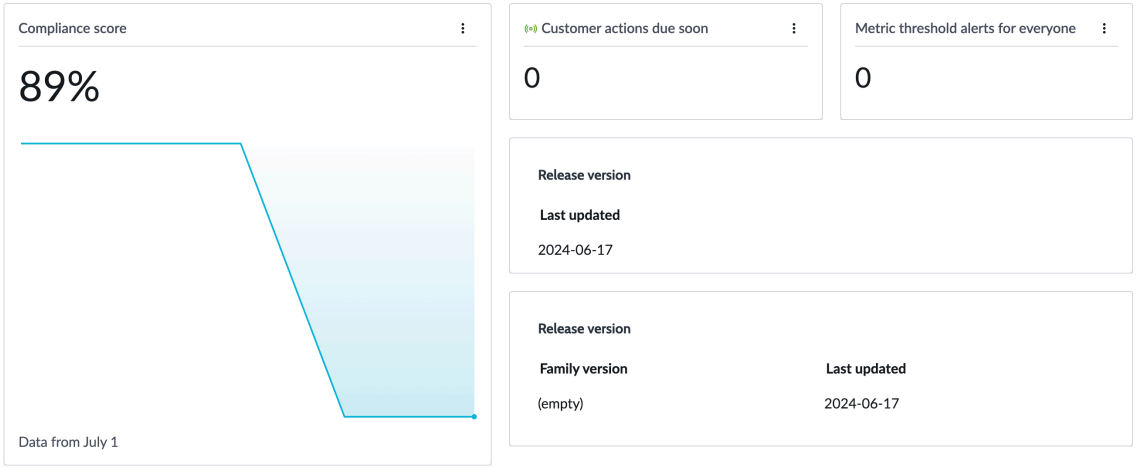
[セキュリティ体制ダッシュボード] の横にある下矢印を使用して、インスタンスダッシュボードを切り替えます。



ダッシュボードは、インスタンスセキュリティの一側面に関連するウィジェットを含む複数のセクションに分かれています。ダッシュボードのウィジェットを選択すると、インスタンスセキュリティのその側面に関する詳細が表示されます。

概要

At a glance



[概要] セクションには、インスタンスのコンプライアンススコア、期限切れのカスタマーアクション、リリース情報など、インスタンスのセキュリティの概要が表示されます。

コンプライアンススコア

ウィジェットの下部に播種された日付から開始して、経時的なインスタンスコンプライアンススコアの割合を表示します。ハードニングコンプライアンススコアのトレンドに移動するには、このウィジェットを選択します。

自分用のメトリクスしきい値アラート

現在のユーザーにアサインされているメトリクスしきい値イベント [n_vsc_metric_threshold_event] レコードの数を表示します。これらのレコードリストに移動するには、このウィジェットを選択します。

全員のメトリクスしきい値アラート。

未アサインのメトリクスしきい値イベント [n_vsc_metric_threshold_event] レコードの数を表示します。これらのレコードリストに移動するには、このウィジェットを選択します。

期限が間近の顧客アクション

期限が近い顧客アクションの数を表示します。顧客アクションに移動するには、このウィジェットを選択します。

アンチウイルスのダウンロードファイル

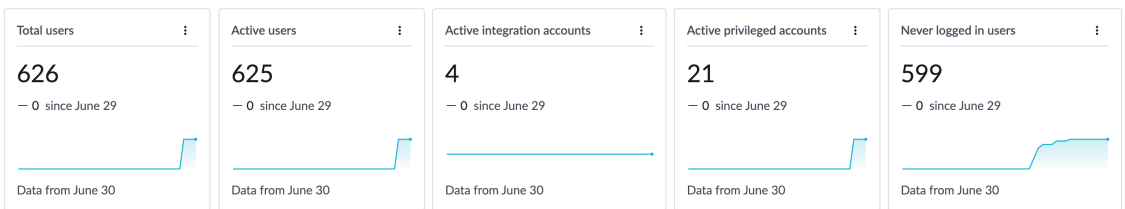
インスタンスにアップロードされた隔離されたファイルの数を表示します。このウィジェットを選択すると、これらの隔離されたファイルがセキュリティ測定基準の [ウィルス対策] セクションに表示されます。

リリースバージョン

インスタンスのファミリーバージョンと前回のインスタンス更新日を表示します。

ユーザー

Users



[ユーザー] セクションには、インスタンスのユーザーに関する情報が表示されます。このセクションのウィジェットには、ユーザー情報と、この情報の経時的な変化を示す線グラフが表示されます。ウィジェットを選択すると、詳細が表示されます。

合計ユーザー

ユーザー [sys_user] テーブルのユーザー数を表示します。このウィジェットを選択すると、セキュリティ測定基準の [アクティブセッション] セクションにこれらのレコードの詳細が表示されます。

アクティブユーザー

ユーザー [sys_user] テーブルのアクティブなユーザーの数を表示します。アクティブユーザーとは、[アクティブ] フィールドが選択されているユーザーレコードのことです。このウィジェットを選択すると、セキュリティ測定基準の [アクティブセッション] セクションにこれらのレコードの詳細が表示されます。

アクティブな統合アカウント

ユーザー [sys_user] テーブルの統合アカウントの数を表示します。統合アカウントは、[**Web** サービスへのアクセスのみ] フィールドが選択されているユーザーレコードです。このウィジェットを選択すると、セキュリティ測定基準の [アクティブセッション] セクションにこれらのレコードの詳細が表示されます。

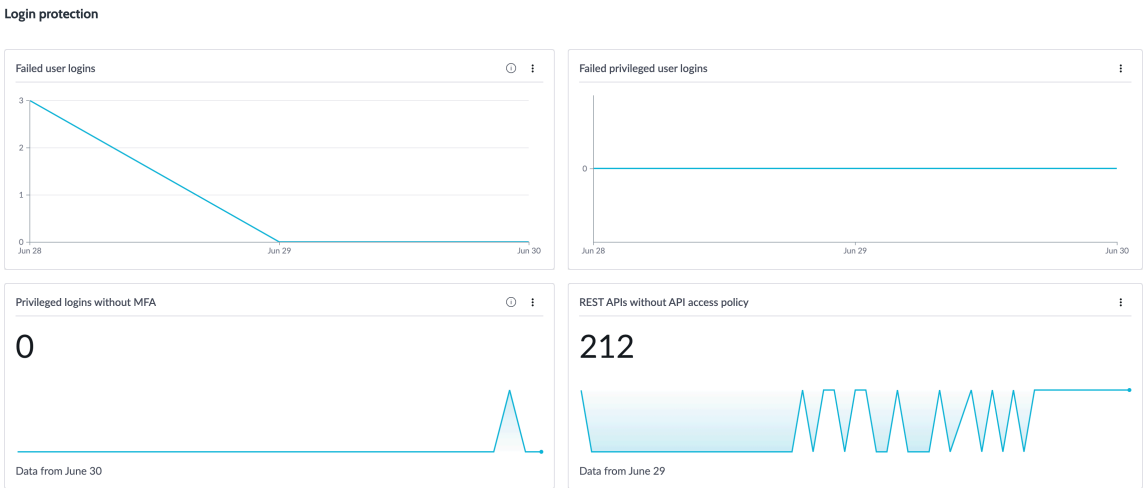
アクティブな特権アカウント

ユーザー [sys_user] テーブルのアクティブな特権ユーザーの数を表示します。特権ユーザーとは、[アクティブ] フィールドが true、[内部統合ユーザー] が false、[内部統合ユーザー] が非アクティブなユーザーレコードのことです。このウィジェットを選択すると、セキュリティ測定基準の [アクティブセッション] セクションにこれらのレコードの詳細が表示されます。

一度もログインしていないユーザー

過去 60 日間に作成され、[前回のログイン時刻] フィールドに値がないユーザー [sys_user] テーブルのユーザーの数を表示します。このウィジェットを選択すると、セキュリティ測定基準の [アクティブセッション] セクションにこれらのレコードの詳細が表示されます。

ログイン保護



[ログイン保護 **(Login protection)**] セクションには、特権ユーザーの失敗したログイン試行を含む、失敗したログインに関する情報が含まれています。これらのウィジェットには、この情報の経時的な変化を示す線グラフが含まれています。ウィジェットを選択すると、詳細が表示されます。

ユーザーログイン失敗

ユーザーログイン試行の失敗回数を表示します。このウィジェットを選択して移動し、セキュリティ測定基準ログイン失敗の詳細を表示します。

特権ユーザーのログイン失敗数

特権アカウントからのユーザーログイン試行の失敗回数を表示します。特権ユーザーとは、[アクティブ] フィールドが true、[内部統合ユーザー] が false、[内部統合ユーザー] が非アクティブなユーザーレコードのことです。このウィジェットを選択して移動し、 セキュリティ測定基準ログイン失敗の詳細を表示します。

MFA のない特権ログイン

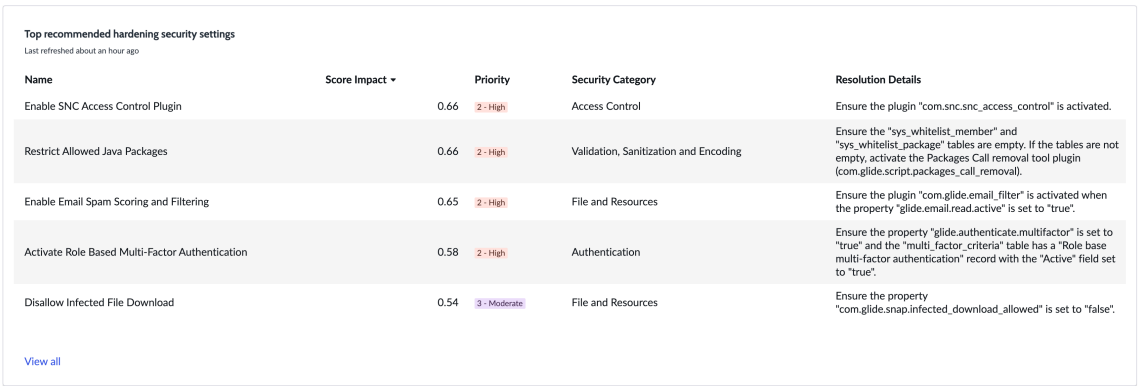
マルチファクター認証 (MFA) 用に構成されていない特権アカウントの数を表示します。このウィジェットを選択して移動し、これらのアカウントのリストを表示しますセキュリティ測定基準。

API アクセスポリシーのない REST API

API アクセスポリシーで制限されていない REST API のリストを表示します。このウィジェットを選択して移動し、 セキュリティ測定基準にあるこれらの REST API のリストを表示します。

インスタンス強化

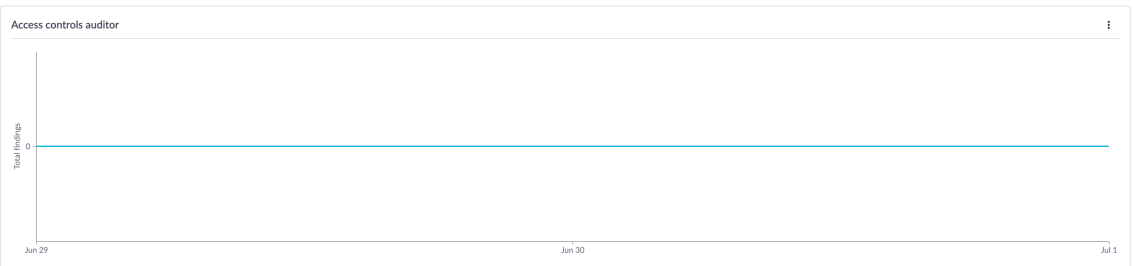
Instance hardening



[インスタンス強化] セクションには、インスタンスのセキュリティを向上させるために変更できる推奨強化セキュリティ設定が含まれています。このセクションを使用して、これらの変更の優先度と潜在的な影響を確認します。

インスタンスの傾向

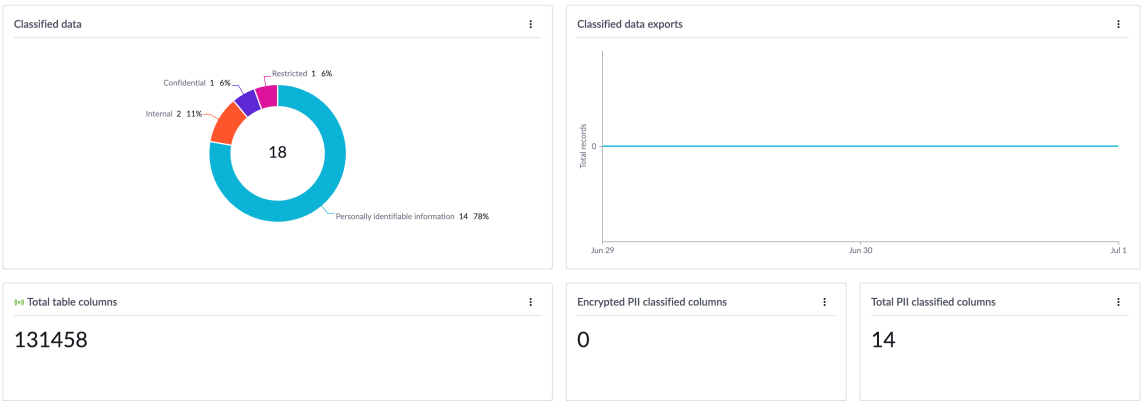
Instance trends



[インスタンストレンド **(Instace trends)]** ダッシュボードには、アクセス制御監査ツールスキャンスイートの結果が表示されます。

データ保護

Data protection



[データ保護] セクションを使用して、個人識別可能情報 (PII) などの分類済みデータの概要を確認します。ダッシュボードは、分類されたデータのエクスポートも追跡します。

分類済みデータ

インスタンスの分類済みデータをタイプ別に分けて円グラフで表示します。チャートのセクションを選択して、これらのレコードの詳細を表示します。データ分類の詳細については、「 データ分類」を参照してください。

分類されたデータエクスポート

インスタンスからエクスポートされた分類済みレコードの数を表示します。このウィジェットを選択すると、分類されたエクスポートのリストがセキュリティ測定基準の [エクスポート] セクションに表示されます。

合計テーブル列

インスタンスのテーブルのすべての列 (フィールド) の数を表示します。このウィジェットを選択すると、セキュリティ測定基準データ分類セクションでこれらの列を確認できます。

暗号化された PII 分類列

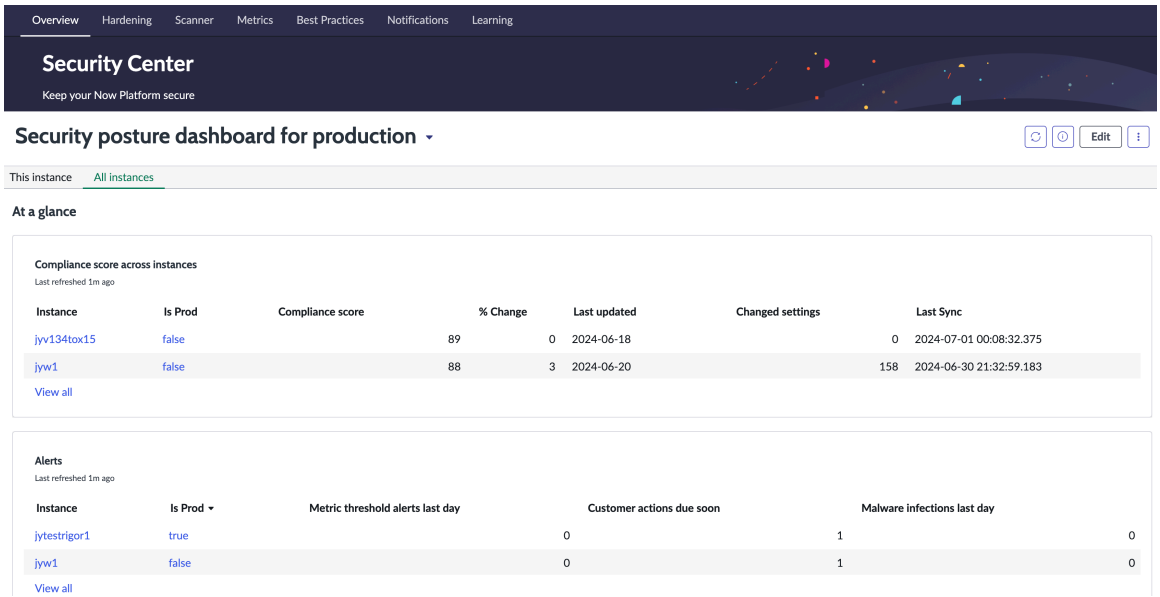
個人識別可能情報 (PII) として分類された暗号化レコードの数を表示します。セキュリティ測定基準のデータ分類セクションでこれらのレコードを確認するには、このウィジェットを選択します。

PII 分類列の合計

個人識別可能情報 (PII) として分類されたすべてのレコードの数を表示します。セキュリティ測定基準のデータ分類セクションでこれらのレコードを確認するには、このウィジェットを選択します。

自動翻訳

複数のインスタンスのレビュー



ダッシュボードの上部にある [すべてのインスタンス **(All instances)**] タブを使用して、本番インスタンスを離れることなく非本番インスタンスのセキュリティ体制を表示できます。[すべてのインスタンス **(All instances)**] タブには、[このインスタンス **(This instance)**] タブと同じ情報の要約版が表示されますが、すべての非本番インスタンスから取得したデータも含まれます。

デフォルトでは、[すべてのインスタンス] タブには、ログインしている本番インスタンスと、すべての本番環境のすべての非本番インスタンスに関する情報が表示されます。

信頼構成を変更することで、このダッシュボードに表示されるインスタンスを追加または削除できます。インスタンス間のデータを可視化すると、ダッシュボード内にインスタンスを表示できます。このプロセスの詳細については、「データ同期アプリケーションの基本信頼構成」を参照してください。

ダッシュボードのカスタマイズ

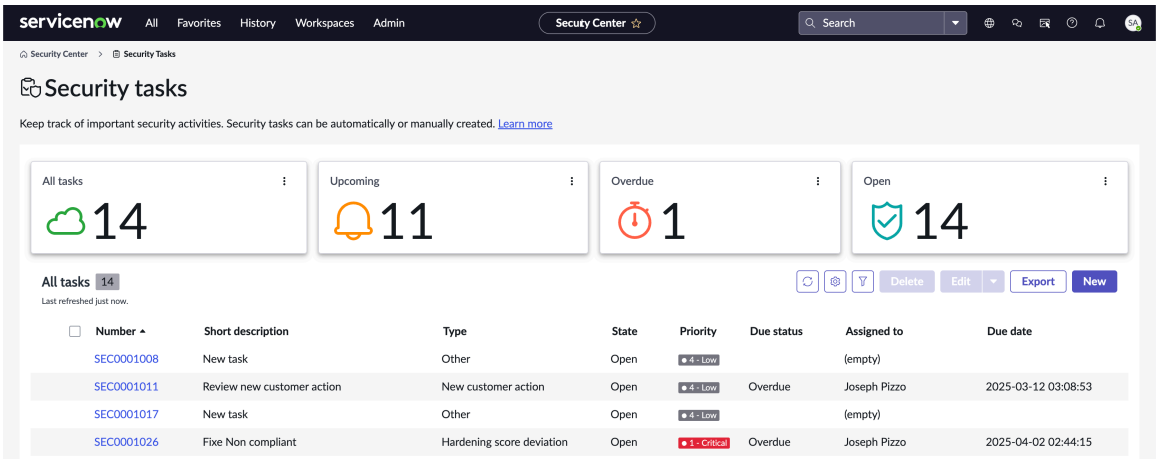
インスタンスセキュリティ体制ダッシュボードはカスタマイズできませんが、[その他のアクション] ([⋮]) アイコンを選択してから、[複製] を選択することで、ダッシュボードを複製できます。複製したダッシュボードを変更できます。

セキュリティタスク

セキュリティタスクを使用して、すべてのセキュリティ関連タスクを 1 か所で監視、優先順位付け、およびアサインします。

セキュリティタスクを使用して、インスタンスのセキュリティ体制を改善および維持するために必要なすべてのタスクを追跡して整理する時間を節約できます。

セキュリティタスクのレビューと管理



次の場所に移動して、セキュリティタスクにアクセスします。 すべて > セキュリティセンター > セキュリティタスク. デフォルトでは、すべてのタスクがリストに表示されます。リストの上にあるカードを選択することで、予定されているタスク、期限切れのタスク、またはオープンタスクを簡単にフィルタリングできます。これらのカードには、これらの各カテゴリの現在のタスク数も表示されます。通常のテーブル フィールド フィルター オプションを使用してフィルターをさらに絞り込み、フィルター ボタン ( ) を選択し、 [詳細ビュー] ボタンを選択して、後で使用するカスタム フィルターを作成して保存します。

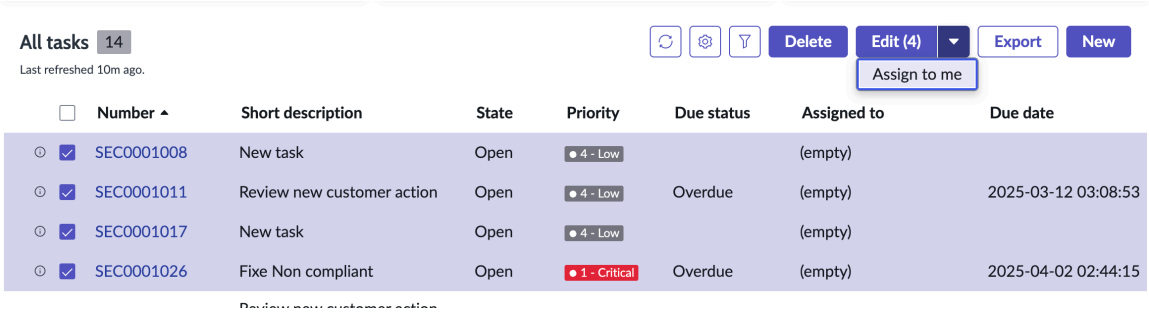
ユーザーロール

セキュリティタスクでは、次のロールを使用します。

ユーザー	必要なロール	説明
システムアドミニストレーター	アドミン	システム管理者は、セキュリティタスクを表示、作成、アサイン、削除できます。
Security Center Viewer (セキュリティセンタービューアー)	sn_vsc_security_center.viewer	閲覧者はセキュリティタスクを表示できますが、作成、削除、編集はできません。
セキュリティタスクマネージャー	sn_vsc_task_manager	セキュリティタスクマネージャーは、セキュリティセンター内で、アサインされたタスクの作業、タスクマネージャー関連のページの表示、タスクの作成と管理を行うことができます。他のユーザーに割り当てられたタスクを表示することはできますが、他のユーザーに割り当てられたタスクを編集することはできません。タスクマネージャーは、セキュリティセンターのタスクマネージャー以外のページにアクセスできません。
ITIL	itil	ITIL ユーザーは、自分にアサインされたセキュリティタスクを操作できますが、アドミンおよびセキュリティタスクマネージャーのロールのようにすべてのタスクを表示することはできません。

セキュリティタスクを自分にすばやくアサイン

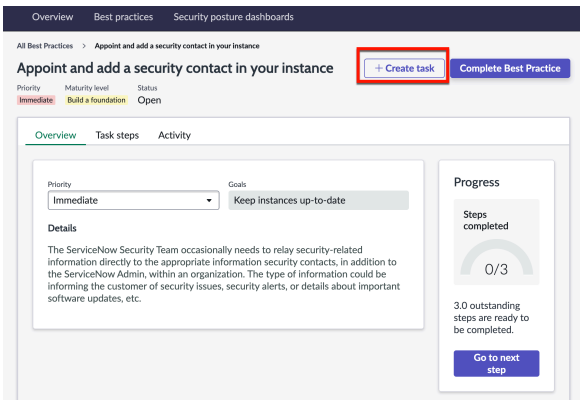
リスト内のアイテムの左側にあるチェックボックスを使用して、複数のアイテムを選択します。複数のタスクが選択されている場合は、[削除] ボタンを使用して選択したタスクを削除したり、[編集] ボタンを使用して選択したタスクを自分にアサインしたりできます。

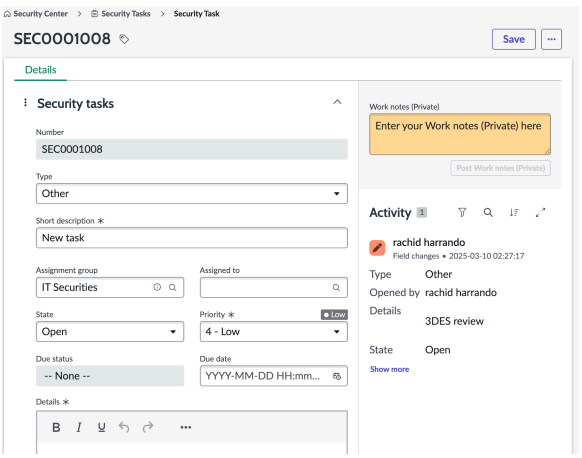


セキュリティタスクを作成

さまざまなプラットフォームセキュリティページにある **[+** タスクを作成] ボタンを使用して、セキュリティタスクを作成します。

たとえば、セキュリティセンターのベストプラクティスツールの詳細ページで、アドミニストレーターは [**+** タスクを作成] ボタンを選択して、このベストプラクティスを完了するためのタスクを作成してアサインできます。





自動的に生成されたタスク

セキュリティタスクは自動的に生成できます。セキュリティタスクの自動生成は、プラットフォームで発生した関連イベントによってトリガーされます。セキュリティタスクの自動生成 で生成されたセキュリティタスクの詳細を確認してください。

セキュリティタスクの編集とアサイン

セキュリティタスクを編集してユーザーにアサインし、期日を定義し、これらのタスクを完了するユーザーに追加の詳細を提供します。詳細については、「セキュリティタスクの編集」を参照してください。

タスクをエクスポート

セキュリティタスクは、選択した形式にエクスポートできます。詳細については、「セキュリティタスクのエクスポート」を参照してください。

セキュリティタスクの自動生成

インスタンスがセキュリティタスクを生成する方法とタイミングについて説明します。

自動翻訳

自動生成されたセキュリティタスク

セキュリティタスクは自動的に生成できます。セキュリティタスクの自動生成は、プラットフォームで発生した関連イベントによってトリガーされます。たとえば、次のようになります。

測定基準のしきい値違反

しきい値に違反すると、メトリクスのセキュリティタスクが生成されます。同じメトリクスに複数の違反がある場合でも、メトリクスに対するオープンタスクは 1 つだけです。違反したメトリクスのセキュリティタスクがクローズされると、しきい値に再び違反したときに新しいタスクが生成されます。

イベント通知

セキュリティタスクは、セキュリティイベント通知がトリガーされたとき (ポリシーの条件が満たされたとき) に生成されます。メトリクスと同様に、1 つのポリシーに対してオープンタスクは 1 つだけで、前のタスクがクローズされている場合は新しいタスクが生成され、通知が再度トリガーされます。セキュリティイベント通知の詳細については、「 セキュリティイベント通知」を参照してください。

ハードニングスコア偏差

ハードニングスコアが設定されたしきい値 (デフォルトは 3) を下回ると、セキュリティタスクが生成されます。たとえば、ハードニングスコアが 97 で、翌日に 94 の場合、スコア 94 が計算された直後にセキュリティタスクが作成されます。オープンタスクは 1 つだけ作成されます。スコアが翌日再び 94 から 91 に低下した場合、プラットフォームは別のタスクを生成しません。ハードニングスコアの詳細については、「 ハードニングコンプライアンススコアのトレンド」を参照してください。

顧客アクション

セキュリティタスクは、顧客アクションがインストールされるたびに生成されます。顧客アクションの詳細については、「 顧客アクション」を参照してください。

バナーのお知らせ

新しいバナーのお知らせごとにセキュリティタスクが生成されます。バナーのお知らせの詳細については、「 セキュリティバナーのお知らせ」を参照してください。

自動的に生成されたセキュリティ設定

自動的に生成されたセキュリティタスクの構成オプションは、セキュリティセンターのプロパティページにあります。これらの設定を見つけるには、次に移動します: すべて > セキュリティセンター > セキュリティセンタープロパティ.

自動セキュリティタスクを有効化/無効化

[はい/いいえ] フィールドを選択すると、インスタンスで自動化されたセキュリティタスクが有効になります。

ハードニングスコアのデグレードしきい値

このフィールドの値は、セキュリティタスクを生成するためにハードニングスコアを (最後の日次スコア以降) 低下させる必要がある量を表します。この値は正の整数である必要があります。デフォルト値は 3 です。

セキュリティタスクの編集

セキュリティセンターでセキュリティタスクを作成、編集、削除、またはエクスポートする方法を学びます

始める前に

必要なロール:admin または sn_vsc_task_manager

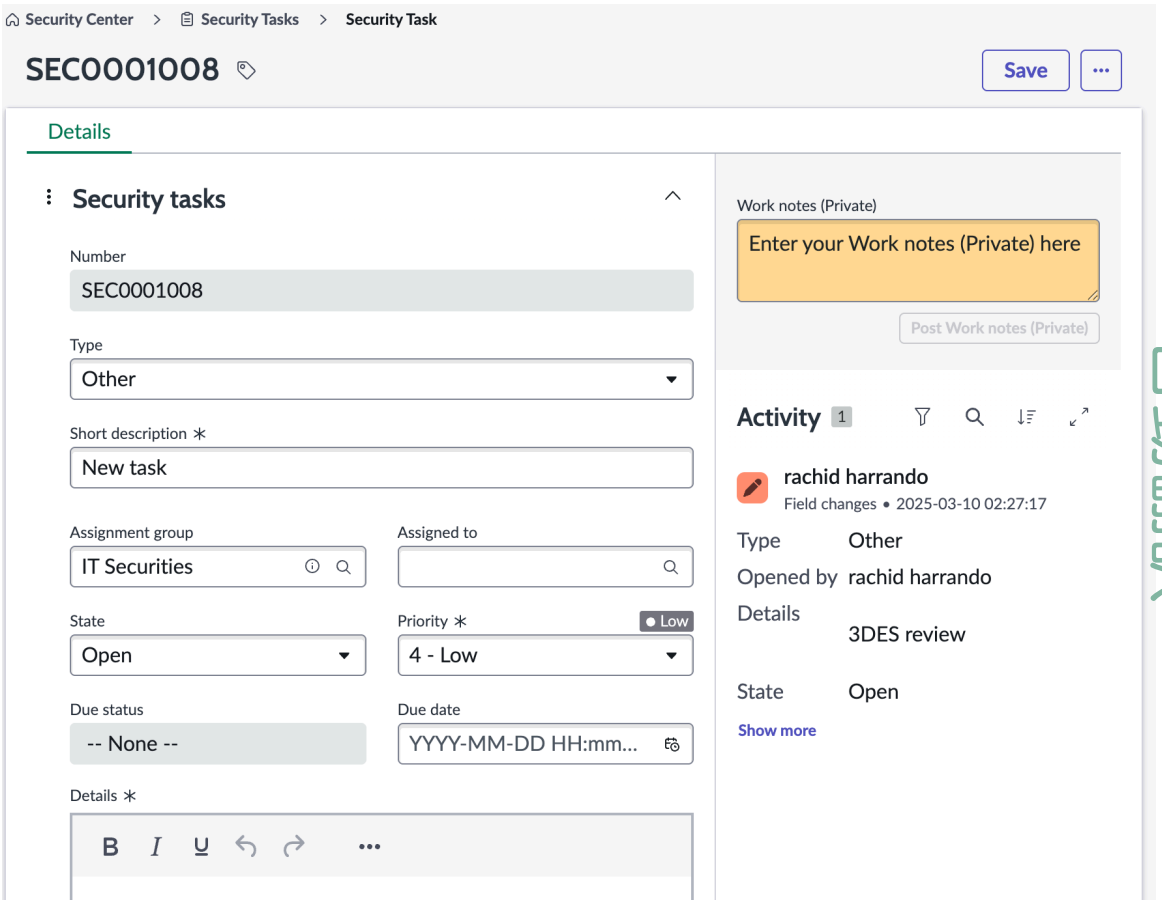
手順

1. 次の場所に移動して、セキュリティタスクリストにアクセスします。 すべて > セキュリティセンター > セキュリティタスク.
2. [セキュリティタスク] リストから、次の 2 つの方法でセキュリティタスクを編集できます。

セキュリティタスクフォームで編集

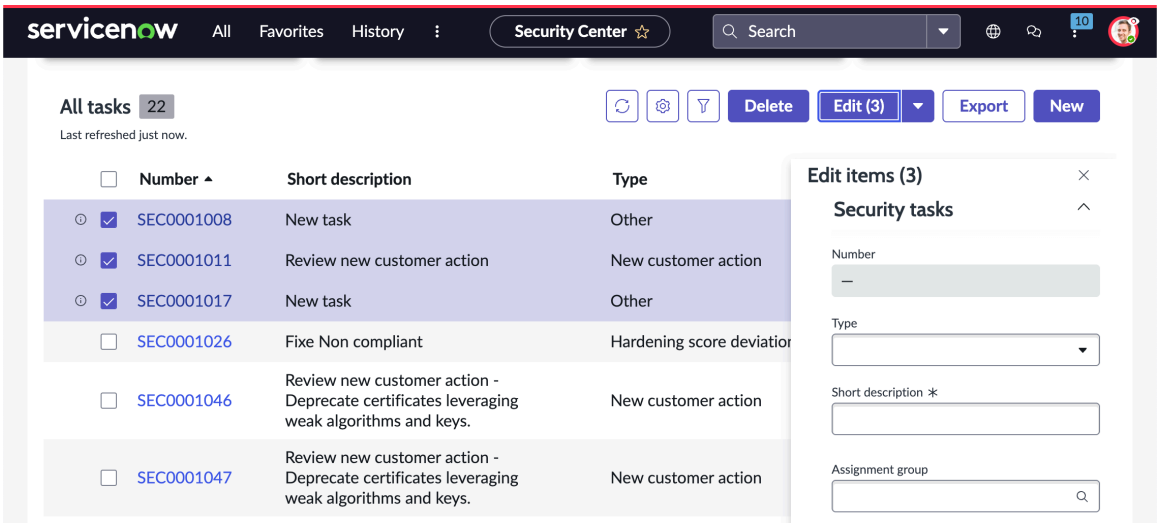
リストからセキュリティタスク番号を選択してセキュリティタスクレコードを開き、その詳細を表示します。ここでは、編集、別のユーザーへのアサイン、作業メモの追加を行うことができます。

重要: タスクがアサインされるユーザーは、アドミニストレーターであるか、sn_vsc_task_manager ロールを持っている必要があります。



セキュリティタスクリストから編集

左側のチェックボックスをオンにして 1 つ以上のセキュリティタスクを選択し、[編集] ボタンを選択します。



3. 必要に応じてフィールドに入力します。

フィールド	説明
番号	タスクの識別に使用する自動生成番号
タイプ	タスクタイプを選択: ○ 測定基準のしきい値違反 ○ 新規顧客アクション ○ イベント通知 ○ ハードニングスコア偏差 ○ 新しいバナーのアナウンスメント
簡単な説明	タスクの短い説明
アサイン先グループ	タスクのアサイン先グループ
アサイン先	タスクにアサインされた人。このユーザーは、[アサイン先グループ] フィールドで選択したアサイン先グループに属し、 *admin* または *sn_vsc_task_manager* ロールを持っている必要があります。
状況	タスクのステータス: ○ 保留中 ○ 開く ○ 進行中 ○ クローズ ○ キャンセル
優先度	タスクの優先度: ○ 1- 重大 ○ 2- High ○ 3 - 中 ○ 4 - 低

フィールド	説明
期限ステータス	タスクが [予定] か [期限切れ] か ([期日] フィールドの日付による)。
期日	タスクを完了させる必要のある日時。
詳細	このタスクの詳細、リンク、またはユーザーが知る必要のあるその他の情報を提供するために使用されます。
作業メモ	このタスクに関するメモ。このフィールドに入力されたメモにはタイムスタンプが付けられ、画面右側の [アクティビティ] リストに表示されます。
作業メモ (プライベート)	作業メモを非公開で追加するには、このフィールドにメモを入力し、[作業メモの投稿 (プライベート)] ボタンを選択します。

4. [保存] を選択してセキュリティタスクレコードを保存します。リストからレコードを編集する場合は [更新] を選択します。
5. オプション: 必要に応じて、右側のチェックボックスをオンにして [削除] を選択するか、フォームで [その他] アイコン (...) を選択して [削除] を選択することで、リストからタスクを削除できます。

セキュリティタスクのエクスポート

セキュリティタスクをファイルにエクスポートし、ダウンロードして他のソフトウェアで使用できる方法を学びます。

始める前に
必要なロール：admin

手順

1. 次の場所に移動して、セキュリティタスクリストにアクセスします。 すべて > セキュリティセンター > セキュリティタスク.
2. 左側のチェックボックスをオンにして、1 つ以上のセキュリティタスクを選択します
3. [エクスポート] ボタンを選択します。
4. [ファイルタイプ] で、セキュリティタスクのエクスポート先のファイル形式を選択します。Excel、CSV、JSON、または PDF 形式から選択できます。
5. **[Delivery Type]** で、[**Download**] または **[Email]** を選択します。

   ダウンロード

   エクスポートファイルはブラウザを使用してダウンロードされます。

   メール

   [メール] フィールドが表示されます。[メール] フィールドにメールアドレスを入力すると、エクスポートファイルが指定されたメールアドレスに配信されます。

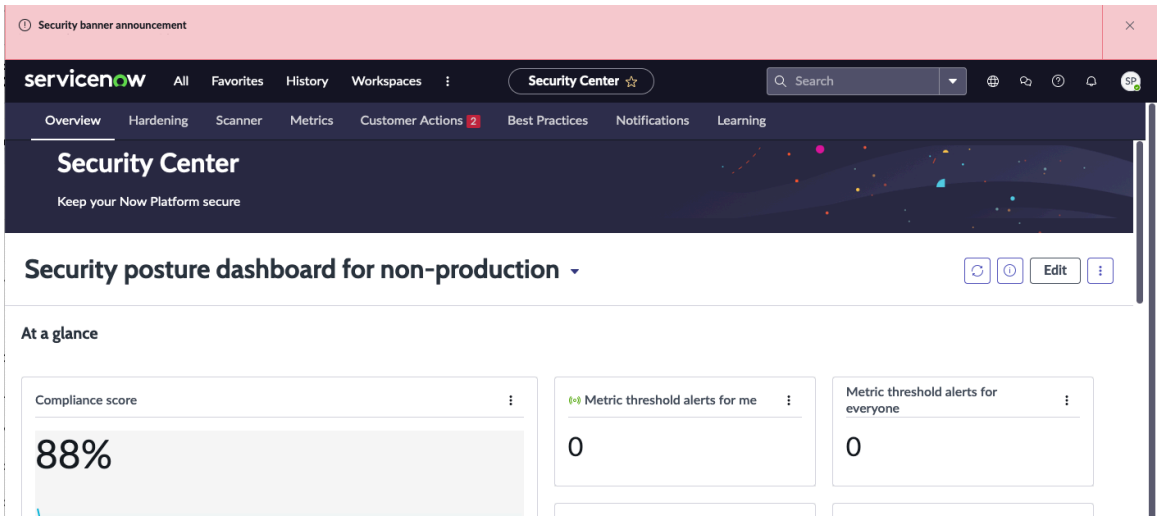
セキュリティ学習

単一のページからセキュリティ学習教材にアクセスします。

[ラーニング] に移動して、セキュリティホワイトペーパー、電子書籍、ナレッジベース (KB) 記事、製品ドキュメント、およびコミュニティディスカッションを統合ビューから参照します。コンテンツは、見出しと内容を端的に表す UI カード別に整理されているため、正しいリソースをすばやく特定できます。

セキュリティバナーのお知らせ

インスタンス UI 内でアドミニストレーターに表示される視認性の高いバナーを使用して、セキュリティバナーのお知らせを有効にし、緊急および重大なセキュリティアラートに関する最新情報を常に把握できるようにします。

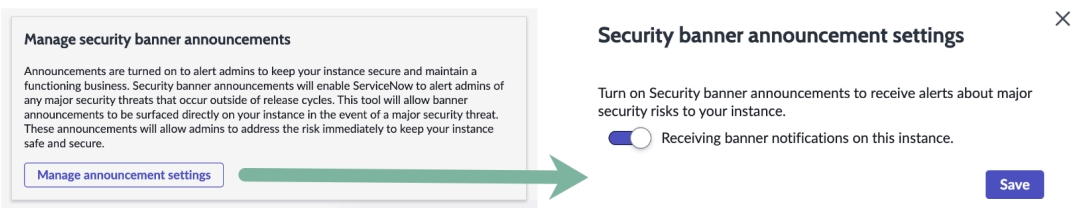


セキュリティバナーのお知らせは、最近発見された潜在的なセキュリティの脅威に対する修正についての最新情報をお届けするために、ServiceNow から送信され、顧客アドミニストレーターに表示されるお知らせです。これらのアラートにはセキュリティリスクの概要とリンクが含まれ、リンク先で詳細を確認してインスタンスをセキュリティ保護するためのアクションを実行できます。

アドミニストレーターは [閉じる] (x) ボタンを選択してバナーを閉じることができますが、バナーの有効期限が過ぎるまで、バナーは新しいセッションごとに再表示されます。アドミニストレーターは、sn_vsc.configure_customer_push_action システムプロパティ値を false に設定することで、バナーのお知らせを無効にすることができます。

セキュリティバナーのお知らせを有効または無効にする

セキュリティバナー機能はデフォルトで有効になっています。セキュリティバナーのお知らせを有効または無効にするには、次の場所に移動します システムセキュリティ > セキュリティセンター > 通知. このページから、[お知らせ設定の管理 **(Manage announcement settings)]** ボタンを選択します。



インスタンスセキュリティセンター

インスタンスセキュリティコントロールのコンプライアンスレベルの監視、セキュリティイベントモニタリングメトリクスの表示、インスタンスセキュリティ設定の構成と維持はすべて、インスタンスセキュリティセンター内から行います。インスタンスセキュリティセンターは、複数の主要なセキュリティコンポーネントを単一のコントロールコンソールに統合し、インスタンスベースのセキュリティイベントの検出、保護、応答を支援します。

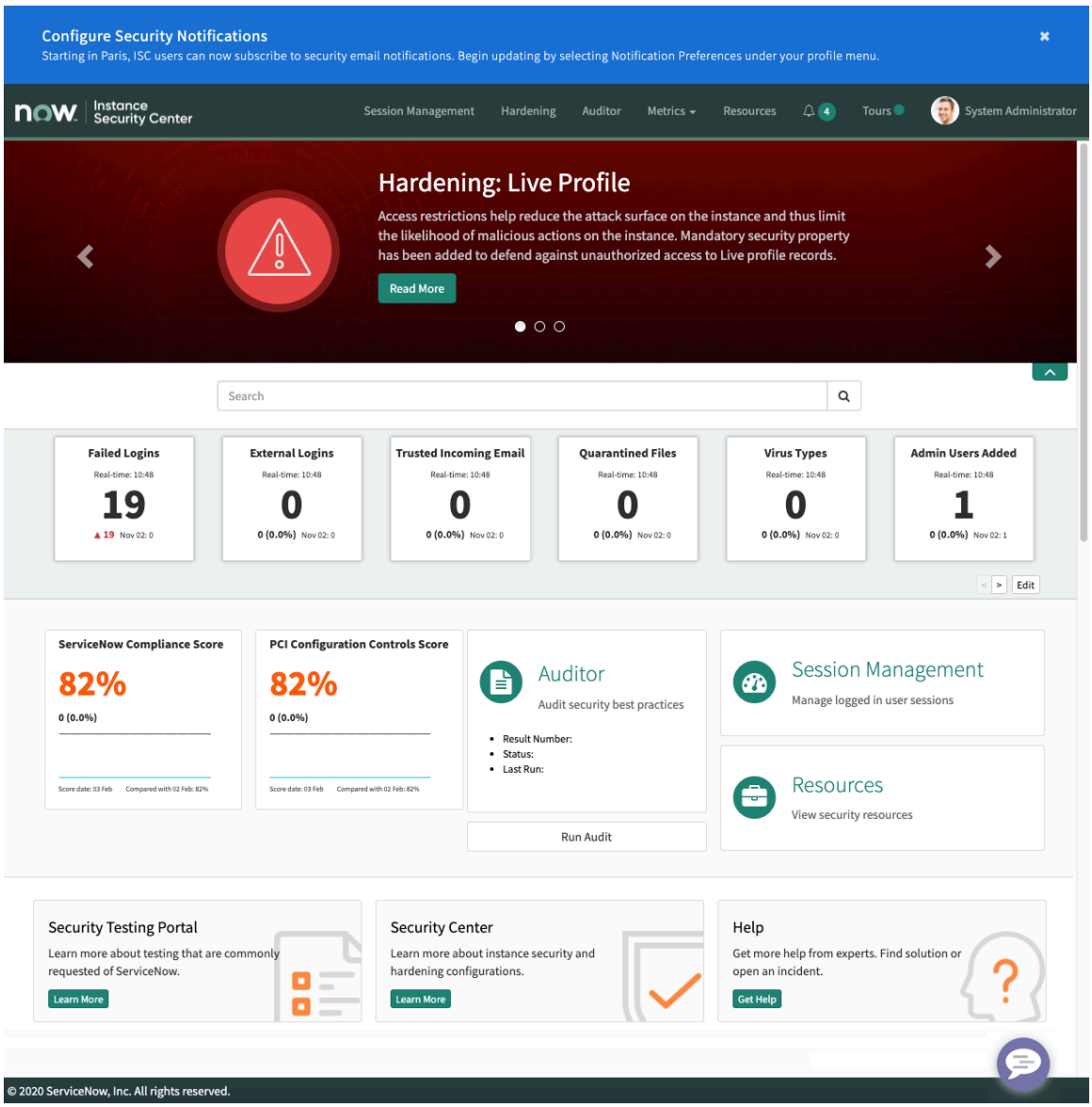
重要:

インスタンスセキュリティセンター (ISC) は 2024 年 9 月の時点で販売が終了しており、サポートもされていないため、新たにアクティブ化することもできません。

ServiceNow セキュリティセンター (SSC) が、今後推奨されるソリューションです。詳細については、「インスタンスセキュリティセンターから ServiceNow セキュリティセンターへの移行」を参照してください。

インスタンスセキュリティセンターのコンポーネント

Instance Security Center にアクセスするには、次に移動します: システムセキュリティ > インスタンスセキュリティセンター またはシステム管理ホームページ。



インスタンスセキュリティセンターのホームページには、次のようなセキュリティコンポーネントが含まれています。

- アドミニストレーターメッセージ
- ローテーションセキュリティバナー
- 検索
- セキュリティイベントリボン

- 日次コンプライアンススコア
- PCI 構成コントロールスコア
- セッション管理
- ハードニング
- 監査ツール
- メトリクス (ユーザー、メール、ウイルス対策)
- リソース
- セキュリティ通知
- ツアー
- セキュリティテストポータル
- セキュリティセンター
- ヘルプ
- 仮想エージェントアクセス

インスタンスセキュリティセンターのホームページから、インスタンスのセキュリティコンプライアンススコアを表示し、全体的なセキュリティ健全性を監視できます。その後、インスタンスのセキュリティに関連するシステムプロパティを構成または更新して、セキュリティ要件に準拠させることができます。

注: インスタンスセキュリティセンターはドメインセパレーションをサポートしていません。

ユーザーロール

インスタンスセキュリティセンターを使用するには、admin または security_dashboard_user ロールが必要です。

ユーザーごとのサブスクリプション管理の詳細については、「サブスクリプション管理 でのユーザーごとのサブスクリプションの管理」を参照し、アカウント担当者にお問い合わせください。

ロール

必要なロール	ユーザー	メリット
アドミン	アドミニストレーターはアクセス制御リスト (ACL) ルールをオーバーライドし、すべてのロールチェックに合格できるため、このロールはすべてのシステム機能性、機能、データにアクセスできます。より対象が絞られたロールを使用できる場合は、このロールをユーザーに割り当てないでください。	セキュリティセンターツールを活用してインスタンスのセキュリティ体制を改善し、セキュリティ関連の動作を監視します。
sn_vsc.security_center_viewer	このロールでは、アドミン以外のユーザーがセキュリティセンターの情報を表示できますが、セキュリティセンターツールを変更したり、セキュリティセンターツールを使用してイン	インスタンスのセキュリティ体制を監視し、セキュリティ動作を監視するため、セキュリティセンターツールを可視化します。

ロール (続く)

必要なロール	ユーザー	メリット
	スタンス構成を変更したりすることはできません。 たとえば、プラットフォームオーナー、セキュリティ運用アナリスト、またはコンプライアンスのステークホルダーが、セキュリティセンターで利用可能なセキュリティ KPI、セキュリティインサイト、およびセキュリティ学習教材の一部を表示する必要がある場合があります。	

警告: アップグレードのたびにインスタンスセキュリティセンターが最新のセキュリティ情報を受信するようにするには、このモジュールをカスタマイズしないでください。インスタンスのセキュリティ設定を変更する場合は、まず非本番環境でテストしてください。

アドミニストレーターメッセージ

主にアドミン向けのメッセージとリマインダーが、ローテーションするセキュリティバナーの上に表示されます。

たとえば、[セキュリティ通知の設定 (Configure Security Notifications)] というメッセージが表示され、アドミニストレーターがセキュリティ通知の設定を行っていない場合は設定するように促されます。また、そのための適切なページを示します。

注: アドミニストレーターメッセージバナーは、admin 以外のユーザーの場合、または admin ユーザーにアクション可能なアイテムがない場合は表示されません。

ローテーションセキュリティバナー

インスタンスのセキュリティ健全性の監視を支援するために、重要なインスタンスのセキュリティメッセージがローテーションバナーに表示されます。

- 通常、2 ～ 3 件のセキュリティメッセージが定期的にローテーションされます。
- バナーの下部にあるドットは、現在のセキュリティメッセージの合計数を示しています。
- メッセージ間を移動するには、ドットを選択するか、メッセージの両側に表示される矢印を選択します。

バナーの背景色は、メッセージの相対的な重大度を示します。

色	説明
赤	タイムリーな対応、または重大なセキュリティイベントの保護または対応方法に関する推奨事項が必要な、重大なセキュリティ状況
ダークグレー	重大でない警告メッセージ
青	一般情報メッセージ。

バナーのテキストコンテンツを折りたたむ (最小化する) には、 を選択します。テキストコンテンツを最大化するには、もう一度選択します。

- インスタンスセキュリティセンターを再度使用すると、テキストコンテンツは前のセッションでの使用方法に応じて、折りたたまれるか展開されて表示されます。
- テキストコンテンツ自体が変更されると、すべてのユーザーに対して最大化されて表示されます。

検索

検索バーを使用して、インスタンスセキュリティセンター全体を対象に、セキュリティ問題の理解と解決に役立つセキュリティリソースを検索します。次のセキュリティ関連リソースを検索できます。

- Now Support ナレッジベース の記事
- インスタンスセキュリティセンターのページ
- 外部 Now Support リンク
- 日次コンプライアンススコアや外部受信メールなど、PA セキュリティウィジェット
- バナーコンテンツ

イベントリボン

イベントリボンを使用して、現在のインスタンスの主要なセキュリティイベントモニタリングメトリクスを表示します。

- メトリクスを手動でスクロールするには、左右の矢印キーを選択します。
- イベントリボンを設定するには、[編集] を選択します。

イベントリボンとその設定方法については、「セキュリティイベントの監視」および「セキュリティイベントリボンの構成」を参照してください。

日次コンプライアンススコア

[日次コンプライアンススコア] セクションには、[日次コンプライアンススコア]、[セッション管理]、[ハードニング]、[監査ツール **(Auditor)]**、および [リソース] というタイルが含まれています。

日次コンプライアンススコアを使用して、セキュリティの観点からインスタンスの健全性を評価します。

日次コンプライアンススコアは、パーセンテージスコアです。これは、インスタンスのセキュリティプロパティの現在の設定が、ハードニング設定 で公開されているコンプライアンス値にどの程度準拠しているかに基づいています。

- 日次コンプライアンススコアの計算と強化設定の影響の詳細については、「日次コンプライアンススコアの確認とセキュリティプロパティの設定」を参照してください。
- [更新] ボタンを使用すると、アドミニストレーターは日次コンプライアンススコアをただちに再計算できます。詳細については、「日次コンプライアンススコア、傾向、およびグラフデータの更新方法 」を参照してください。

ハードニング

このプロセスを使用して、日次コンプライアンススコアに影響する特定のセキュリティ設定プロパティを調整します。

1. [コンプライアンス構成の強化 (Hardening Compliance Configurations)] ページにアクセスしてインスタンスセキュリティの強化を実行するには、[日次コンプライアンススコア] タイルまたは [ハードニング] リンクを選択します。
2. すべてのセキュリティコントロールを表示するか、推奨されるセキュリティコントロールのみを表示するかを指定します。次に、作業するカテゴリを選択します。
3. 選択したカテゴリで、セキュリティ構成プロパティをそれぞれ設定します。[その他の情報] をクリックすると、プロパティの詳細情報が表示されます。

コンプライアンスをさらに強化するためのセキュリティ構成プロパティの強化と最適化の詳細については、「 インスタンスのセキュリティ設定調整によるコンプライアンスの強化」を参照してください。

傾向データとグラフデータの更新方法の詳細については、「日次コンプライアンススコア、傾向、およびグラフデータの更新方法」を参照してください。

監査ツール

監査ツールを実行してインスタンスをスキャンし、誤ったセキュリティ定義を見つけます。 インスタンスのセキュリティ体制を改善するために修正できる結果を提供します。

[監査人] ページにアクセスするには、[監査人] タイルまたは [監査人] リンクを選択します。詳細については、「誤ったセキュリティ定義のスキャン」を参照してください。

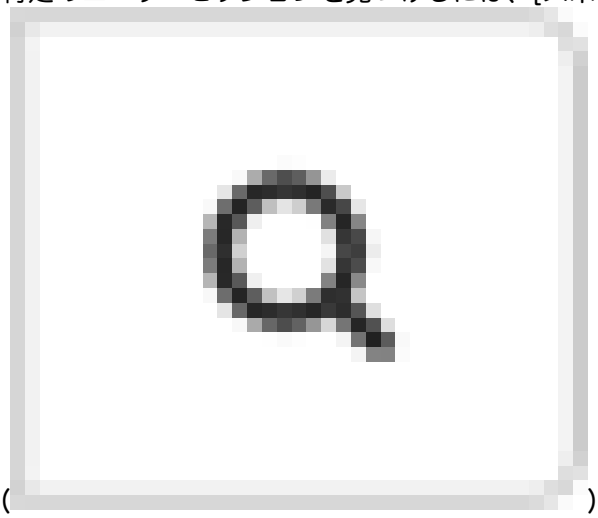
セッション管理

セッション管理を使用して次のことを行います。

- ユーザーログインセッションを表示および管理します。
- 接続している現在のノードのユーザーログインセッションを確認します。
- ユーザー名や IP アドレスなど、各セッションの詳細情報を表示します。
- セキュリティ上のリスクを引き起こす特定のユーザーセッションを隔離してロックアウトします。

[セッション管理] ページにアクセスするには、[セッション管理] タイルまたはリンクを選択します。

フィールド	説明
ユーザー	このログインセッションに関連付けられたユーザーの名前。

フィールド	説明
	• 特定のユーザーセッションを見つけるには、[スポットライト検索] アイコン  ( ) を選択して、ユーザー、ユーザーエージェントキーワード、または IP アドレスで検索します。 たとえば、特定タイプのブラウザからの現在のログインをすべて検索する場合は、ブラウザ名をキーワードとして [ユーザーエージェント] フィールドに入力します。 • ユーザー名をクリックして、ユーザープロファイルレコードにアクセスします。ユーザープロファイルを変更できるのは、admin ロールが割り当てられている場合のみです。 **注:** ユーザープロファイルの詳細については、「ユーザーの作成」を参照してください。
MFA	ログインしているユーザーに対してマルチファクター認証 (MFA) が有効になっているかどうかを示すチェックボックス。MFA の詳細については、「 多要素認証」を参照してください。
Active (アクティブ)	ログインしているユーザーがアクティブか非アクティブかを示すチェックボックス
ユーザーエージェント	ユーザーログインセッションのブラウザのタイプとデバイスのオペレーティングシステム
IP アドレス	ログインしているユーザーの IP アドレス。
前回アクセス	このユーザーセッションが最後にインスタンスにアクセスした日時。 **注:** 特定のログインセッションの詳細情報を表示するか、またはセッション自体をロックアウトするには、[ユーザーエージェント]、**[IP** アドレス]、または [前回アクセス] フィールドを選択します。

メトリクス

次のタイプのメトリクスの詳細を表示します。

メトリクスのタイプ	説明
ユーザー	インスタンス内のユーザーアクティビティに関連付けられているセキュリティメトリクス。 [ユーザーメトリクス] ページにアクセスするには、[メトリクス] リンクを選択し、[ユーザーメトリクス] を選択します。
エクスポート	インスタンスからエクスポートされるデータに関連付けられているセキュリティメトリクス。 [メトリクスをエクスポート] ページにアクセスするには、[メトリクス] リンクを選択し、[メトリクスをエクスポート] を選択します。
認証	使用頻度の低い IP アドレス、失敗したログイン、ユーザーが使用する認証スキームのタイプなど、認証に関連するセキュリティメトリクス。 [メトリクスをエクスポート] ページにアクセスするには、[メトリクス] リンクを選択し、[認証メトリクス] を選択します。
メール	インスタンスの受信メールに関連する例外動作に関連付けられたセキュリティメトリクス。 [メールメトリクス (Email Metrics)] ページにアクセスするには、[メトリクス] リンクを選択し、[メールメトリクス **(Email Metrics)]** を選択します。
ウィルス対策	インスタンス内のウイルス対策イベントアクティビティに関連付けられているセキュリティメトリクス。 [ウイルス対策メトリクス (Antivirus Metrics)] ページにアクセスするには、[ウイルス対策] タイルを選択するか、[メトリクス] リンクを選択して、[ウイルス対策] を選択します。

注: 各タイプのメトリクスの監視の詳細については、「インスタンスメトリクスの監視」を参照してください。

リソース

インスタンスセキュリティに関連する Now Support ナレッジベース 記事、リソース、およびブログにアクセスします。これらのリソースには、セキュリティ設定、コーディング、コンプライアンス、修正、および関連トピックが含まれます。[リソース] ページにアクセスするには、以下の手順を実行します。

1. [リソース] タイルまたはリンクをクリックします。
2. [リソース] ページで、カテゴリを選択します。

カテゴリ	説明
推奨ガイドライン	ハードニング設定 およびセキュアコーディングガイド [KB0623354] に関する記事など、推奨されるセキュリティガイドラインへのアクセス
セキュリティリソース	ナレッジベース内の、以下のようなセキュリティ関連リソースへのアクセス

カテゴリ	説明
	○ カスタマーインスタンスのセキュリティテスト ○ クラウドセキュリティ、信頼、およびコンプライアンスセンターに関するナレッジベース記事

セキュリティ通知

インスタンスセキュリティセンターに、通知ベルアイコン () が未読のセキュリティ通知の合計数とともに表示されます。通知は、既読としてマークするまで保持され、未読としてカウントされます。

1. ベルアイコンをクリックすると、未読のセキュリティ通知の最初の 5 件が表示されます。

   admin ログイン、**admin** ロック解除 **(Admin Unlocked)**、失敗したログイン、高権限ロール **(High Privilege Role)**、代理操作、セキュリティ昇格、および週次ダイジェスト **(Weekly Digest)** のイベントがインスタンスで発生すると、通知が表示されます。これらのセキュリティイベントの詳細については、「セキュリティイベントの監視」を参照してください。

2. 特定のセキュリティイベントの詳細情報を表示するには、[通知] を選択します。

   たとえば、高権限ロールの通知をクリックすると、ロール (sys_user_role) テーブルを表示できます。このテーブルを使用して、暦日に特権ロールが割り当てられたユーザーを確認します。この履歴を使用すると、ロールが適切に割り当てられているかどうかを判断できます。

3. 未読の通知が 5 件より多い場合は、[すべての通知を表示] を選択すると、すべての未読通知が一覧表示される [すべての通知] ページにアクセスできます。
   - 特定のセキュリティイベントの詳細情報を表示するには、[通知] を選択します。
   - リストされているすべての通知を既読にするには、[すべて既読としてマーク] を選択します。

> **注:** アドミニストレーターは、セキュリティイベントのタイプごとに特定のタイプの通知を送信するように設定することもできます。詳細については、「セキュリティイベント通知の設定 」を参照してください。

ツアー

[ツアー] リンクをクリックして、インスタンスセキュリティセンターのビジュアルガイドツアーを表示します。

- ガイド付きツアーには、 ホームページにリストされているセキュリティモニタリング機能のみが含まれています。
- ホームページのタイルまたはリンクを選択したときにアクセスするセキュリティ機能は含まれません。

セキュリティテストポータル、セキュリティセンター、およびヘルプ

Now Support サービスポータル は、インスタンス、タスク、およびアカウントを管理するために使用する中心的なリソースです。インスタンスのセキュリティおよび技術的な問題を診断して解決するために利用できる有用なリソースにもアクセスできます。

これらのリソースにアクセスするには、次のタイルで [詳細] または [ヘルプ] を選択します。

タイル	説明
セキュリティテストポータル	Now Support セキュリティテストポータルのセキュリティダッシュボードにアクセスします。
セキュリティセンター	Now Support セキュリティポータルのセキュリティコンプライアンスにアクセスします。
ヘルプ	Now Support セキュリティポータルの、次のヘルプリソースにアクセスします。 • エキスパートに一般的な質問に対する回答を見つけてもらうことができます。 • ケースをオープンして、ServiceNow グローバルテクニカルサポートに問題または機能停止を報告します。 • 以下ののようなセルフサービスサポートリソース ◦ ビデオ ◦ ドキュメント ◦ ServiceNow コミュニティ ◦ ナレッジベース ◦ 既知のエラーポータル ◦ セキュリティ RFX データベース • 使用が推奨される ServiceNow コミュニティ の質問

仮想エージェントアクセス

仮想エージェント は、メッセージングインターフェイス内の会話を通じてユーザーサポートを提供するためのプラットフォームです。

関連プラグインがインストールされている場合、アドミニストレーターは [仮想エージェント] アイコンをクリックして仮想エージェントおよび 自然言語理解 (NLU) (NLU) の機能にアクセスできます。

次のタスクを実行できます。

- セキュリティ関連の質問をして、簡単な要約回答と詳細への参照リンクを取得する
- 次のようなプロセスに関連する回答を取得する
  - インスタンスセキュリティセンター
  - プラットフォームセキュリティ
  - ServiceNow セキュリティポリシー
  - 信頼、ガバナンス、リスク
- Now Support Security ポータルでナレッジベースのトピックなどのセキュリティ関連リソースを検索する

注: 仮想エージェントを使用およびアクティブ化する方法の詳細については、以下を参照してください。

- 仮想エージェント
- ISC 仮想エージェントインターフェイスのアクティブ化

インスタンスセキュリティセンターから **ServiceNow** セキュリティセンターへの移行

インスタンスセキュリティセンター (ISC) から ServiceNow セキュリティセンター (SSC) に移行する際の主な違いについて説明します。

重要:

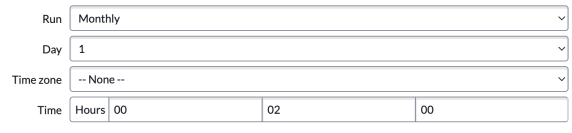
インスタンスセキュリティセンターは、2024 年 9 月に販売終了予定のレガシー製品です。ServiceNow セキュリティセンターが、今後お客様に採用を推奨するソリューションです。詳細については、Now Support ナレッジベースの Deprecation Process (廃止プロセス) の記事 (KB0867184) を参照してください。

ServiceNow セキュリティセンター (SSC) は、組織が ServiceNow 展開のセキュリティを管理するのを支援するように設計された一連の専用ツールで構成されるセキュリティアプリケーションです。SSC を使用すると、組織はシームレスなユーザーエクスペリエンスでセキュリティ体制を改善し、コンプライアンスレベルを強化できます。

このドキュメントでは、SSC で強化された機能と、以前に ISC で実行していたタスクを SSC で実行する方法について説明します。

セキュリティハードニング

機能拡張	ISC と SSC
ユーザーインタラクションの更新	• 概要ページまたはハードニング比較ページに移動して最新のスコアを確認します。 • 概要ページに移動して [スコアを更新] を選択し、最新のスコアを手動で更新します。 • 設定構成を変更するには、次のいずれかのページで設定名のリンクを選択し、[設定の詳細 **(Settings detail)**] ページで編集します。 ○ 概要 ○ ハードニング > すべての設定 ○ ハードニング > 比較
コンプライアンススコアを再計算するスケジュールの設定	SSC を使用すると、システムがコンプライアンススコアの更新をトリガーするタイミングをスケジュールできます。 **1.** [スクリプトの実行を予定] [sysauto_script] テーブルに移動します。 **2.** [SC - コンプライアンスを毎月計算 (SC - Calculate Compliance Monthly)] レコードを探して開きます。

機能拡張	ISC と SSC
	3. [実行]、[日]、[タイムゾーン]、[時間] の各フィールドを使用して、スケジュールを設定します。  **4.** [更新] を選択して、変更を保存します。

セキュリティ **KPI** と測定基準

機能拡張	ISC と SSC
KPI の機能拡張	SSC を使用すると、次のことができます。 • 経時的なスコアの傾向を確認します。 • 同じインターフェイスと分析機能を使用して、すべての KPI と測定基準を監視します。
測定基準の機能拡張	SSC では、次のメトリクスが強化されています。 • 65 を超える測定基準が追加されています。 • 単一のユーザーインターフェイスから KPI と測定基準を監視および分析できます。 • 測定基準のターゲットとしきい値を作成できます。
ユーザーインタラクションの更新	SSC のメトリクスにアクセスするには、次の場所に移動します。 概要 > メトリクスをクリックし、ナビゲーションメニューから [メトリクス] を選択します。または、[すべての測定基準 **(All metrics)]** に移動することもできます。

セキュリティスキャナー

機能拡張	ISC と SSC
新機能	SSC には次の機能が含まれています。 • スキャンを手動で実行するか、特定の時間に実行するようにスケジュールします。 • 独自のスキャンチェックを作成します。 • 独自のスキャンスイートを作成します。 • 2 つのスキャンの結果を比較します。
拡張されたメール機能	インスタンスのセキュリティイベントに関する最新情報を常に入手できるようにメール通知を設定します。

機能拡張	ISC と SSC
ユーザーインタラクションの更新	• 次の場所に移動して監査ツールスキャンを実行します 概要 > スキャン > スイート をクリックし、リストから [監査ツールスイート] を選択します。 • スキャン結果を確認するには、次に移動します: 概要 > スキャン > 結果.

その他のセキュリティ機能

機能拡張	ISC と SSC
セッションの監視機能	• アナリティクスハブ を使用してセッションアクティビティを可視化できます。 • に移動してセッションアクティビティを監視する メトリクス > セッション.
リソースを検索	次の場所で 1 つのページですべての学習リソースにアクセスする 概要 > ラーニング.

セキュリティセンター機能のオプトアウト

インスタンスで セキュリティセンター 機能を無効にするには、「 KB1702514:インスタンスセキュリティセンターのコンポーネントを無効にするためのガイドライン (セキュリティセンターの使用中)」 に記載されている手順に従います。

セキュリティイベントの監視

潜在的なセキュリティイベントを特定して防止できるように、インスタンスのイベントメトリクスを分析します。

重要:

インスタンスセキュリティセンター (ISC) は 2024 年 9 月の時点で販売が終了しており、サポートもされていないため、新たにアクティブ化することもできません。

ServiceNow セキュリティセンター (SSC) が、今後推奨されるソリューションです。詳細については、「インスタンスセキュリティセンターから ServiceNow セキュリティセンターへの移行」を参照してください。

[インスタンスセキュリティ (Instance Security)] ホームページにあるイベントリボンでは、これらのメトリクスとそれに付随する詳細を分析して、インスタンス内の潜在的なセキュリティイベントを特定できます。

- イベントメトリクスごとにリアルタイムの単一スコア数が表示されます。この数値は、このインスタンスで 1 日にイベントが発生した回数を示しています。これらの単一スコアレポートは、対応するイベントが発生すると自動的に更新されます。
- 各イベントメトリクスには、日付範囲内のコンプライアンスの傾向とグラフ情報も含まれています。パフォーマンスアナリティクスジョブを実行する場合、この情報は毎日更新されます。詳細については、「イベント傾向の詳細分析」セクションを参照してください。

イベントタイプ

次のタイプのイベントのうち、少なくとも 6 件のイベントを監視できます。イベントが 6 件を超える場合は、イベントリボンの下にある左右の矢印を使用してスクロールします。イベントリボンの設定方法については、「セキュリティイベントリボンの構成」を参照してください。

通知設定	説明
admin ログイン	admin ロールを割り当てられたユーザーが、暦日の間にこのインスタンスにログインしようとした回数。
admin ユーザーが追加されました	暦日の間にこのインスタンスに追加された admin ロールを持つユーザーの数。たとえば、カウントが 10 であるが、admin ロールを割り当てられたユーザーが 4 名の場合は、セキュリティ上の問題がある可能性があります。
外部の受信メール	詳細については、「メールメトリクス 」を参照してください。
外部ログイン	暦日の間にこのインスタンスにログインした、snc_external ロールを持つユーザーの数 通常、これらのログインは、メンテナンス、サポート、コンサルティング、または監査の目的で発生します。このメトリクスを監視することで、外部ログイン試行が正規のものであり、潜在的なセキュリティ上の問題ではないことを確認できます。 外部ユーザーロールの割り当ての詳細については、「明示的なロール」を参照してください。
失敗したログイン	暦日の間にこのインスタンスで失敗したログインの数 このメトリクスは、ログインが試行され、インスタンスのセキュリティが侵害されていることを示している可能性があります。
代理操作	暦日の間にこのインスタンスで代理操作されたログインの数。ユーザーの代理操作の詳細については、「ユーザーの代理操作を行う 」を参照してください。
隔離されたファイル	暦日の間にこのインスタンスで アンチウイルススキャン を実行したときに隔離されたファイルの数。隔離されたファイルと アンチウイルススキャン の詳細については、「ウイルス対策メトリクス」および「アンチウイルススキャン」を参照してください。
セキュリティ昇格	暦日の間に、セキュリティアドミニストレーターが標準ユーザーに対して、割り当てられたユーザーロールを高い権限のセキュリティロールに変更することによって、セキュリティ権限を強化した回数。このような高権限のセキュリティ ロールに

通知設定	説明
	は、oauth_admin、admin、security_admin、および impersonator が含まれます。 • このメトリクスは、誰かが無許可のユーザーのセキュリティを昇格しようとした可能性があることを示しています。特定のセキュリティ侵害を検出するために、このメトリクスを単独で使用しないでください。代わりに、このメトリクスは、セキュリティ侵害の発生を確認するために別のメトリクスをチェックする必要があることを示す指標として扱ってください。 • ユーザーセキュリティの昇格の詳細については、「特権ロールへの昇格」および「昇格された権限ロール」を参照してください。
SNC ログイン	暦日の間にハイホッピング技術を使用してこのインスタンスにログインしたカスタマーサービス & サポート担当者の数。通常、これらのログインは、メンテナンス、サポート、コンサルティング、または監査の目的で発生します。 ServiceNow 社員のアクセスを制御する方法については、「ServiceNow アクセス制御」を参照してください。
スパム	詳細については、「メールメトリクス 」を参照してください。
信頼できる受信メール	詳細については、「メールメトリクス 」を参照してください。
信頼できない受信メール	詳細については、「メールメトリクス 」を参照してください。
ウイルスタイプ	暦日の間にこのインスタンスで発生したさまざまなタイプのウイルス対策イベントの数。ウイルス対策イベントタイプの詳細については、「ウイルス対策メトリクス」を参照してください。

イベント傾向の詳細分析

イベントメトリクスの傾向の詳細を表示するには、イベント数をクリックして [アナリティクスハブ] ページにアクセスします。インスタンスに表示される詳細は、メトリクスのタイプによって異なります。

たとえば、[セキュリティダッシュボードイベントログ] ページに試行の失敗を一覧表示するには、次の操作を実行します。

- [失敗したログイン] メトリクスを選択します。
- [アナリティクスハブ] ページで、[レコードを表示] をクリックします。
- 失敗したログイン試行の 1 つをクリックします。
- 詳細には、ログインを試行したユーザーの名前、ユーザーの IP アドレス、およびユーザーがアクセスしようとしたテーブル名が含まれます。

コア UI アナリティクスハブまたはプラットフォームアナリティクス KPI 詳細でイベントしきい値トリガーを設定して、インジケーターのスコア範囲内で特定のイベントが発生したときにアラートを提供できます。希望のスコアとイベントの実際のスコアとの差異を視覚化できるターゲットを設定することもできます。

たとえば、[失敗したログイン] のメトリクスのしきい値を 10 に設定できます。1 日に 10 回以上ログイン試行が失敗すると、特定のセキュリティ担当者にアラートが送信されます。同様のターゲットを設定して、1 日に 10 回ログインに失敗したときに、[アナリティクスハブ] で視覚的にハイライト表示させることもできます。

[イベント] リボンタイルと [アナリティクスハブ] に表示される傾向データとグラフは、パフォーマンスアナリティクスジョブが現地時間の 02:00 に実行された後に更新されます。詳細については、「日次コンプライアンススコア、傾向、およびグラフデータの更新方法 」を参照してください。

関連トピック

- インスタンスセキュリティセンター
- Now Intelligence
- アナリティクスハブ
- パフォーマンスアナリティクスのターゲットとしきい値

セキュリティイベントリボンの構成

Instance Security Center ホームページのセキュリティイベントリボンを構成して、運用内でインスタンスセキュリティの追跡に関連するイベントのみを含めるようにします。リボンに表示されるセキュリティイベントタイルの順序を変更することもできます。

始める前に

必要なロール：security_dashboard_user または admin

このタスクについて

セキュリティイベントリボンには、当初、標準セキュリティイベントの完全なセットが入力されています。自身の組織に関連しないイベントを削除してリボンをカスタマイズできます。

- たとえば、セキュリティ上の問題が内部担当者のアクションに起因していると思われる場合は、[admin ログイン]、[追加された admin ユーザー]、および [セキュリティ昇格] のイベントインジケーターを含めます。
- これらのインジケーターは、admin ロールを持つユーザーがログインを試行した回数、または admin ユーザーが追加された場合は、セキュリティロールを昇格させるために何が試行されたかを監視します。

> **注:** イベントリボンに表示されるセキュリティイベントのタイプについては、「セキュリティイベントの監視」を参照してください。

手順

1. 移動先 すべて > システムセキュリティ > インスタンスセキュリティセンター.
2. イベントリボンで、[編集] (< > Edit) をクリックします。
   [イベントを編集] フォームの [選択済み] 列には、既にリストされているイベントが含まれています。
3. イベントリボンにセキュリティイベントを追加するには、セキュリティイベントを [利用可能] 列から [選択済み] 列に移動します。
   リボンに表示されるイベントの順序を変更するには、イベントを選択し、上向きと下向きの矢印をクリックして正しい位置に移動します。

- インスタンスセキュリティセンターのイベントリボンに表示される順序でイベントを配置します。
- [利用可能] 列の上部に配置したイベントは、インスタンスセキュリティセンターのイベントリボンの左側に順番に表示されます。列の下部に配置されたイベントは、イベントリボンの右側に表示されます。

4. イベントリボンからセキュリティイベントを削除するには、次の操作を実行します。
   - [選択済み] 列で、イベントリボンから削除するセキュリティイベントを選択します。
   - イベントを [選択済み] 列から [利用可能] 列に移動します。
5. [保存] をクリックします。

関連トピック

インスタンスセキュリティセンター

セキュリティイベント通知の設定

特定のセキュリティイベントが発生したときに受信する通知のタイプを設定します。タイプごとに、メール、Now Mobile のプッシュ通知、または Slack や Microsoft Teams などのサードパーティのメッセージングアプリケーションで通知を受信するかどうかを指定します。

始める前に

サードパーティのメッセージングアプリケーションでセキュリティイベント通知を送信できるようにするには、Messaging Notification (com.glide.notification.messaging) プラグインをアクティブ化する必要があります。個々のユーザーが独自に設定する必要があります。詳細については、「メッセージングアプリケーションでの通知」を参照してください。

必要なロール：admin。

手順

1. インスタンスセキュリティセンターのホームページでプロファイルメニューをクリックし、[通知設定] をクリックします。

通知設定

通知設定	説明
admin ログイン	admin ロールを割り当てられた他のユーザーが別の IP アドレスからこのインスタンスにログインするたびに、選択したタイプの通知を送信します。
アドミンによるロック解除 (Admin Unlock)	高い権限を持つユーザーのアカウントのロックが解除されるたびに、選択したタイプの通知を送信します。
失敗したログイン	他のユーザーが *glide.user.max_unlock_attempts* プロパティで定義された試行回数よりも少ない回数でこのインスタンスへのログインに失敗するたびに、選択したタイプの通知を送信します。このプロパティを設定しない場合、デフォルト値は 5 です。

自動翻訳

通知設定	説明
	このプロパティの詳細については、「失敗したログイン試行のロックアウトを指定する」を参照してください。
追加された HP ロール (HP Role Added)	高い権限のセキュリティロール (oauth_admin、admin、security_admin、および impersonator ロールなど) が別のユーザーに付与されるたびに、選択したタイプの通知を送信します。 ユーザーセキュリティの昇格の詳細については、「特権ロールへの昇格」および「昇格された権限ロール」を参照してください。
代理操作	他のユーザーが代理操作を行うたびに、選択したタイプの通知を送信します。 ユーザーの代理操作の詳細については、「ユーザーの代理操作を行う」を参照してください。
セキュリティ昇格	他のユーザーがこのインスタンスでセキュリティアドミンロールに昇格するたびに、選択したタイプの通知を送信します。
週次ダイジェスト	選択したタイプの通知について週次ダイジェストを送信します。含まれる内容： ○ 1 週間にこのインスタンスで行われたすべてのセキュリティアクティビティのサマリー ○ インスタンスの現在の日次コンプライアンススコア

2. セキュリティイベントのタイプごとに、適切なチェックボックスをオンにして送信する通知のタイプを指定します。
それぞれに複数の通知方法を選択できます。

チェックボックス	説明
メール	このタイプのセキュリティイベントについてメールを送信します。
Slack	Slack を介してこのタイプのセキュリティイベントの通知を送信します。 注: この列は、ServiceNow AI Platform に対して Slack 統合を設定した場合にのみ表示されます。
チーム	Microsoft Teams を介してこのセキュリティイベントの通知を送信します。

チェックボックス	説明
	注: この列は、ServiceNow AI Platform に対して Microsoft Teams 統合を設定した場合にのみ表示されます。
プッシュ	このタイプのセキュリティイベントについて、Now Mobile にプッシュ通知を送信します。 注: この列は、Now Mobile に初めてログインした場合にのみ表示されます。
すべて選択	すべてのタイプのセキュリティイベントについて、特定のタイプの通知を選択します。 たとえば、各セキュリティイベントタイプに対してメール通知を受信する場合は、[メール] の上にある [すべて選択] をクリックします。

3. [保存] をクリックします。

日次コンプライアンススコアの確認とセキュリティプロパティの設定

日次コンプライアンススコアのメトリクスとセキュリティ構成プロパティを確認して、提案されたセキュリティ要件にインスタンスが準拠しているかどうかを確認します。[コンプライアンス構成の強化 (Hardening Compliance Configurations)] ページで非準拠のセキュリティプロパティを更新すると、日次のコンプライアンススコアに影響を与えることがあります。

重要:

インスタンスセキュリティセンター (ISC) は 2024 年 9 月の時点で販売が終了しており、サポートもされていないため、新たにアクティブ化することもできません。

ServiceNow セキュリティセンター (SSC) が、今後推奨されるソリューションです。詳細については、「インスタンスセキュリティセンターから ServiceNow セキュリティセンターへの移行」を参照してください。

日次コンプライアンススコアは、パーセンテージスコアです。これは、インスタンスのセキュリティプロパティの現在の設定が、ハードニング設定 で公開されているコンプライアンス値にどの程度準拠しているかに基づいています。

インスタンスの日次コンプライアンススコアを定期的に確認します。日次コンプライアンススコアを評価するときは、次のガイドラインに従ってください。

- 90 % 以上は、インスタンスが重要なセキュリティコントロールに準拠していることを示します。
- 50 % 以上 90 % 未満の場合は、セキュリティコンプライアンスのレベルが中程度であることを示します。
- 50 % 未満は、セキュリティコンプライアンスのレベルが低いことを示します。

インスタンスのセキュリティ設定調整によるコンプライアンスの強化

[コンプライアンス構成の強化 (Hardening Compliance Configurations)] ページを使用して、インスタンスの日次コンプライアンススコアに影響する非準拠セキュリティプロパティを強化して最適化

します。これを使用することで、インスタンスは公開されたセキュリティ強化基準に確実に準拠し、同時に会社のセキュリティ要件を満たせるようになります。

始める前に
必要なロール：security_dashboard_user または admin

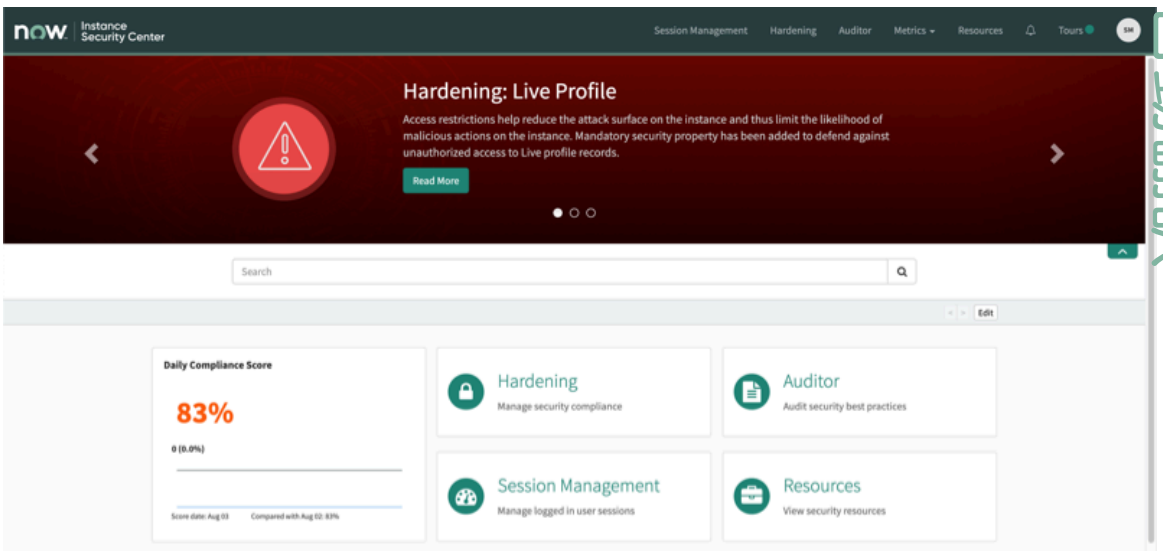
詳細な説明、コンプライアンス値、ServiceNow AI Platform のセキュリティ関連システムプロパティおよびプラグインについては、「ハードニング設定」の内容を参照してください。

- 一部のコンプライアンス値がインスタンスに適していない場合でも、セキュリティ関連のプロパティを設定または更新するときは、必ずインスタンスセキュリティ強化設定を参照してください。
- これらのプロパティを更新するときは、インスタンスが引き続き期待どおりに動作することを確認してください。セキュリティへの影響を判断するには、専門知識を持つ適切な内部担当者に相談してください。

注: admin ロールを持っている場合は、セキュリティコントロールを表示して編集できます。security_dashboard_user ロールを持っている場合は、セキュリティコントロールを表示できますが、編集することはできません。

手順

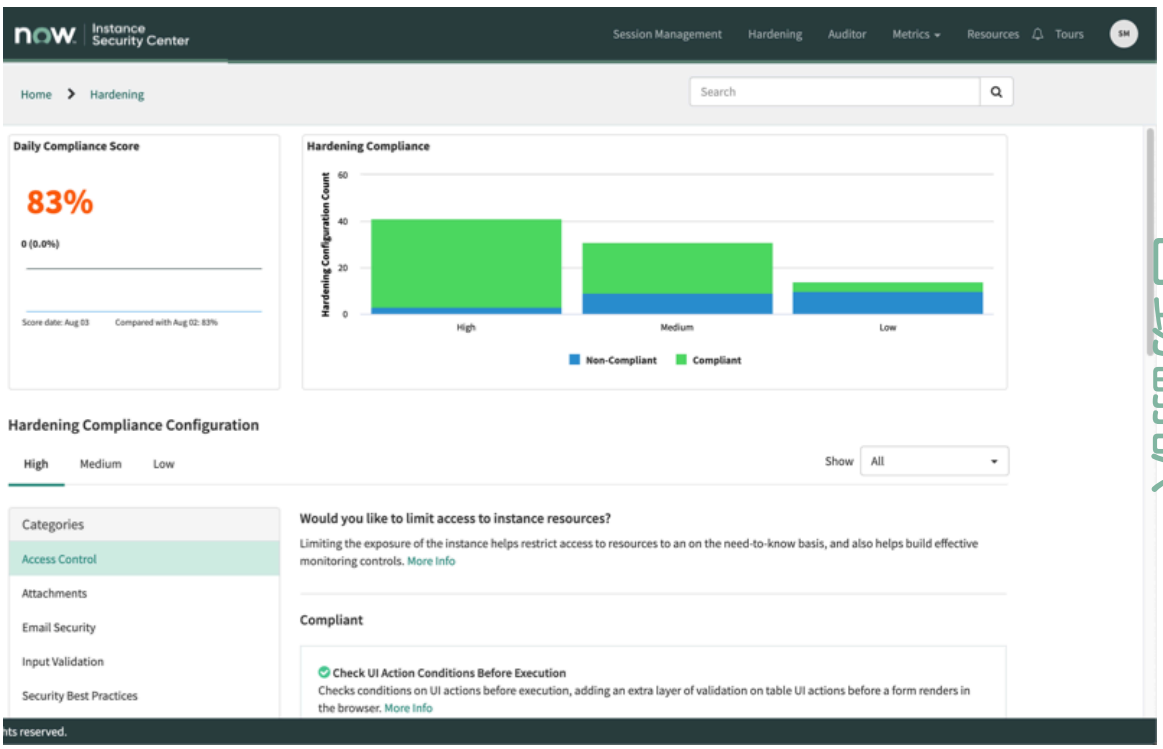
1. 移動先 すべて > システムセキュリティ > インスタンスセキュリティセンター.
2. [日次コンプライアンススコア] タイルまたは [ハードニング] リンクをクリックして、[コンプライアンス構成の強化 (Hardening Compliance Configurations)] ページにアクセスします。



3. [コンプライアンスの強化] チャートで、準拠および非準拠のセキュリティ構成プロパティの統計情報を表示します。

オプション	説明
準拠	インスタンスセキュリティ強化設定のコンプライアンス値に準拠するセキュリティ構成プロパティの数

オプション	説明
	注: [コンプライアンス構成の強化 (Hardening Compliance Configurations)] では、準拠するセキュリティプロパティの設定を変更することはできません。変更する場合は、[システムのプロパティ] で更新する必要があります。詳細については、「システムプロパティを追加する」を参照してください。
非準拠	インスタンスセキュリティ強化設定のコンプライアンス値に準拠しないセキュリティ構成プロパティの数。非準拠プロパティの設定を更新できます。

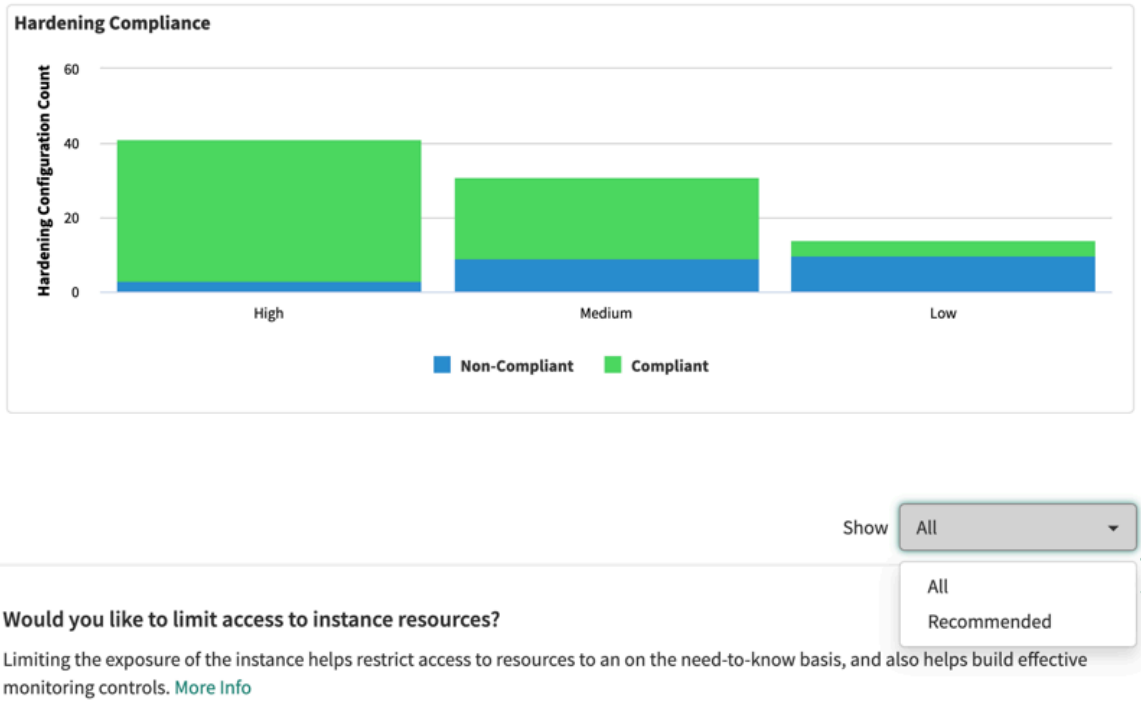


注: 日付範囲内の準拠または非準拠のセキュリティスコアの数を表示するには、[日次コンプライアンススコア] の下のスライダーにある青いドットを移動します。

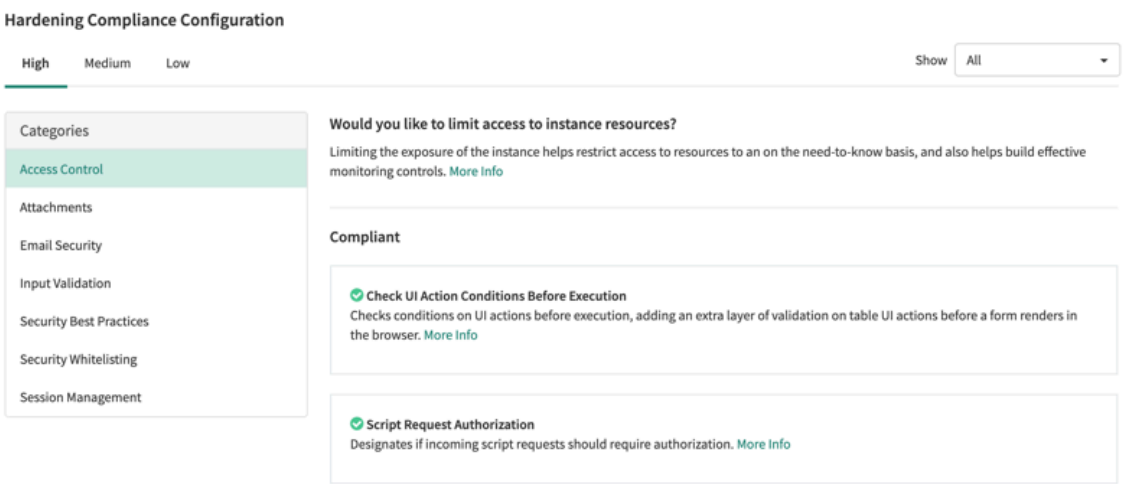
4. チャートの下にある [表示] リストで、すべてのセキュリティ構成プロパティにアクセスするのか、推奨されるプロパティのみにアクセスするのかを指定します。

オプション	説明
すべて	(デフォルト) 選択した各カテゴリのすべての準拠および非準拠のセキュリティ構成プロパティ
推奨	選択した各カテゴリには、推奨されるセキュリティ構成プロパティのみが表示されます。これらのセキュリティ構成プロパティは、ServiceNow AI Platform を保護するため

オプション	説明
	に使用される最も重要なプロパティを選択したサブセットです。 これらのセキュリティ構成プロパティは、ServiceNow AI Platform を保護するために設定する必要がある最低限の設定数であると考えてください。 **注:** インスタンスを完全に保護するには、[すべて] オプションを使用します。これには、推奨されるすべてのセキュリティ構成プロパティも含まれます。



5. [カテゴリ] で、アクセスしたいセキュリティ構成プロパティを含むカテゴリを選択します。



オプション	説明
アクセス制御	アクセス制御は、特定のリソースの使用をどのユーザーが許可されているかに基づいて、特定のリソースへのユーザーアクセスを許可するか拒否するかを決定します。詳細については、「インスタンスセキュリティ強化設定」の「アクセス制御」を参照してください。
添付ファイル	添付ファイルのセキュリティコントロールを使用すると、受信した添付ファイルを検証して、攻撃者から送信された悪意のあるファイルからインスタンスを保護できます。詳細については、「インスタンスセキュリティ強化設定」の「AttachmentCreator SOAP Web サービスでファイルの MIME タイプを検証する (Security Center 1.3 の新機能、1.5 で更新)」を参照してください。
メールセキュリティ	メールセキュリティには、すべての受信メールに対応する適切なセキュリティポリシーを確実に実施するためにアドミニストレーターが構成できる、セキュリティ構成プロパティが含まれます。詳細については、「インスタンスセキュリティ強化設定」の「スパムメールのスコアリングとフィルタリングを有効化する (Security Center 1.3 で更新)」を参照してください。
入力の検証	入力検証には、ソースに関係なくアドミニストレーターが不正な形式のデータの入力を最小限に抑えるために構成できる、セキュリティ関連プロパティが含まれています。詳細については、「インスタンスセキュリティ強化設定」の「検証、サニタイズ、およびエンコーディング」を参照してください。
安全な通信	セキュア通信プロパティは、HTTP トラフィックの転送を保護するためにアドミニストレーターが構成できるプロパティです。詳細については、「インスタンスセキュリティ強化設定」の「通信」を参照してください。
セキュリティベストプラクティス	セキュリティのベストプラクティスには、アドミニストレーターが一定の時間間隔内に定期的に実行する必要があるセキュリティタスクが含まれており、関連する構成プロパティが含まれています。詳細については、「インスタンスセキュリティ強化設定」の「セキュリティのベストプラクティス」を参照してください。
セキュリティ包含リスト	セキュリティ包含リストには、動作を既知の包含リストに制限するようにアドミニストレーターが構成できる、セキュリティ関連プロパティが含まれています。
セッション管理	セッション管理には、アドミニストレーターが ServiceNow AI Platform で安全なセッション管理を確保するために構成できる、セキュリティ関連プロパティが含まれています。詳細に

オプション	説明
	ついては、「インスタンスセキュリティ強化設定」の「セッション管理」を参照してください。

6. 選択したカテゴリの非準拠セキュリティプロパティを設定します。
   - 特に指定しないかぎり、スイッチをオンにスライドするとセキュリティプロパティが推奨設定に設定されます。たとえば、ほとんどのコントロールは true または false に設定しますが、中には値の入力 (カンマ区切りの値リストなど) が必要なものがあります。
   - セキュリティコントロールのための専用のインスタンスセキュリティ強化設定トピックにアクセスして詳細を確認するには、[その他の情報] をクリックします。

結果
日次コンプライアンススコアは、非準拠のセキュリティコントロール設定に対して行われた変更に応じて増減します。

関連トピック

インスタンスセキュリティセンター

日次コンプライアンススコア、傾向、およびグラフデータの更新方法

Instance Security Center の傾向およびグラフデータは、パフォーマンスアナリティクスジョブがローカル時間の 02:00 に実行された後に更新されます。これは [日次コンプライアンススコア] タイル、イベントリボンのタイル、さらにイベントタイルの 1 つをクリックした場合には [アナリティクスハブ] ページの詳細に表示されます。

[AppSec] 日次データ管理ジョブは、夜間に次のタスクを実行する定期スケジュールジョブです。

1. 最初にスケジュール設定した時点で、[AppSec] 日次データ管理ジョブと [PA AppSec] 日次データ収集ジョブに、有効なユーザーがアサインされているかどうかが確認されます。
   - [実行方法] フィールドに有効なユーザーを入力した場合、ジョブは処理を続行します。有効なユーザーとは、インスタンスからロックアウトされておらず、admin ロールが割り当てられているユーザーのことです。
   - 無効なユーザーを入力した場合、インスタンスセキュリティセンターのローテーションセキュリティバナーの上にエラーメッセージが表示されます。

     注: スケジュール済みジョブの実行時にアサインされたユーザーを更新する方法の詳細については、「データコレクションジョブの作成またはスケジュール」を参照してください。

2. ビジネスロジックを実行して、[コンプライアンス構成の強化 (Hardening Compliance Configurations)] ページで構成したセキュリティプロパティのコンプライアンスステータスを設定します。詳細については、「日次コンプライアンススコアの確認とセキュリティプロパティの設定」を参照してください。
3. [PA AppSec] 日次データ収集パフォーマンスアナリティクスジョブを実行して、コンプライアンスデータを収集し、日次コンプライアンススコアを更新します。

日次コンプライアンススコアの手動更新

admin ロールが割り当てられている場合は、[リフレッシュ] をクリックして、いつでも日次コンプライアンススコアを更新して再計算することができます。

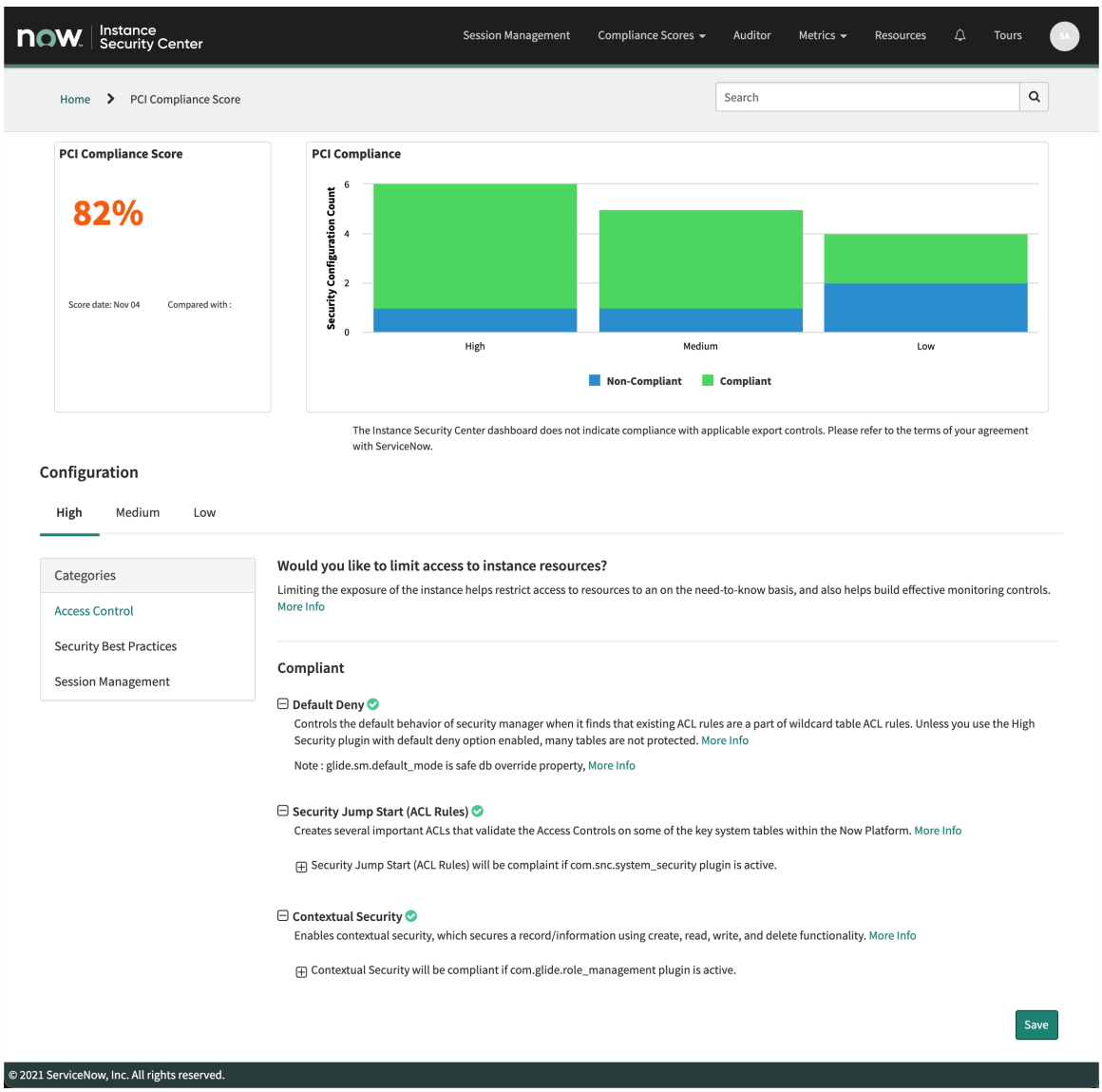
注: [リフレッシュ] ボタンは、security_dashboard_user ロールが割り当てられているユーザーには表示されません。

- リフレッシュ機能は、日次データ収集パフォーマンスアナリティクスジョブと同じタスクを実行しますが、バッチ処理ではなくリアルタイムで実行します。
- 通常は、日次コンプライアンススコアを更新して、インスタンスセキュリティアクティビティの影響をすぐに表示する場合に使用します。
- 更新されたスコアが表示されるまでに少し時間がかかる場合があります。

注: アップグレードを実行すると (London から Zurich へなど)、Instance Security Center (ISC) プラグインが自動的にアクティブ化されます。提供される修正スクリプトによって、ロールが割り当てられていないカスタムユーザーが自動的に割り当てられます。

PCI コンプライアンススコアダッシュボード

PCI コンプライアンススコアダッシュボードは、インスタンスがペイメントカード業界 (PCI) のセキュリティ標準にどの程度準拠しているかを示します。ダッシュボードを使用してコンプライアンススコアを表示し、構成を変更してセキュリティを向上します。



必要な ServiceNow AI Platform ロール

PCI コンプライアンススコアダッシュボードを表示するには、security_dashboard_user または admin が必要です。

PCI コンプライアンススコアダッシュボードへのアクセス

ダッシュボードを開くには、次の場所に移動します。 システムセキュリティ > インスタンスセキュリティセンター. Instance Security Center から、ヘッダーの [コンプライアンススコア] をクリックし、次に **[PCI** コンプライアンススコア] を選択します。

ユースケース

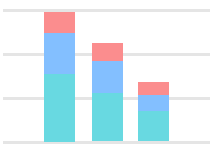
組織内のさまざまなユーザーがこのダッシュボードをどのように使用するかの例については、次のユースケースを参照してください。

ユーザー	ダッシュボードの使用法
セキュリティダッシュボードユーザー (security_dashboard_user)	インスタンスセキュリティコンプライアンスを継続的に監視および管理します。
アドミン (admin)	インスタンスセキュリティコンプライアンスを継続的に監視し、セキュリティ脅威を検出して対応します。

インジケーター

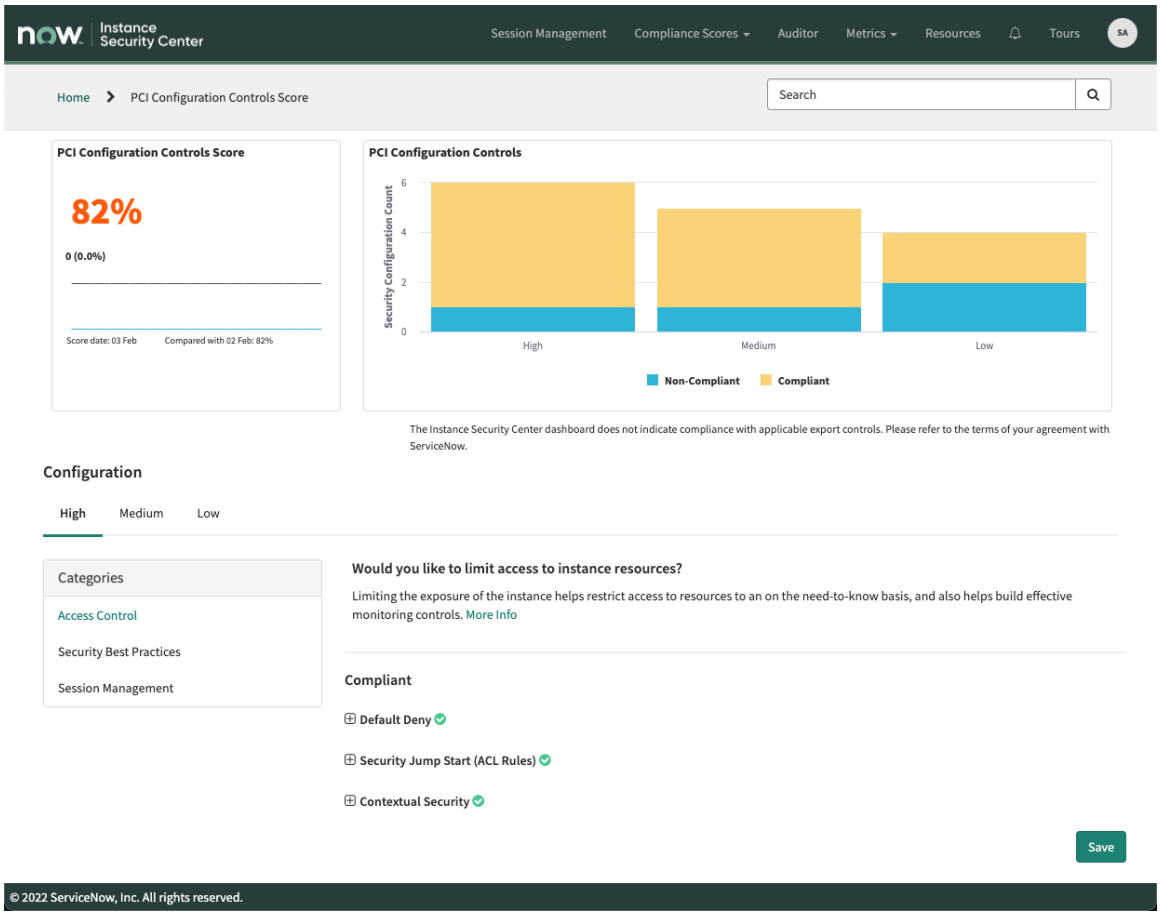
インジケーター	説明
PCI コンプライアンススコア	インスタンスのコンプライアンススコアをパーセンテージで表示します。この割合は、コンプライアンス標準を満たすインスタンスのセキュリティ構成の割合を表します。このインジケーターには、コンプライアンススコアが計算された日付と、以前に計算されたスコアとの比較も表示されます。

データの表示方法

タイトル	タイプ	ソーステーブル	説明
PCI コンプライアンス	 積み重ね式の棒グラフ	セキュリティ構成 [isc_security_configuration]	高、中、低のカテゴリで準拠と非準拠のセキュリティ構成を表示します。レポートの領域をクリックすると、一致するセキュリティ構成が表示されます。

PCI 構成コントロールスコアのダッシュボード

PCI 構成コントロールスコアのダッシュボードを使用して、PCI 構成を確認し、非準拠のセキュリティチェックを特定します。非準拠のセキュリティチェックの構成は、Instance Security Center から変更できます。



必要な **ServiceNow AI Platform** ロール

PCI コンプライアンススコアダッシュボードを表示するには、security_dashboard_user または admin ロールが必要です。

PCI 構成コントロールスコアのダッシュボードへのアクセス

ダッシュボードを開くには、次の場所に移動します。 システムセキュリティ > インスタンスセキュリティセンター. Instance Security Center から、ヘッダーの [コンプライアンススコア] をクリックし、次に **[PCI** 構成コントロールスコア] を選択します。

ユースケース

組織内のさまざまなユーザーがこのダッシュボードをどのように使用するかの例については、次のユースケースを参照してください。

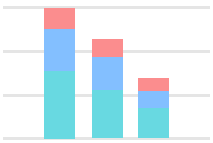
ユーザー	ダッシュボードの使用法
セキュリティダッシュボードユーザー (security_dashboard_user)	インスタンスの PCI 構成コントロールのコンプライアンスを継続的に監視および管理します。PCI 構成を変更してコンプライアンスを確保し、インスタンスのセキュリティを向上します。
アドミン (admin)	PCI 構成コントロールコンプライアンスを継続的に監視し、潜在的なセキュリティ脅威を検出して対応します。

インジケーター

PCI 構成コントロールスコア

インスタンスの PCI 構成コントロールスコアをパーセンテージで表示します。このパーセンテージは、コンプライアンス標準を満たすインスタンスのセキュリティ PCI コントロール構成の割合を表します。このインジケーターには、コンプライアンススコアが計算された日付と、以前に計算されたスコアとの比較も表示されます。

データの表示方法

タイトル	タイプ	ソーステーブル	説明
PCI 構成コントロール	 積み重ね式の棒グラフ	セキュリティ構成 [isc_security_configuration]	高、中、低のカテゴリで準拠と非準拠の PCI コントロール構成を表示します。レポートの領域をクリックすると、一致するセキュリティ構成が表示されます。

誤ったセキュリティ定義のスキャン

監査ツールを実行してインスタンスをスキャンし、誤ったセキュリティ定義を見つけます。 インスタンスのセキュリティ体制を改善するために修正できる結果を提供します。

重要:

インスタンスセキュリティセンター (ISC) は 2024 年 9 月の時点で販売が終了しており、サポートもされていないため、新たにアクティブ化することもできません。

ServiceNow セキュリティセンター (SSC) が、今後推奨されるソリューションです。詳細については、「インスタンスセキュリティセンターから ServiceNow セキュリティセンターへの移行」を参照してください。

監査ツールは、システム構成を分析するインスタンスの健全性の「総合」アセスメントを実行します。セキュリティスキャンでは、現在のセキュリティ構成がベストプラクティスの定義、およびセキュリティプロパティのコンプライアンス値と比較されます。

インスタンスのセキュリティの観点から、継続して実行すべきことや改善できる可能性がある箇所についてのインサイトと推奨事項を提供します。 これらのインサイトと推奨事項は、次の質問に答えるために役立ちます。

- 適切なセキュリティ関連プロパティが設定されていますか？
- High Security プラグインが有効になっていますか？
- 適切なアクセス制御ルールが存在しますか？

監査ツールの実行とスキャン結果の分析

1. 監査ツールを実行するには、インスタンスセキュリティセンターのホームページで [監査] をクリックします。
2. 完了したら、[スキャン結果] を開いてセキュリティ検出結果を確認および分析します。

3. 特定のスキャン結果の詳細を確認するには、結果番号をダブルクリックします。この情報には、ステータス、スキャンタイプ、実行時間、およびエラーメッセージが含まれています。
4. 監査ツールの各検出結果には、解決策の詳細と、それに対処する方法に関する製品コンテンツの URL が含まれています。文書化されたガイドラインに従って、各検出結果の問題を解決します。

関連トピック

- ハードニング設定
- 高セキュリティプラグインを有効にする (Security Center 1.3 で更新)
- アクセス制御リストのルール

インスタンスメトリクスの監視

インスタンスのユーザー、エクスポート、認証、メール、およびウイルス対策のメトリクスを監視します。たとえば、インスタンスの信頼できないドメインと信頼できるドメインからのスパム、外部メール、受信メールのメトリクスをチェックすることで、メールのセキュリティを監視できます。これらのメトリクスを分析して、インスタンスで実行されているアクティビティに関連した例外的なセキュリティ動作を探します。

重要:

インスタンスセキュリティセンター (ISC) は 2024 年 9 月の時点で販売が終了しており、サポートもされていないため、新たにアクティブ化することもできません。

ServiceNow セキュリティセンター (SSC) が、今後推奨されるソリューションです。詳細については、「インスタンスセキュリティセンターから ServiceNow セキュリティセンターへの移行」を参照してください。

ユーザーメトリクス

ユーザーメトリクスを分析して、インスタンスで特定のタイプのユーザーアクティビティに関連した例外動作を探します。

先月/過去 6 か月間/去年のログインなし

先月、過去 6 か月以内、および昨年の暦年に、インスタンスにログインしていないユーザーの数を示します。特定のメトリクスのユーザー詳細を表示するには、次の操作を実行します。

- メトリクスをクリックすると、指定された期間中にインスタンスにログインしていないユーザーのリストが表示されます。
- ユーザー名をクリックすると、そのユーザーの詳細が表示されます。

高権限のロールを持つユーザー

次のような高い権限のロールタイプを持つユーザーの数を示します。

ユーザーロール	説明
admin	セキュリティ上の制約に関係なく、システムの機能とデータすべてに対してアクセスを提供するプライマリアドミニストレーターロール
ais_high_security_admin	ユーザーが AI 検索 の高セキュリティ設定にアクセスできるようにするための、昇格された権限ロール。詳細については、以下を参照してください。 ロールのアサイン先 AI 検索 アドミニストレーターとユーザー .

ユーザーロール	説明
password_reset_admin	パスワードリセットアクティビティのステータスを表示し、潜在的なセキュリティ上の脅威を特定し、パスワードセキュリティポリシーに準拠しているかどうかを監視できるようにするアドミニストレーターロール。詳細については、以下を参照してください。 パスワードリセット およびパスワード変更レポートとログ .
script_include_admin	スクリプトインクルードにもアクセスできるアドミニストレーターロール
security_admin	ユーザーがアクセス制御と高セキュリティ設定を作成および変更できるようにするための、昇格された権限ロール。詳細については、「security_admin ロール」を参照してください。
user_admin	ユーザー、ロール、ユーザーグループ、ロール、および部門のアサインも管理できるアドミニストレーターロール。

注: これらの管理ロールタイプの詳細については、「特別管理ロール 」を参照してください。

特定のユーザーロールメトリクスのユーザー詳細を表示するには、次の操作を実行します。

- ユーザー数ロールメトリクスをクリックすると、その高い権限のロールタイプを持つユーザーのリストが表示されます。
- ユーザー名をクリックすると、そのユーザーの詳細が表示されます。その後、これらのセキュリティ上重要なロールが適切な担当者に割り当てられているかどうか判断できます。

ユーザーの傾向

次のタイプのユーザーについて、一定期間のカウント傾向情報を表示します。

カウントタイプ	説明
アクティブユーザー	インスタンスでアクティブとしてマークされているユーザーの数
非アクティブなユーザー	インスタンスで非アクティブとしてマークされているユーザーの数
ロックアウト	インスタンスからロックアウトされているユーザーの数

特定のユーザー数のユーザー詳細 (ロックアウトされたユーザーなど) を表示するには、次の操作を実行します。

- [ロックアウトされたユーザー] メトリクスをクリックします。
- [アナリティクスハブ] ページで、[レコードを表示] をクリックします。
- ユーザー名をクリックすると、そのユーザーの詳細が表示されます。その後、このユーザーがロックアウトされた理由があるかどうかを判断して、その状況を解決できます。

イベントの傾向

次のタイプのイベントについて、一定期間のカウント傾向情報を表示します。

イベントタイプ	説明
admin ログイン	特定の日にログインした、権限の高いアドミンユーザーロールを持つユーザーの数

イベントタイプ	説明
外部ログイン	暦日の間にこのインスタンスにログインした、snc_external ロールを持つユーザーの数通常、これらのログインは、メンテナンス、サポート、コンサルティング、または監査の目的で発生します。このメトリクスを監視することで、外部ログイン試行が正規のものであり、潜在的なセキュリティ上の問題ではないことを確認できます。
失敗したログイン	特定の日に失敗したログイン試行回数
代理操作	特定の日に他のユーザーの代理操作でログインしているユーザーの数
セキュリティ昇格	暦日の間に、セキュリティアドミニストレーターが標準ユーザーに対して、割り当てられたユーザーロールを高い権限のセキュリティロールに変更することによって、セキュリティ権限を強化した回数。このような高権限のセキュリティ ロールには、oauth_admin、admin、security_admin、および impersonator が含まれます。
SNC ログイン	特定の日にハイホッピング技術を使用してこのインスタンスにログインしたカスタマーサービス & サポートの数

特定のイベント数のユーザー詳細 (代理操作など) を表示するには、次の操作を実行します。

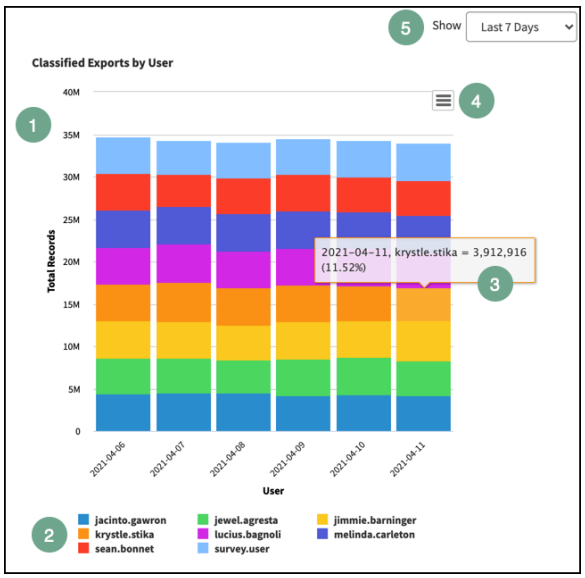
- ユーザー数のメトリクスをクリックします。[セキュリティダッシュボードイベントログ] ページには、そのタイプのイベントのイベントログが一覧表示されます。
- ユーザー名をクリックすると、そのイベントの詳細が表示されます。

関連トピック

アナリティクスハブ

エクスポートメトリクス

エクスポートメトリクスを分析して、最もよくエクスポートされるデータと、最も多くのデータをエクスポートするユーザーを確認します。



1. エクスポートチャート

   各レポートでは、色分けされたキーを使用して、エクスポートを実行したユーザーを示すエクスポートイベントの数が表示されます。列のカラー部分をクリックすると、そのエントリに一致するエクスポートイベント [isc_export_event] レコードリストが表示されます。

2. レポートキー

   レポートの下部にあるキーは、どの色でどのユーザーやテーブルを特定しているのかを示します。

3. プレビューポップアップ

   チャート内のエントリをポイントすると、ポップアッププレビューが表示されます。このプレビューには、ユーザーまたはテーブルの名

前、エクスポートの数、およびその列の合計に対する割合が表示されます。

4. 画像のエクスポート

   レポートを画像として保存するには、アイコンをクリックします。レポートは PNG または JPEG 形式で保存できます。

5. レポート日付範囲

   [表示] リストを使用して、過去 24 時間または過去 7 日以内のエクスポートを表示します。

エクスポートメトリクスレポート

エクスポートメトリクスのページには 4 つのレポートが表示されます。

ユーザー別のエクスポート

[ユーザー別のエクスポート] レポートを使用すると、最も多くのデータをエクスポートしているユーザーを確認できます。

ユーザー別に分類されたエクスポート

[ユーザー別に分類されたエクスポート] レポートを使用すると、機密情報、制限付き情報、個人情報などの分類に一致するデータを最も多くエクスポートしているユーザーを確認できます。アドミニストレーターは、このレポートで使用する分類を [設定] タブで定義できます。

テーブル別のエクスポート

[テーブル別のエクスポート] レポートを使用すると、最も頻繁にエクスポート元になるテーブルを確認できます。

テーブル別に分類されたエクスポート

[テーブル別に分類されたエクスポート] レポートを使用すると、機密情報、制限付き情報、個人情報などの分類に一致し、最も頻繁にエクスポート元になるテーブルを確認できます。アドミニストレーターは、このレポートで使用する分類を [設定] タブで定義できます。

注: メトリクスレポートをエクスポートすると、エクスポートイベントのみが追跡されます。REST API やワークフローなどの他のソースからのエクスポートは、この機能の一部として追跡されることはありません。

エクスポートメトリクスの設定

[設定] タブの設定オプションを使用して、レポート結果を絞り込みます。

[設定] タブをクリックして、エクスポートメトリクスの設定にアクセスします。

設定構成フィールド

エクスポートメトリクスの構成

設定	説明
メトリクスの分類	このフィールドで分類の追加または削除を行い、[ユーザー別に分類されたエクスポート] および [テーブル別に分類されたエクスポート] レ

エクスポートメトリクスの構成 (続く)

設定	説明
	ポートに組み込むエクスポートを決定します。これらのレポートは、次の分類をサポートしています。 • 個人識別可能情報 • 機密 • 制限付き • 内部 • 公開 データ分類の詳細については、「データ分類」を参照してください。
アラートの分類	このフィールドで分類の追加または削除を行い、インスタンスセキュリティ通知をトリガーするエクスポートを決定します。[メトリクスの分類] フィールドでサポートされている分類がここでサポートされます。これらのアラートの詳細については、[インスタンスセキュリティセンター] ページのセキュリティ通知に関するセクションを参照してください。[レコードのしきい値] フィールドは、インスタンスがトリガーされてアラートが発生する前にエクスポートされるレコードの数を定義します。
レコードのしきい値	アラートをトリガーするためにエクスポートする必要があるレコードの数。アラートをトリガーするには、これらのレコードが [アラートの分類] フィールドにリストされている分類とも一致する必要があります。

[保存 (⌘ + s)] ボタンを使用して設定を保存します。

認証メトリクス

認証メトリクスを分析して、使用頻度の低い IP アドレス、失敗したログイン、ユーザーが使用している認証スキームのタイプなど、認証に関連する情報を表示します。

[認証メトリクス] ページを使用して、認証構成に関連するレポートを表示します。このタブには次のレポートが表示されます。

注: [認証メトリクス] ページには、**REST API Access Policy** プラグイン (com.glide.rest.policy) が必要です。この機能の詳細については、「REST API アクセスポリシー」を参照してください。

認証ポリシーのない API

アクセスポリシーのないすべての API のリアルタイム数を表示します。

強化：アカウント復旧フロー

強化：ロールベースの MFA 機能関連の設定

Web サービスアクセスのみのアカウント

User [sys_user] テーブルで Web サービスアクセスオプションが有効になっているすべてのユーザーレコードの数を表示します。

30 日後に期限が切れる X509 証明書

X.509 Certificates [sys_certificate] テーブルで、30 日以内に期限が切れる X.509 証明書の総数を表示します。

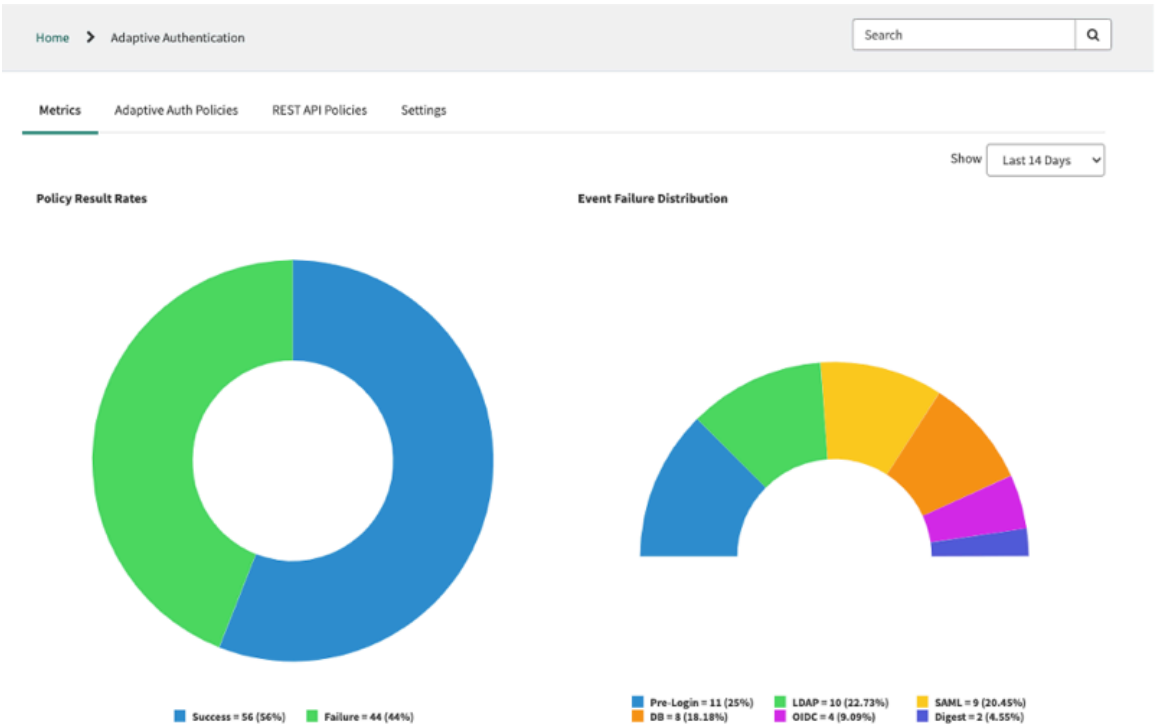
適応認証メトリクス

適応認証メトリクスを分析して、インスタンスでの適応認証の使用状況を監視し、インサイトを追加します。

適応認証メトリクスページを使用して、1 つの場所で適応認証に関連付けられたレポート、設定、およびポリシーを表示します。セキュリティアドミニストレーターは、レポートを使用して適応認証ポリシーの結果を監視できます。このデータを使用して、インサイトを得てポリシーを調整し、パフォーマンスを向上します。

注: 適応認証メトリクスページでは、**Adaptive Authentication** (com.snc.adaptive_authentication) プラグインが必要です。この機能の詳細については、「適応認証」を参照してください。

メトリクス



[メトリクス] タブを使用して、適応認証構成に関連するレポートを表示します。このタブには次のレポートが表示されます。

- ポリシー結果評価
- イベント失敗の分布
- イベント成功の分布
- 拒否された IP アドレス
- 認証ユーザーログイン

- API ユーザーログイン
- 認証傾向

[表示] リストを使用して、表示するレポートの期間を選択します。

適応認証ポリシー

[適応認証ポリシー **(Adaptive Auth Policies)**] タブを使用して、インスタンスの適応認証ポリシーとポリシーコンテキストを表示します。これらのリストのエントリをクリックすると、関連するレコードが表示されます。これらのレコードの詳細については、「適応認証」を参照してください。

設定

[設定] タブを使用して、適応認証システムのプロパティを表示および設定します。これらのプロパティの詳細については、「適応認証プロパティの設定」を参照してください。

メールメトリクス

メールメトリクスを分析して、インスタンスに配信される受信メールに関連した例外動作を探します。たとえば、メトリクスが特定のドメインからのスパムメールのスパイクを示している場合は、インスタンスへの配信を防ぐ受信アクションを定義できます。

各メールメトリクスについて、暦日の間に配信または送信されたメールのタイプごとにカウント数が表示されます。

通知設定	説明
外部の受信メール	該当の暦日に外部メールドメインからインスタンスに配信された受信メールの数 **注:** 外部メールドメインとは、*security.list.internal.domains* システムプロパティにリストされていないドメインです。このプロパティは内部のメールドメインのみを追跡するからです。このプロパティの詳細については、「利用可能なシステムプロパティ」を参照してください。
スパム	該当の暦日にインスタンスに配信され、スパムとしてマークされた受信メールの数件数が過去の傾向と一致しない場合、インスタンスのセキュリティを侵害しようとする試みが行われたことを示している可能性があります。
信頼できる受信メール	該当の暦日に信頼できると指定されたメールドメインからインスタンスに配信された受信メールの数
信頼できない受信メール	該当の暦日に信頼できないと指定されたメールドメインからインスタンスに配信された受信メールの数。 [信頼できない/信頼できるドメイン] フォームで信頼できないメールドメインまたは信頼できるメールドメインを指定して、それらのドメイン

通知設定	説明
	から送信された受信メールを追跡できるようにすることができます。信頼できないメールドメインまたは信頼できるメールドメインを指定する方法については、「信頼できないメールドメインと信頼できるメールドメインの指定」を参照してください。

メールメトリクスをクリックした後、次のいずれかをクリックすることで、インスタンスで発生する可能性があるメール セキュリティの問題を調べることができます。

コマンド	説明
チャート	選択したメールタイプ (スパム、外部受信、信頼できない、または信頼できる) の受信メール数と数の傾向
レコード	選択したメールタイプの日次カウントを侵害する個々のメールレコード
詳細情報	選択したメールタイプの追加情報

注: メールメトリクスは、インスタンスに配信される受信メールにのみ適用されます。メトリクスは、全社的なメールサーバーを介して処理される通常のトラフィックには適用されません。受信アクションの定義と受信メールの処理への影響については、「受信メールアクション」を参照してください。

信頼できないメールドメインと信頼できるメールドメインの指定

特定のメールドメインを、信頼できないドメインまたは信頼できるドメインに指定して、インスタンス内のこれらのソースからの受信メールのメトリクスを監視できるようにします。

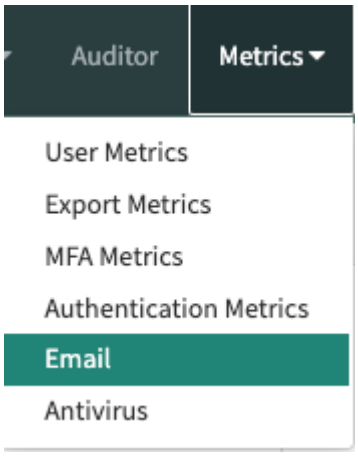
始める前に
必要なロール：security_dashboard_user または admin

このタスクについて
信頼できないドメインまたは信頼できるドメインがインスタンスにメールを送信すると、その日次カウントが [メール] ページの [信頼できない受信メール] または [信頼できる受信メール] メトリクスに表示されます。これらのドメインからのメールアクティビティを追跡し、メールログを使用して特定の受信メールを表示できます。また、信頼できないドメインまたは信頼できるドメインからアクティビティが発生するたびに通知するユーザー (通常はマネージャーまたはセキュリティアナリスト) を指定することもできます。

注: メールドメインを信頼できないと指定する場合、これはセキュリティトラッキングのみを目的としています。アドミニストレーターは、信頼できないドメインからのメールを無視するようにシステムアドレスフィルターを設定することもできます。メールをフィルタリングして配信をブロックする方法については、「システムアドレスフィルター」を参照してください。

手順

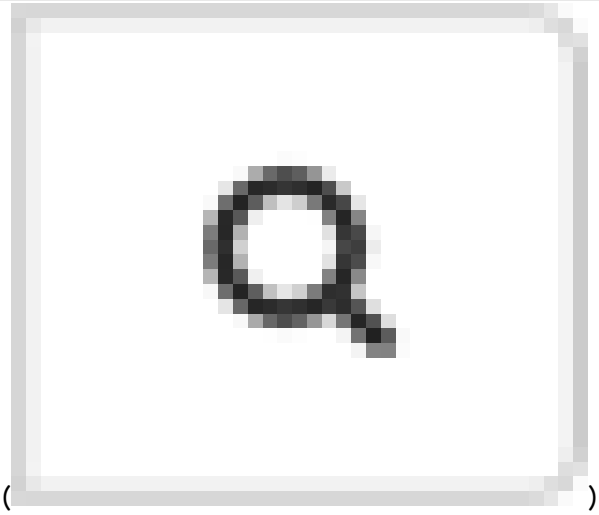
1. 移動先 すべて > システムセキュリティ > インスタンスセキュリティセンター.
2. インスタンスセキュリティセンターのホームページで、[メトリクス] メニューから [メール] を選択します。



3. [メール] ページの [信頼できない/信頼できるドメイン] セクションで、[新規] をクリックします。
4. フォームのフィールドに入力します。

[信頼できない/信頼できるドメイン] フォーム

フィールド	説明
ドメイン	信頼できないまたは信頼できるという指定を行うメールドメイン名。たとえば、ServiceNow の従業員がインスタンスに信頼できるメールを送信できるようにするには、「servicenow.com」と入力します。
カテゴリ	メールドメインが信頼できないか信頼できるかを示すカテゴリ： 信頼できない メールドメインを信頼できないものとして指定します。これを使用して、インスタンスに不審なメールや潜在的なセキュリティ上の脅威をもたらすメールを送信するドメインを特定します。 信頼できる メールドメインを信頼できるものとして指定します。ドメインからの受信メールにセキュリティ上の脅威がないことがメトリクスで示されている場合に、これを使用してそのドメインを識別します。信頼できるドメインに指定すると、受信メールアクティビティを経時的に追跡できます。
アクティブ	指定されたメールドメインに対して、信頼できない/信頼できるステータスを有効または無効にするためのチェックボックス
通知	信頼できないドメインまたは信頼できるドメインでアクティビティが発生したときにメールで通知するユーザーの名前。スポットライト検索アイコン

フィールド	説明
	 ( ) をクリックして、ユーザーの名前を検索します。通知を送信しない場合、[通知 **(Notify)]** フィールドは空白のままにします。

5. [保存] を選択します。

結果
信頼できない/信頼できるメールドメインの情報は、[メール] ページの [信頼できない/信頼できるドメイン] リストにも追加されます。

関連トピック

- インスタンスセキュリティセンター
- メールメトリクス

ウイルス対策メトリクス

アンチウイルススキャン プラグインが有効になっている場合は、インスタンスでアンチウイルススキャンが実行され、添付ファイルからのウイルス感染からインスタンスを保護します。

次のメトリクスはアクティビティの最後 60 日間表示され、アンチウイルススキャン機能の有効性を評価できます。

ウイルス対策イベント

ウイルス対策イベントは、インスタンス内のウイルス対策イベントの数を日付別に示します。ウイルス対策イベントにアクセスするには、次に移動します: システムセキュリティ > インスタンスセキュリティセンター をクリックし、[メトリクス] タブを選択します。色分けされたグラフの線は、次のタイプのウイルス対策イベントを表しています。

色	説明
青	指定された日付にこのインスタンスでアンチウイルススキャンによって隔離されたファイルの数
緑	インスタンスにダウンロードされ、指定された期間隔離された感染ファイルの数。これらのファイルは主に、ウイルスまたは不正なコードを含むメールの添付ファイルです。
黄	指定された日付にこのインスタンスで削除された隔離ファイルの数

色	説明
オレンジ	指定された日付にこのインスタンスで復元された隔離ファイルの数 **注:** アンチウイルススキャンが実行されて誤検出が見つかった場合、隔離ファイルを復元してインスタンスでアクセスできるようにすることができます。

- [アナリティクスハブ] ページにアクセスして、特定の日付の詳細なスコアカードとアナリティクス情報を表示するには、[ウイルス対策イベント] グラフで色付きの線をクリックします。たとえば、青色のグラフィックラインをクリックすると、特定の日付に隔離されたファイルのアナリティクス情報が表示されます。
- [アナリティクスハブ] ページで次のブレークダウンを表示するには、をクリックし、次の項目をクリックします。

ブレークダウン	説明
AppSec - ウイルス対策イベントソース	ウイルス対策イベントのソース ○ アップロード時：感染したファイル (通常は添付ファイル) のアップロードが原因で発生しました。 ○ 隔離から：感染したファイル (通常は添付ファイル) の隔離によって発生しました。 ○ ダウンロード時：感染したファイル (通常は添付ファイル) のダウンロードが原因で発生しました。 ○ レコードから： テーブル内の感染したレコードが原因で発生しました。
AppSec - ウイルス対策イベントタイプ	ウイルス対策イベントのタイプ ○ 隔離：ファイル (通常は添付ファイル) の隔離が原因で発生しました。 ○ ダウンロード済み：ファイル (通常は添付ファイル) のダウンロードが原因で発生しました。 ○ 復元済み：隔離されたファイルの復元が原因で発生しました。 ○ 削除済み：隔離されたファイルの削除が原因で発生しました。
AppSec - ウイルス対策アップローダー	アンチウイルススキャン アプリケーションによって検出されたウイルスの感染源のファイルをアップロードしたログインユーザーの名前。

隔離されたファイル

アンチウイルススキャンによって隔離されたインスタンス内の感染ファイルのリスト：

フィールド	説明
ファイル名	感染ファイルの名前
コンテンツタイプ	ファイル内の感染したコンテンツのタイプ。たとえば、**application/x-dosexec** は感染したアプリケーションまたは DOS の実行可能ファイルですが、**text/plain** は感染した .txt ファイルです。

フィールド	説明
テーブル	感染ファイルを含むテーブルの名前。たとえば、インシデントファイルレコードの場合は [インシデント] が表示されます。
ウィルス	アンチウイルススキャンによって隔離されたファイルの名前
検出	感染ファイルが検出された日時
作成者	感染ファイルを隔離したユーザーの名前
作成日時	隔離ファイルレコードが作成された日時
テーブル Sys ID	隔離ファイルレコードに割り当てられたテーブルシステム識別子

注: [隔離されたファイル] タイルと [ウイルスタイプ] タイルを [イベント] リボンに追加することもできます。詳細については、「セキュリティイベントの監視」と「セキュリティイベントリボンの構成」を参照してください。

関連トピック

- アンチウイルススキャン
- アンチウイルススキャン の構成
- 隔離されたファイルの確認
- ウイルス対策アクティビティの確認
- アナリティクスハブ
- パフォーマンスアナリティクスのブレークダウン
- アナリティクス、インテリジェンス、レポート

MFA メトリクスダッシュボード

MFA メトリクスダッシュボードには、インスタンスのマルチファクター認証構成に関する情報が表示されます。ダッシュボードを使用して、MFA 構成がセキュリティ標準を満たしていることを確認します。

必要な ServiceNow AI Platform ロール

PCI コンプライアンススコアダッシュボードを表示するには、security_dashboard_user または admin が必要です。

MFA メトリクスダッシュボードへのアクセス

ダッシュボードを開くには、次の場所に移動します。 システムセキュリティ > インスタンスセキュリティセンター. Instance Security Center からヘッダーの [メトリクス] をクリックし、**[MFA** メトリクス] を選択します。

ユースケース

組織内のさまざまなユーザーがこのダッシュボードをどのように使用するかの例については、次のユースケースを参照してください。

ユーザー	ダッシュボードの使用法
セキュリティダッシュボードユーザー (security_dashboard_user)	インスタンスセキュリティコンプライアンスを継続的に監視および管理します。

ユーザー	ダッシュボードの使用法
アドミン (admin)	インスタンスセキュリティコンプライアンスを継続的に監視し、セキュリティ脅威を検出して対応します。

インジケーター

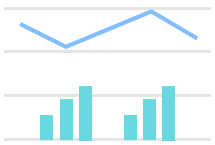
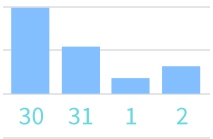
MFA に登録されているユーザー

MFA に登録されているインスタンスのユーザーの合計数を表示します。クリックしてアナリティクスハブを開き、詳細を表示します。

MFA バイパスを使用しているユーザー

MFA バイパスを使用しているユーザーの合計数を表示します。クリックしてアナリティクスハブを開き、詳細を表示します。

データの可視化

タイトル	タイプ	ソーステーブル	説明
高権限 MFA ユーザー	 棒		
MFA ユーザーの傾向	 傾向図		

ISC 仮想エージェントインターフェイスのアクティブ化

admin ロールを持っている場合は、ISC 仮想エージェント会話プラグイン (com.glide.isc_virtualagent) をアクティブ化できます。このプラグインをアクティブ化すると、仮想エージェントおよび 自然言語理解 (NLU) (NLU) コンテンツパックがインストールされ、インスタンスセキュリティセンターから仮想エージェントにアクセスできるようになります。

始める前に

重要:

インスタンスセキュリティセンター (ISC) は 2024 年 9 月の時点で販売が終了しており、サポートもされていないため、新たにアクティブ化することもできません。

ServiceNow セキュリティセンター (SSC) が、今後推奨されるソリューションです。詳細については、「インスタンスセキュリティセンターから ServiceNow セキュリティセンターへの移行」を参照してください。

インスタンスセキュリティセンターの仮想エージェントインターフェイスを利用できるのは、有料の仮想エージェントサブスクリプションに登録し、Glide 仮想エージェント (com.glide.cs.chatbot)

プラグインをアクティブ化しているユーザーに限られます。詳細については、「仮想エージェントの有効化」を参照してください。

必要なロール：admin。

このタスクについて

ISC 仮想エージェントインターフェイスを使用すると、次のタスクを実行できます。

- セキュリティ関連の質問をして、簡単な要約回答と詳細への参照リンクを取得する
- インスタンスセキュリティセンター、プラットフォームセキュリティ、ServiceNow セキュリティポリシー、信頼、ガバナンス、リスク、およびその他のプロセスに関連する回答を取得する
- Now Support Security ポータルでナレッジベースのトピックなどのセキュリティ関連リソースを検索する

ISC 仮想エージェントのプラグイン

プラグイン	説明
ISC 仮想エージェント会話 [com.glide.isc_virtualagent]	Instance Security Center 用の ISC 仮想エージェント会話コンテンツパックをアクティブ化します。
仮想エージェント会話用の ISC NLU モデル [com.glide.isc_nlu]	インスタンスセキュリティセンターの 自然言語理解 (NLU) (NLU) コンテンツパックを有効にします。

注: com.glide.isc_virtualagent プラグインを有効にすると、 com.glide.isc_nlu が自動的にアクティブ化されます。ただし、初めて com.glide.isc_nlu プラグインをアクティブ化する場合は、 com.glide.isc_virtualagent も手動でアクティブ化する必要があります。

手順

1. 移動先 すべて > システムアプリケーション > 利用可能なすべてのアプリケーション > すべて.
2. フィルター基準と検索バーを使用してプラグインを検索します。

   名前または ID でプラグインを検索できます。プラグインが見つからない場合は、ServiceNow 担当者から要求する必要があります。
3. [インストール] を選択して、インストールプロセスを開始します。

   注: ドメインセパレーションと代理アドミンがインスタンスで有効になっている場合、管理ユーザーはグローバルドメインに含まれている必要があります。それ以外の場合、次のエラーが表示されます： 「別の操作が実行されているため、アプリケーションのインストールは利用できません： <プラグイン名> のプラグインの有効化 (Application installation is unavailable because another operation is running: Plugin Activation for <plugin name>)」

   インストールが完了するとメッセージが表示されます。プラグインとともにインストールされるコンポーネントの詳細については、「 アプリケーションとともにインストールされているコンポーネントの検索」を参照してください。

ハードニング設定

ServiceNowセキュリティセンター (SSC) のハードニング設定コンテンツには、ServiceNow AI Platform内のセキュリティ関連のシステムプロパティとプラグインの詳細な説明とコンプライアンス

値が含まれています。セキュリティセンターのハードニング設定アプリを使用して、これらのプロパティを設定できます。

概要と目的

セキュリティセンターは、現在のインスタンスセキュリティ設定がセキュリティセンターのハードニング設定のコンプライアンス値にどの程度準拠しているかに基づいて、毎日のコンプライアンススコアを計算し、パーセンテージで表します。

インスタンスのスコアに影響を与える可能性がある特定のセキュリティ設定を、セキュリティセンターから直接管理できます。

ハードニング設定の構成について、テーブルに示す属性を使用して説明します。

ハードニング設定構成の詳細

構成属性	説明
概要	推奨事項の概要を提供します。
構成名	プロパティまたはプラグイン名。
構成タイプ	システムプロパティ (*sys_properties_list.do*) など、セキュリティセンターの外部でプロパティを構成できる場所について説明します。
データタイプ	構成に必要な値のタイプを説明します。例としては、true/false ブーリアン、インストール、プラグイン、文字列などがあります。
推奨値	インスタンスのセキュリティコンプライアンスを強化するためにセキュリティセンターによって推奨される値。
デフォルト値	ベースシステムで構成が設定されている値。
カテゴリ	ハードニング設定のカテゴリの名前とリンク。
セキュリティリスク	重大度スコア：スコアは、脆弱性が悪用される可能性に応じて、インスタンスに対する潜在的なセキュリティリスクを示します。セキュリティ脆弱性は、**0.0** ～ **10.0** のスケールで CVSS (共通の脆弱性採点システム) スコアを使用して個別に考慮され、採点されます。詳細については、「https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator 」を参照してください。
	CVSS スコアに応じた重大度： • 重大：9.0 ～ 10.0 • 高：7.0 ～ 8.9 • 中：4.0 ～ 6.9 • 低：0.01 ～ 3.9 • なし：0.0

ハードニング設定構成の詳細 (続く)

構成属性	説明
	セキュリティリスクの詳細：設定構成の重要性と、推奨構成を利用しない場合のリスクについて説明します。
依存関係と前提条件	ハードニング構成の前、またはそれともに必要な関連設定または構成。
機能への影響	このハードニング設定がインスタンスの操作に与える影響。
参照	構成ドキュメントまたはその他の役立つ情報へのリンク。

注: 一部の構成は カスタマーサービス & サポート によってのみ完了でき、そのように示されます。

インスタンスが強化要件を満たしていることを確認する方法の詳細については、「セキュリティの強化」を参照してください。

その他のリソース

ユーザーの参照用に、ServiceNow AI Platform には製品ドキュメントの広範な構成機能情報が含まれています。インスタンスの保護 にあるリンクを使用すると、セキュリティコンテンツの大半にアクセスできます。また、以下も参照してください。

- 利用可能なシステムプロパティ
- 一般的なセキュリティ設定のプロパティ
- 高セキュリティ設定

ハードニング設定のベースラインバージョン

ハードニング設定のベースラインバージョンがファミリーおよびストアリリースにどのように対応しているかを確認します。

セキュリティセンター は、インスタンスからシステムプロパティのサブセットを取り込み、その構成の詳細と、アプリ内での非準拠のセキュリティへの影響を表示することで機能します。ベースラインは、Security Center アプリの各リリースで取り込まれるシステムプロパティの参照ポイントとして機能します。

ハードニング設定ベースラインの概要

セキュリティセンターのバージョン	ハードニング設定のベースラインバージョン	サポートされているファミリー	ストアリリース日	デフォルトでインストールされる内容
SSC v1.1	ベースライン v1.0	Utah、Vancouver	2023 年 5 月	Vancouver ファミリー

ハードニング設定ベースラインの概要 (続く)

セキュリティセンターのバージョン	ハードニング設定のベースラインバージョン	サポートされているファミリー	ストアリリース日	デフォルトでインストールされる内容
SSC v1.2	ベースライン v1.0	Utah、Vancouver	2023 年 8 月	ストアのみ
SSC v1.3	ベースライン v2.0	バンクーバー、ワシントン DC	2023 年 11 月	Washington ファミリ
SSC v1.5	ベースライン v4.0	Washington DC、Xanadu	2024 年 5 月	Xanadu
SSC v1.6	ベースライン v4.0	Washington DC、Xanadu	2024 年 8 月	ストアのみ
SSC v 2.0	ベースライン v5.0	ザナドゥ、横浜	2024 年 11 月	Yokohama

新しいハードニング設定

新しいハードニング設定は、ベースラインリリースごとに分類されます。

ベースライン **7.0** の新しいハードニング設定

Security Center ベースラインバージョン 7.0 で、新しいハードニング設定がリリースされました。

- サービスポータルフォームでクロススコープ権限チェックを有効にする (Security Center 7.0 の新機能)
- Glide DB 関数のクエリー ACL を検証する (Security Center 7.0 の新機能)
- ドキュメント分類を使用して、パブリックにアクセス可能なドキュメントを制限する (Security Center 7.0 の新機能)
- システムフィールドへの書き込みアクセスをアドミンユーザーに制限 (Security Center 7.0 の新機能)
- エージェントベースの Office 365 グループ メンバーシップの変更に承認を要求する (Security Center 7.0 の新機能)
- 機密データテーブルとフィールドをデータ生成から除外する (Security Center 7.0 の新機能)
- カタログ変数検索の読み取りロールを強制する (Security Center 7.0 の新機能)
- 有効なクエリ文字列の選択を強制する (Security Center 7.0 の新機能)
- ベアラー承認の場合の制限付きバインディング機能 (Security Center 7.0 の新機能)
- OAuth 2 トークン付与でリソース所有者のパスワード認証情報 (ROPC) を無効にする (Security Center 7.0 の新機能)
- 証明書の信頼を強制する (Security Center 1.3 で更新、 2.0 で削除、7.0 で追加)
- 3DES キーの使用を防止する (Security Center 7.0 の新機能)
- Impact ワークスペースモジュールの説明フィールドで、信頼できるドメインへの HTML リンクを許可する (Security Center 7.0 の新機能)

- コンテキスト検索に未検証のリダイレクトが含まれていないことを確認する (Security Center 7.0 の新機能)
- Impact ワークスペースモジュールの説明フィールドの HTML をサニタイズ (Security Center 7.0 の新機能)

ベースラインバージョン **6.0** の新しいハードニング設定

Security Center ベースラインバージョン 6.0 で、新しいハードニング設定がリリースされました。

- アプリケーションデータに対してのみアプリケーション固有の ACL を適用する
- 従来の JQuery UI の使用を無効にする
- 高リスク UI ページの推奨事項の表示
- ログイン時に現在のパスワードポリシーコンプライアンス要件を強制する
- CSRF 検証をバイパスする警告をユーザーが受け入れるのを防ぐ (Security Center 1.3 および 1.5 で更新)
- パスワードの最小長を設定する (セキュリティセンター 2.2 で更新)
- 高保証セッションのセッション長を制限する
- 廃止された TLS バージョンを無効にする
- シングルサインオン (SSO) が有効になっているユーザーのローカルログインを無効にする
- マルチファクターセットアップで許可されるバイパスを減らす
- 代理操作ユーザーがアプリケーションデータを表示できないようにします
- 詳細な HTTP 要求のログ記録を防止する
- SAML 要求のリレーステータスを有効にして、リプレイ攻撃を防止する
- 高保証セッションのログイン試行失敗を最小限に抑える
- モバイルセッションへの継続的認証ポリシーの適用
- TripleDES/3DES 暗号化アルゴリズムの使用を無効にする

ベースラインバージョン **5.0** の新しいハードニング設定

セキュリティセンターのベースラインバージョン 5.0 で、新しいハードニング設定がリリースされました。

- HR ライフサイクルイベントデータに ACL を強制する (Security Center 2.0 の新機能)
- すべての翻訳済み HTML フィールドをサニタイズ (Security Center 2.0 の新機能)
- サービスポータルウィジェット許可リストを構成する (Security Center 2.0 の新機能)
- HR コアデータに ACL を強制する (Security Center 2.0 の新機能)
- HR 仮想エージェントデータに ACL を強制する (Security Center 2.0 の新機能)
- サービスポータルウィジェットテーブル許可リストを構成する (Security Center 2.0 の新機能)
- サービスアプリケーション情報にセキュリティスコープを適用する (Security Center 2.0 の新機能)
- 空の ACL の作成を防止 (Security Center 2.0 の新機能)
- 仮想エージェント埋め込み Web クライアントへの非認証アクセスを防止します
- トークン認証情報の自動トークンクリーンアップの設定 (Security Center 2.0 の新機能)

- ロールによるグローバルアプリ開発の制限 (Security Center 2.0 の新機能)
- 簡易リストウィジェットでエンコードクエリの ACL を有効にする (Security Center 2.0 の新機能)
- OAuth トークンの有効期限が切れた後にセッションを無効にする (Security Center 2.0 の新機能)
- 許可された MIME 子タイプを設定する (Security Center 2.0 の新機能)
- 代理操作をアドミンに制限する (Security Center 2.0 の新機能)

ベースラインバージョン **4.0** の新しいハードニング設定

セキュリティセンターのベースラインバージョン 4.0 で、新しいハードニング設定がリリースされました。

- 外来の明示的なロールアクセスコントロール条件のレビュー (セキュリティセンター 1.5 で削除)
- サービス組織の作業指示管理クエリルールを有効にする (Security Center 1.5 の新機能、2.0 で更新)
- フローコンテキストの読み取りアクセスを制限する (Security Center 1.5 の新機能)
- ポリシーベースのセッションアクセスのモバイルリフレッシュトークンの間隔を制限する (Security Center 1.5 の新機能)
- モバイルのポリシーベースのセッションアクセスを有効にする (Security Center 1.5 の新機能)
- 非アクティブなユーザーのログインを防ぐ (Security Center 1.5 の新機能)
- イベント管理アサイン先グループアドミンロールの構成 (Security Center 1.5 の新機能)
- HR ケース管理のエージェントワークスペースにセキュリティスコープを適用する (セキュリティセンター 1.5 の新機能、2.0 で更新)
- セキュリティスコープライセンスと許可プレイブックを強制する (Security Center 1.5 の新機能、2.0 で更新)

ベースラインバージョン **2.0** の新しいハードニング設定

セキュリティセンターのベースラインバージョン 2.0 で、新しいハードニング設定がリリースされました。

- アーカイブテーブル ACL のチェックを確認する (Security Center 1.3 の新機能、1.5 で更新)
- アプリケーションスコープの制限を強制する (Security Center 1.3 の新機能、1.5 で削除)
- 強化された Java セキュリティマネージャーを有効にする (Security Center 1.3 の新機能)
- 証明書の失効を検証する (Security Center 1.3 の新機能)
- モバイルアプリケーションのバックグラウンド処理中にペーストボードをクリアすることを必須とする (Security Center 1.3 の新機能、1.5 で更新)
- Protected Tables プラグインの有効化 (Security Center 1.3 の新機能)
- 厳格な特権昇格を強制する (Security Center 1.3 の新機能)
- 統合のアクティブなセッションの有効期間を制限する (Security Center 1.3 の新機能)
- 非アクティブなセッションを積極的に無効化する (Security Center 1.3 の新機能、1.5 および 2.0 で更新)
- MID 監査ログを有効化する (セキュリティセンター 1.3 の新機能、1.5 で更新)
- インポートセット API 内での安全な複数挿入操作の使用 (セキュリティセンター 1.3 の新機能)
- ネットワークエラーに OCSP チェックを強制する (Security Center 1.3 の新機能、2.0 で更新)

- ダッシュボードの共有にセキュリティルールを強制する (セキュリティセンター 1.3 の新機能)
- OAuth パラメーターを POST 本文に制限する (Security Center 1.3 の新機能)
- GraphQL エンドポイントのトレーニングおよび予測フローの添付ファイルサイズの制限 (Security Center 1.3 の新機能、1.5 で更新)
- GlideRecord スコープフェンシングの従来の動作を無効にする (セキュリティセンター 1.3 の新機能、1.5 および 2.0 で更新)
- 認証情報エイリアスの使用を強制する (セキュリティセンター 1.3 の新機能、1.5 で更新)
- 必須の JMS 接続ファクトリ (Security Center 1.3 の新機能、1.5 および 2.0 で更新)
- トレーニングおよび予測フローの添付ファイルサイズの制限 (Security Center 1.3 の新機能、1.5 で更新)
- セッション監査イベントのログ記録 (Security Center 1.3 の新機能、1.5 で更新)
- サービスカタログアイテム追加ページへのアクセスに書き込みアクセスを必須とする (Security Center 1.3 の新機能)
- アクティブセッションタイムアウト例外ロールを定義 (Security Center 1.3 の新機能)
- 証明書ベースの認証が強制されていません (セキュリティセンター 1.3 の新機能)
- Information Request Playbook に対してスコープ付き ACL アクセスを強制する (Security Center 1.3 の新機能、1.5 で更新)
- 記事に対するユーザーコメントを非表示にする (Security Center 1.3 の新機能)
- ダッシュボードの作成/削除にはアクセスチェックが必要であることを確認する (Security Center 1.3 の新機能、2.0 で更新)
- デバイスの暗号化とパスコードの要件を強制する (Security Center 1.3 の新機能)
- AttachmentCreator SOAP Web サービスでファイルの MIME タイプを検証する (Security Center 1.3 の新機能、1.5 で更新)
- 証明書の失効を検証する (Security Center 1.3 の新機能)
- HR アプリの ACL 評価で代理操作をチェックする (セキュリティセンター 1.3 の新機能、1.5 で更新)
- カスタマーサービスアプリケーションでのゲストウォークアップエクスペリエンスに Captcha を必須とする (セキュリティセンター 1.3 の新機能、1.5 で更新)
- イベント管理 HTTP プロセッサで認証を必須とする (Security Center 1.3 の新機能、1.5 で更新、2.0 で削除)
- ゲストのアクティブセッションのライフスパンを制限する (Security Center 1.3 の新機能)
- ターゲットのクローンを禁止する (Security Center 1.3 の新機能)
- SVG ファイルに安全コンテンツセキュリティポリシーを設定する (Security Center 1.3 の新機能)
- CSRF 対策トークンの検証時間 (Security Center 1.3 の新機能)
- ナレッジベースへのアクセスを制限する (Security Center 1.3 の新機能)
- 公共機関デジタルサービスのスコープセキュリティを強制する (Security Center 1.3 の新機能)
- 私用メールからの HR ケースの更新を制限する (セキュリティセンター 1.3 の新機能、1.5 で更新)
- UI のアクティブセッションのライフスパンを制限する (Security Center 1.3 の新機能)
- 安全なリファラーポリシーを強制する (Security Center 1.3 の新機能)

更新されたハードニング設定

更新されたハードニング設定は、ベースラインリリースごとに分割されます。

ベースラインバージョン **6.0** の更新されたハードニング設定

Security Center ベースラインバージョン 6.0 のリリースに伴い、一部のハードニング設定が更新されました。

ドキュメント	更新回数
仮想エージェント埋め込み Web クライアントへの非認証アクセスを防止します	• 新しい簡単な説明:仮想エージェント埋め込み Web クライアントへの非認証アクセスの防止 • 以前の簡単な説明:公開されている仮想エージェント埋め込み Web クライアント sn_va_web_client_app_embed
空の ACL の作成を防止 (Security Center 2.0 の新機能)	ルール: スクリプト:スクリプトが更新され、検出の精度が向上しました。
簡易リストウィジェットでエンコードクエリの ACL を有効にする (Security Center 2.0 の新機能)	• CVSS スコア (新規):4.3 • CVSS スコア (旧):5.3
すべての翻訳済み HTML フィールドをサニタイズ (Security Center 2.0 の新機能)	• CVSS スコア (新規): 4.6 • CVSS スコア (旧):8.8
HTML サニタイザーを有効にする (セキュリティセンター 1.3 で更新)	ルールスクリプト:検出精度を向上させるためにスクリプトが更新されました
X-Frame-Options：SAMEORIGIN セキュリティヘッダーを実装 (セキュリティセンター 1.3 で更新)	• CVSS スコア (新規):5.9 • CVSS スコア (旧):7.1
GlideSystemUserSession スクリプト作成可能 API へのアクセスを制限する (セキュリティセンター 1.3 および 2.0 で更新)	• 修正 (新規):プロパティ *glide.sandbox.usersession.allow_unsanitized_mess* が **false** に設定されていることを確認します。このプロパティのシステムプロパティ [sys_properties] レコードが存在しない場合は作成します。 • 修正 (旧):プロパティ *glide.sandbox.usersession.allow_unsanitized_mess* が **false** に設定されていることを確認します。
ターゲットのクローンを禁止する (Security Center 1.3 の新機能)	• 説明 (新規): *glide.db.clone.allow_clone_target* が推奨値の **false** に設定されていない場合、インスタンスはクローンターゲットとして、またはクローン作成に使用されるインスタンス URL と認証情報を指定するレ

ドキュメント	更新回数
	コードとして使用できます。システムクローンは、データベース内のすべてのものを、あるインスタンスから別のインスタンスにコピーします。クローン作成プロセスでインスタンスデータベースが上書きされて、データが失われたり、データの完全性が失われたりする可能性があるため、これはセキュリティ上のリスクです。修正として、*glide.db.clone.allow_clone_target*が **false** に設定されていることを確認します。このプロパティを推奨値の false に設定しないと、インスタンスをクローンターゲットとして使用できます。クローン作成プロセスでインスタンスデータベースが上書きされる可能性があるため、これはセキュリティ上のリスクです。 • 説明 (旧): *glide.db.clone.allow_clone_target* が推奨値の **false** に設定されていない場合、インスタンスはクローンターゲットとして使用できます。これにより、クローンプロセスによってインスタンスデータベースが上書きされ、整合性と可用性が失われるリスクがあります。
OAuth パラメーターを POST 本文に制限する (Security Center 1.3 の新機能)	• CVSS スコア (新規):4.2 • CVSS スコア (旧):7.4
URL 許可リストのチェックを強制する (セキュリティセンター 1.3、1.5、および 2.0 で更新)	• 説明 (新規): *glide.security.url.whitelist.strict_check* が推奨値の **true** に設定されていない場合、 *glide.security.url.whitelist* が空であるときにすべての外部 URL がリダイレクトに許可されます。*glide.security.url.whitelist*が空でない場合は、ホワイトリスト内の外部 URL のみが許可されます。したがって、*glide.security.url.whitelist.strict_check* を true に設定するか、許可された外部 URL を使用して *glide.security.url.whitelist* が空でない値に設定されるようにすると、インスタンスは安全な状態のままになります。すべての外部 URL がリダイレクトに許可されている場合、攻撃者はユーザーを悪意のある Web サイトにリダイレクトする可能性があります。 • 説明 (旧): *glide.security.url.whitelist.strict_check* が推奨値の **true** に設定されておらず、*glide.security.url.whitelist* が組織

自動翻訳

ドキュメント	更新回数
	の承認済み URL に設定されていない場合、すべての外部 URL がリダイレクトに許可されます。これにより、攻撃者はユーザーを悪意のある Web サイトにリダイレクトする可能性があります。 • CVSS スコア (新規):6.3 • CVSS スコア (旧):8.3
allowlistDisable エンティティ拡張のある XMLdoc2 エンティティ検証を必須とする (セキュリティセンター 1.3 で更新)	• **glide.stax.whitelist_enabled**システムプロパティがシステムプロパティ [sys_properties] テーブルに存在しない場合、または推奨値の **true** に設定されていない場合、**glide.stax.allow_entity_resolution**システムプロパティの値が **true** に設定されていれば、すべての外部エンティティが許可されます。カスタマイズでエンティティの拡張が必要ない場合は、**glide.stax.allow_entity_resolution** システムプロパティを使用して外部エンティティの拡張を無効にします。XML は解析を完了しますが、内部または外部エンティティは含まれません。 ◦ **glide.stax.allow_entity_resolution**を **true** に設定すると、すべての外部エンティティは、**glide.stax.whitelist_enabled** プロパティの設定に従って、対象エンティティの解決または拡張を試みます。 ◦ **glide.stax.allow_entity_resolution**を **false** に設定すると、すべてのエンティティの解決と拡張がブロックされます。このプロパティの詳細については、「XMLDocument2 ストリーミングパーサー内のエンティティ拡張を無効にする (セキュリティセンター 1.5 で更新)」を参照してください。 **glide.stax.whitelist_enabled**が **true** に設定されている場合は、**glide.xml.entity.whitelist** プロパティでカンマ区切りの FQDN のリストを定義します。XML エンティティ処理プロパティを使用して到達できるのは、これらの URL のみです。詳細については、「XML 外部エンティティを制限する (セキュリティセンター 1.3 および 2.0 で更新)」を参照してください。攻撃者はこの脆弱性を利用して、外部エンティティ拡張 (XXE) 攻撃でデータを指数関数的に拡張し、すべてのシステムリソースを短時間で消費する可能性があります。

ドキュメント	更新回数
	• 説明 (旧): *glide.stax.whitelist_enabled* が推奨値の **true** に設定されていない場合、すべての外部エンティティが許可されます。これにより、外部エンティティ拡張 (XXE) 攻撃につながる可能性があります。 • ルールスクリプト:検出精度を向上させるためにスクリプトが更新されました
XML 外部エンティティを制限する (セキュリティセンター 1.3 および 2.0 で更新)	• 許可リストを使用して XXE 攻撃から保護し、サーバーが実行する可能性のある任意の HTTP 要求を攻撃者が含めるのを防ぎます。これにより、サーバーと他のエンティティとの信頼関係を利用した追加の攻撃につながる可能性があります。 **glide.xml.entity.whitelist** システムプロパティの値に http://java.sun.com/j2ee/dtds/ を追加し、**glide.xml.entity.whitelist.enabled** システムプロパティを **true** に設定します。 http://java.sun.com/j2ee/dtds/ 以外の値は、**glide.xml.entity.whitelist** プロパティに含めることができますが、初期設定のプラットフォーム状態には不要です。追加の値を確認して、安全かどうかを判断します。 ⚠ **警告:** これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。 • 説明 (旧):「glide.xml.entity.whitelist」が推奨値の「http://java.sun.com/j2ee/dtds/」に設定されておらず、「glide.xml.entity.whitelist.enabled」が「true」に設定されていない場合、悪意のある外部エンティティが許可され、XXE 攻撃が発生する可能性があります。攻撃者は DTD を使用して、サーバーが実行する可能性のある任意の HTTP リクエストを含めることができます。これにより、サーバーと他のエンティティとの信頼関係を利用した追加の攻撃につながる可能性があります。 • ルールスクリプト:検出精度を向上させるためにスクリプトが更新されました
外部ユーザー登録用の電子メールドメインを制限する (セキュリティセンター 1.3、1.5、および 2.0 で更新)	ルールスクリプト:検出精度を向上させるためにスクリプトが更新されました

自動翻訳

ドキュメント	更新回数
AttachmentCreator SOAP Web サービスでファイルの MIME タイプを検証する (Security Center 1.3 の新機能、1.5 で更新)	• 不適切なファイル拡張子を使用して危険なファイルがインスタンスにアップロードされないように、添付ファイルの MIME タイプが検証されるようにしてください。 **glide.attachment.enforce_security_validation** システムプロパティを **true** に設定します。**true** に設定すると、ファイルは正しいファイルタイプ拡張子でアップロードされます。 • 説明 (旧): 「glide.attachment.enforce_security_validation」が推奨値の「true」に設定されていない場合、添付ファイルの MIME タイプの検証が行われず、間違ったファイル拡張子を使用して危険なファイルがシステムにアップロードされる可能性があります。このプロパティが「true」に設定されている場合、ファイルは正しいファイルタイプ拡張子でアップロードされます。 少なくともMIMEタイプの検証を使用して、ファイルのアップロードを検証することがセキュリティのベストプラクティスです。 • CVSS スコア (新規): 6.7 • CVSS スコア (旧):7.5
許可される ServiceNow 内部 IP アドレスを定義する (セキュリティセンター 1.3 および 1.5 で更新)	• 説明 (新規) **glide.ip.authenticate.strict** および **glide.ip.authenticate.allow.secured** システムプロパティを使用して、より幅広いユーザーグループにインスタンスアクセスが不必要に公開されないようにします。 **glide.ip.authenticate.strict** システムプロパティが **true** に設定されている場合、内部ServiceNow担当者とシステムは、必須 IP 範囲からインスタンスへの受信接続のみを行うことができます。これにより、ServiceNowの可視化がインスタンスの重要な内部インフラストラクチャに制限され、サポートスタッフや営業スタッフなどのより広範な ServiceNow 担当者が企業ネットワークを介してアクセスできなくなります。**glide.ip.authenticate.allow.secured** システムプロパティは、通常の認証済みアクセスと非認証診断ページを含む内部ServiceNow受信接続を許可します。

ドキュメント	更新回数
	true に設定されていない場合は、**glide.ip.authenticate.allow** プロパティで定義されたより広い ServiceNow 内部 IP 範囲を使用して、これらの内部 ServiceNow 受信接続が許可されます。 **glide.ip.authenticate.allow.secured** システムプロパティに信頼できる値のみが含まれていて、プロパティ **glide.ip.authenticate.strict** が **true** に設定されていることを確認してください。 • 説明 (旧):「glide.ip.authenticate.strict」が「true」に設定されている場合、ServiceNow の内部担当者とシステムは、必須 IP 範囲からインスタンスへの受信接続のみを行うことができます。この制限により、ServiceNow は重要な内部インフラストラクチャに対するインスタンスを可視化できなくなり、サポートスタッフや営業スタッフなど、より広範な ServiceNow 担当者が企業ネットワーク経由でアクセスできなくなります。 「true」に設定すると、「glide.ip.authenticate.allow」プロパティを使用して内部 ServiceNow 受信接続が許可されます。「true」に設定されていない場合は、「glide.ip.authenticate.allow」で定義されているより広範な ServiceNow 内部 IP 範囲を使用して、内部 ServiceNow 受信接続が許可されます。
XMLDocument2 ストリーミングパーサー内のエンティティ拡張を無効にする (セキュリティセンター 1.5 で更新)	• 説明 (新): インスタンスのエンティティ拡張を無効にして、システムファイルの読み取り機能やサービス拒否などの攻撃からインスタンスを保護します。システム プロパティを使用して、ストリーミング パーサー (XMLDocument2) による解析中に XML エンティティが展開されないようにします。 インスタンスでエンティティ拡張を無効にするには、 **glide.stax.allow_entity_resolution** システムプロパティを **false** に設定します。このプロパティがシステムプロパティ [sys_properties] テーブルに表示されない場合、デフォルト値は **true** です。プロパティレコードを作成し、値を **false** に設定して値を変更します。 • 説明 (旧): 「glide.stax.allow_entity_resolution」が推

ドキュメント	更新回数
	奨値の「False」に設定されていない場合、このプロパティを使用すると、ストリーミング パーサー (XMLDocument2) による解析中に XML エンティティを展開できます。XML エンティティの拡張は、システム ファイルの読み取り機能やサービス拒否などの攻撃につながる可能性があります。 • 修正 (新規):プロパティ *glide.stax.allow_entity_resolution* がsys_propertiesテーブルに存在し、 **false** に設定されていることを確認します。プロパティがsys_propertiesリストに表示されない場合、デフォルト値は **true** です。 • 修正 (旧):プロパティ「glide.stax.allow_entity_resolution」が「false」に設定されていることを確認します。
空の ACL でデフォルトで拒否する (Security Center 1.3 で更新)	• 説明 (新): リソースに対して ACL が定義されていない場合、またはワイルドカードのテーブルレベルの ACL しかない場合 (incident.* など) に、インスタンスの従来のセキュリティマネージャーがリソースへのアクセスを許可しないようにします。デフォルトでアクセスが許可されている場合、明示的な ACL が設定されていないものはすべて操作の影響を受ける可能性があります。 定義された ACL ルールがない場合、またはワイルドカードのテーブルレベル ACL のみが存在する場合は、 **glide.sm.default_mode** システムプロパティ値を [拒否] に設定してアクセスを禁止します。 • 説明 (旧):「glide.sm.default_mode」が推奨値の「拒否」に設定されていない場合、そのリソースに ACL が定義されていなくても、インスタンスの従来のセキュリティマネージャーはそのリソースへのアクセスを許可します。またはワイルドカードテーブルレベルの ACL のみ。これを「許可」に設定すると、明示的な ACL が設定されていないものはすべて操作の影響を受けやすくなります。 • CVSS スコア (新規):6.3 • CVSS スコア (旧):8.8 • ルールスクリプト:検出精度を向上させるためにスクリプトが更新されました

ドキュメント	更新回数
添付ファイルへの非認証アクセスを制限する	• 説明 (新): 機密情報の漏洩を防ぐために、インスタンス上の画像を保護します。インスタンス上の画像には、末尾 が .iix の URL を使用してアクセスできます。 これらの URL を介して画像にアクセスできないようにするには、**glide.image_provider.security_enabled** システムプロパティを **true** に設定します。 **注:** 元のテーブルが次のいずれかである場合、このプロパティは添付ファイルテーブルの画像では優先されません。 ○ 文房具 [sysevent_email_style] ○ ようこそページセクション [sys_home] ○ システムのプロパティ [sys_properties] 一部の添付ファイルに機密情報が含まれている可能性があるため、非認証ユーザーには制限を適用する必要があります。 • 説明 (旧): 「glide.image_provider.security_enabled」が推奨値の「True」に設定されていない場合、末尾が「.iix」の URL を介してすべての画像にアクセスできます。これにより、画像への認証されていないアクセスが可能になり、機密情報の漏洩につながる可能性があります。
埋め込み HTML コードを無効化する (セキュリティセンター 1.3 で更新)	• 説明 (新): [code] タグを使用して埋め込まれた HTML コードの表示のサポートを無効にします。このタグを使用すると、レンダリングされた HTML をジャーナルフィールドに表示でき、クロスサイトスクリプティング (XSS) 攻撃につながる可能性があります。これらの攻撃により、ログインしているブラウザーのコンテキストで、ユーザーセッションで外部スクリプトが実行される可能性があります。攻撃者はこれらのスクリプトを使用してセッション情報と機密データを盗むことができます。HTML 言語は、スクリプトを書式設定から分離するように設計されていないため、ど

ドキュメント	更新回数
	のシステムでもユーザー制御の HTML を許可することには固有のリスクがあります。 **glide.ui.security.codetag.allow_script**を **false** に設定すると準拠し、このリスクが大幅に軽減されますが、いくつかの小さなリスクが残ります。コードタグのスクリプト部分のみを無効にし、HTML の既知のスクリプトのすべての規則をサニタイズすることに依存します。 **glide.ui.security.allow_codetag** システムプロパティを **false** に設定すると、ジャーナルフィールドとフォームにレンダリングされた HTML が表示されなくなります。 ServiceNow AI Platform は、エスケープおよびエンコード技術を実装することで、多くのインジェクション攻撃とクロスサイト攻撃を軽減します。その結果、ユーザーはジャーナルフィールドに対して HTML 形式の入力の書き込みや送信ができなくなります。しかしジャーナルフィールドでは、コードタグで囲まれたテキストを HTML としてレンダリングできます。 ○ ただし、関連するセキュリティリスクがあります。true に設定すると、悪意のあるユーザーは、ジャーナルフィールドをレンダリングした後に、別のクライアントブラウザーで実行できる有害な HTML JS コードを書き込むことができます。 ○ このプロパティを false に設定すると、[code] タグのサポートを無効にすることで、ジャーナルフィールドで HTML コードがレンダリングされないようにすることができます。 • 説明 (旧):[code] タグを使用して作成された HTML コードの埋め込みのサポートを無効にします。「glide.ui.security.allow_codetag」の値が「false」に設定されていない場合、ジャーナルフィールドとフォームにレンダリングされた HTML は表示されません。埋め込み HTML コードを表示したときに「glide.ui.security.allow_codetag」を「true」に設定すると、クロスサイトスクリプティング (XSS) 攻撃が発生する可能性があります。
パスワードリセットポリシーチェックを有効にする (Security Center 2.0 で更新)	ルールスクリプト:検出精度を向上させるためにスクリプトが更新されました

ドキュメント	更新回数
スパムメールのスコアリングとフィルタリングを有効化する (Security Center 1.3 で更新)	• 技術構成名 (新規):com.glide.email_filter,glide.email.read.active • 技術的な構成名 (旧):com.glide.email_filter • ルールスクリプト:検出精度を向上させるためにスクリプトが更新されました
Excel 計算式をエスケープ (Security Center 1.3 で更新)	• 説明 (新): Excel などのプログラム内の数式をエスケープすることで、ファイルをエクスポートして開いた後に、悪意のある可能性のある数式が実行されないようにします。Excel インジェクションは、Web サイトが Excel ファイル内に信頼できないエントリを埋め込むときに発生します。Microsoft Excel や LibreOffice Call などのスプレッドシートアプリケーションを使用してファイルを開く場合、+、-、=、または @ で始まるセルは、適切にエスケープされない限り、式として解釈されます。悪意のある数式は、スプレッドシートに機密情報が含まれていない場合でも、コード実行を通じて閲覧者のコンピューターを侵害するために使用される可能性があるため、リスクをもたらします。 **glide.export.escape_formulas** システムプロパティを **true** に設定して、これらの数式が実行されないようにします。 • 説明 (旧):プロパティ「glide.export.escape_formulas」を推奨値の「true」に設定すると、Excel などのプログラムで潜在的に悪意のある式が、ファイルをエクスポートして開いた後に実行されるのを防ぐことができます。CSV、Xls、および XLSX のセル値は、スプレッドシートアプリケーションによって式として解釈される可能性があり、適切にエスケープされない限り、悪意のあるコードが実行される可能性があります。 • CVSS スコア (新規):6.4 • CVSS スコア (旧):6.5
JSONP 要求を信頼できる URL に制限する (Security Center 1.3 で更新)	• 説明 (新): AngularJS $httpサービスの信頼できる URL のみが JSONP 要求を許可/拒否できるようにすることで、インスタンスのセキュリティを強化します。これらのプロパティが構成されて有効になっていない場合、任意の URL に対して JSONP 要求が許可されます。

自動翻訳

ドキュメント	更新回数
	angular.jsonp.inclusion_list.urls システムプロパティの値を使用して、信頼され、この目的が許可されている URL のリストを定義します。許可された JSONP を **angular.jsonp.inclusion_list.urls** にリストされている URL のみに制限するには、**angular.jsonp.inclusion_list.enabled** システムプロパティの値を **true** に設定します。 • 説明 (旧):このプロパティは、JSONP 要求を許可/拒否する angularJS $http サービスの信頼できる URL を指定します。プロパティが必要です。これは顧客にとって破壊的変更となる可能性があるため、信頼できる URL を追加する方法が必要です。「angular.jsonp.inclusion_list.enabled」が推奨値の「true」に設定されていない場合、JSONP 要求は任意の URL に対して許可されます。
SNC アクセスコントロールプラグインを有効化する (セキュリティセンター 1.3 で更新)	• 説明 (新): SNC アクセス制御 (com.snc.snc_access_control) プラグインを有効にして ServiceNow カスタマーサービスとサポート担当者が明示的な許可なくインスタンスにアクセスできないようにします。インスタンスへのアクセスはすべて監査されますが、このアクセスを制御することもできます。このアクセス方法は監査可能および追跡済み。 **注:** 製品をサポートおよび管理する能力を持つその他の認定 ServiceNow 運用担当者は、基盤となるインフラストラクチャで管理アクションを実行する必要があります。このプラグインを有効にすると、サポートサービスレベルと可用性 SLA に影響する場合があります。可用性 SLA は、サポートスタッフ担当者がインスタンスへのアクセスを許可された時間から測定されます。 SNC アクセス制御 (com.snc.snc_access_control) プラグインを有効にして、明示的な権限なしでインスタンスへのアクセスを制限します。この機能の詳細については、「ServiceNow アクセス制御」を参照してください。アクティブ化情報については、次を参照してください。ServiceNowアクセス制御を有効にする

ドキュメント	更新回数
	• 説明 (旧):SNC アクセス制御 (com.snc.snc_access_control) プラグインにより、カスタマーサービスとサポート担当者が明示的な権限なしにインスタンスにアクセスできなくなります。ただし、製品をサポートおよび管理する能力を持つその他の認定された ServiceNow Operations 担当者は、基盤となるインフラストラクチャで管理アクションを実行する必要があります。このインフラストラクチャには、SaaS を構成する他のインフラストラクチャコンポーネントの中でも、サーバーとデータベースが含まれます。このアクセス方法は完全に監査可能であり、追跡されます。このプラグインを使用すると、明示的な権限なしでインスタンスへのアクセスを制限できるため、サポートサービスレベルと可用性 SLA に影響を与える可能性があります。可用性 SLA は、サポートスタッフ担当者がインスタンスへのアクセスを許可された時間から測定されます。 • 修正 (新規):プラグイン「com.snc.snc_access_control」がアクティブになっていることを確認します。https://www.servicenow.com/docs/csh?topicname=t_ActivateSNCAccessControl.html&version=lat でのアクティブ化に関するドキュメントをお読みください 。 • 修正 (旧):プラグイン「com.snc.snc_access_control」がアクティブになっていることを確認します。 • CVSS スコア (新規):3.3 • CVSS スコア (旧):8.2
失敗したログインのロック解除タイムアウト期間を最大化する (セキュリティセンター 1.3 で更新)	• 技術構成名 (新規): glide.user.unlock_timeout_in_mins、sysevent_script_action • 技術的な構成名 (旧):glide.user.unlock_timeout_in_mins • 説明 (新): ロックアウトされた後にユーザーがログインできない期間を定義することで、総当たり攻撃からインスタンスを保護します。**glide.user.unlock_timeout_in_mins** システムプロパティは、その値で指定された期間後にユーザーアカウントのロックを解除します。値が指定されていない場合、インスタンスはデフォルトの 15 分後にユーザーアカウントのロックを解除します。 • 説明 (旧):「glide.user.unlock_timeout_in_mins」が推

ドキュメント	更新回数
	奨値の「15」に設定されていない場合は、より短い期間にアカウントの総当たり攻撃をする方が簡単かもしれません。このプロパティは、glide.user.unlock_timeout_in_mins プロパティに指定された期間後にユーザーアカウントのロックを解除します。値を指定しない場合は、デフォルトの 15 分後にユーザーアカウントのロックが解除されます。 • 修正 (新規): **glide.user.unlock_timeout_in_mins**システムプロパティ値を最小の **15** に設定します。**glide.user.unlock_timeout_in_mins**存在しない場合、デフォルトのロックアウト時間は 15 分に設定されます。 **SNC** ユーザーロックアウトチェックと自動ロック解除スクリプトアクション (スクリプトアクション [sysevent_script_action] テーブルにあります) が存在し、アクティブであることを確認します。**[SNC** ユーザーロックアウトチェック (自動ロック解除あり)] スクリプトアクションは、高セキュリティ設定 (com.glide.high_security) プラグインとともにインストールされます。 • 修正 (旧):プロパティ「glide.user.unlock_timeout_in_mins」が「15」以上に設定されていることを確認します。 • ルールスクリプト:検出精度を向上させるためにスクリプトが更新されました
特定の IP 範囲プラグインへのアクセスを制限する (Security Center 1.3 で更新)	• 技術的な構成名 (新規):com.snc.ipauthenticator,ip_access • 技術構成名 (旧):com.snc.ipauthenticator
ターゲットテーブルが空のメールへのアクセスを制限する	• CVSS スコア (新規):6.5 • CVSS スコア (旧):5.4
ダウンロード可能な MIME タイプを制限する (セキュリティセンター 1.3 および 2.0 で更新)	• CVSS スコア (新規):6.4 • CVSS スコア (旧):8
MultiSSO プラグインの更新バージョンを有効にする (Security Center 1.3 および 1.5 で更新)	• 技術的な構成名 (新規):glide.authenticate.multissov2_feature.enabled • 技術構成名 (旧):glide.authenticate.multisso.enabled,glide.authenticate.

自動翻訳

ドキュメント	更新回数
	• 説明 (新): インスタンスでマルチ SSO プラグインが有効になっている場合は、v2 バージョンが有効になっていることを確認して、セキュリティの脆弱性を軽減します。最新バージョンではセキュリティが強化され、アサーション暗号化のサポートや IDP によって開始されるシングルログアウト (SLO) など、より多くの機能が追加されています。最新バージョンが有効になっていない場合、新しいセキュリティ機能は使用できず、インスタンスは廃止されたプラグインを使用するリスクがあります。 KB0756504 の手順に従って、最新バージョンにアップグレードします。このプロセスには、カスタマイズ関連の変更の確認と移行、そしてバージョンのアップグレードが含まれます。完了すると、**glide.authenticate.multissov2_feature.enabled** システムプロパティは自動的に **true** に設定されます。 • 説明 (旧):インスタンスでマルチ SSO プラグインが有効になっている場合は、v2 バージョンを有効にする必要があります。SAML 1.1 および SAML 2.0 を含む MultiSSOv2 より前のバージョンは、ベストプラクティスに従わず、既知の CVE を持つ opensaml ライブラリのバージョンを使用します。既知のCVEが古いopensamlライブラリで悪用された場合、攻撃者はメッセージを偽造し、XML署名ラッピング攻撃によって認証をバイパスしたり、エンティティになりすましたり、中間者攻撃者がプラットフォームに不正にアクセスしたりする可能性があります。 • CVSS スコア (新規):0 • CVSS スコア (旧):7.1
非アクティブなユーザーのログインを防ぐ (Security Center 1.5 の新機能)	ルールスクリプト:検出精度を向上させるためにスクリプトが更新されました
MID 監査ログを有効化する (セキュリティセンター 1.3 の新機能、1.5 で更新)	ルールスクリプト:検出精度を向上させるためにスクリプトが更新されました
アーカイブテーブル ACL のチェックを確認する (Security Center 1.3 の新機能、1.5 で更新)	ルールスクリプト:検出精度を向上させるためにスクリプトが更新されました
アクティブセッションタイムアウト例外ロールを定義 (Security Center 1.3 の新機能)	• CVSS スコア (新規):6.4 • CVSS スコア (旧):7.1

自動翻訳

ドキュメント	更新回数
HTTP 応答の本文サイズを制限する (セキュリティセンター 1.3 の新機能、1.5 で更新)	• 説明 (新): **glide.http.response.get_body.limit.enabled** および **glide.http.response.get_body.limit** システムプロパティを使用して、要求応答本文が大きすぎることが原因で発生する可能性のある OutOfMemoryExceptions を防止します。これらの例外により、サービス拒否 (DoS) 攻撃や、攻撃者がインスタンスを侵害するのに役立つその他の問題が発生する可能性があります。これらのプロパティを推奨値に設定しないと、インスタンスが OutOfMemoryExceptions やサービス拒否攻撃に対して脆弱になる可能性があります。 次のセキュリティの脆弱性からインスタンスを保護するには、次の手順を実行します。 ○ **glide.http.response.get_body.limit.enabled** システムプロパティを **true** に設定します。 ○ **glide.http.response.get_body.limit** システムプロパティが 524,288,000 メガバイト (500 MB) 以下に設定されていることを確認します。 • 説明 (旧):プロパティ glide.http.response.get_body.limit.enabled および glide.http.response.get_body.limit は、要求応答本文が大きすぎるためにメモリ不足例外がスローされるのを防ぐ新機能を有効にするために導入されました。メモリ不足の例外は、サービス拒否攻撃やその他の問題を引き起こし、攻撃者がインスタンスを侵害するのを助ける可能性があります。 • CVSS スコア (新規):3.1 • CVSS スコア (旧):6.4
UI のアクティブセッションのライフスパンを制限する (Security Center 1.3 の新機能)	• 説明 (新): アクティブな HTTP セッションの有効期間を短くすることで、潜在的なセキュリティインシデントの範囲を縮小します。**glide.ui.active.session.life_span** システムプロパティは、非アクティブタイムアウトに関係なく、アクティブな HTTP セッションに最大有効期間を適用します。最大有効期間が長いほど、攻撃者が盗んだセッションを長時間使用できるようになり、セキュリティインシデントの範囲が拡大します。デフォルト値の **0** は、アクティブセッションのタイムアウトを無効にします

自動翻訳

ドキュメント	更新回数
	glide.ui.active.session.life_spanを 1 ～ 720 の値に設定します。この値は、HTTP セッションをアクティブにしておくことができる時間 (分単位) を表します。 • 説明 (旧):この構成では、非アクティブなタイムアウトに関係なく、アクティブなゲスト HTTP セッションに最大有効期間が適用されます。構成値は分単位で、値が 0 の場合はアクティブセッションのタイムアウトが無効になります。最大有効期間が長いほど、攻撃者が盗んだセッションをより長く保持できるので、セキュリティインシデントの範囲が拡大します。この特定のプロパティは、UI セッションタイムアウトに制限されます。

ベースラインバージョン **5.0** の更新されたハードニング設定

一部のハードニング設定は、セキュリティセンターベースラインバージョン 5.0 のリリースで更新されました。

ベースラインバージョン 5 では、レコード全体のスタイルと一貫性のために、簡単な説明がいくつか更新されています。さらに、プロパティがsys_propertyテーブルから削除された場合のデフォルト値の精度を向上させるために、多くのプロパティ関連スクリプトが更新されました。

ドキュメント	更新回数
SOAP 要求に認証を必須とする (Security Center 1.3、1.5、および 2.0 で更新)	• 新しい修正:Glide プロパティ *glide.basicauth.required.soap* が存在し、値が true に設定されていることを確認します。または、プロパティ glide.soap.require_ws_security を true に設定し、製品ドキュメントに従って WS セキュリティプロファイルを構成することで、WS セキュリティのインスタンスを構成します。プロパティがsys_propertiesテーブルに含まれていない場合は、新しいレコードを追加してください。 • 以前の修正:プロパティ *glide.basicauth.required.soap* 値が true に設定されていることを確認します。または、プロパティ *glide.soap.require_ws_security* を true に設定し、製品ドキュメントに従って WS セキュリティプロファイルを構成することで、WS セキュリティのインスタンスを構成します。
ネットワークエラーに OCSP チェックを強制する (Security Center 1.3 の新機能、2.0 で更新)	• 新しい修正:プロパティ *com.glide.communications.httpclient.ocsp_allow_* が存在し、false に設定されていることを確認

ドキュメント	更新回数
	します。プロパティがsys_propertiesテーブルに含まれていない場合は、新しいレコードを追加してください。 • 以前の修正:プロパティ *com.glide.communications.httpclient.ocsp_allow_* が false に設定されていることを確認します。
外部コンテンツ URL を無効にする (Security Center 2.0 で更新)	• 新しい修正:Glide プロパティ *glide.ui.url.external.content* が存在し、値が false に設定されていることを確認します。プロパティがsys_propertiesテーブルに含まれていない場合は、新しいレコードを追加してください。 • 以前の修正:プロパティ *glide.ui.url.external.content* が false に設定されていることを確認します。 • 新しい CVSS スコア:7.2 • 以前の CVSS スコア:8.1 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
XML 外部エンティティを制限する (セキュリティセンター 1.3 および 2.0 で更新)	• 新しい修正:Glide プロパティ *glide.xml.entity.whitelist* が存在し「http://java.sun.com/j2ee/dtds/」に設定されていること、および Glide プロパティ *glide.xml.entity.whitelist.enabled* が存在し、値が true に設定されていることを確認します。プロパティがsys_propertiesテーブルに含まれていない場合は、新しいレコードを追加してください。 • 以前の修正:プロパティ *glide.xml.entity.whitelist* が「http://java.sun.com/j2ee/dtds/」に設定され、プロパティ *glide.xml.entity.whitelist.enabled* が true に設定されていることを確認します。
認証されていない公開レポートを無効にする (Security Center 2.0 で更新)	• 新しい修正:Glide プロパティ *glide.report.published_reports.enabled* が存在し、値が false に設定されていることを確認します。プロパティがsys_propertiesテーブルに含まれていない場合は、新しいレコードを追加してください。

自動翻訳

ドキュメント	更新回数
	• 以前の修正:プロパティ *glide.report.published_reports.enabled* が false に設定されていることを確認します。
パスワードリセットポリシーチェックを有効にする (Security Center 2.0 で更新)	• 新しい修正:Glide プロパティ *glide.enable.password_policy* が存在し、値が true に設定されていることを確認します。プロパティがsys_propertiesテーブルに含まれていない場合は、新しいレコードを追加してください。 • 以前の修正:プロパティ *glide.enable.password_policy* が true に設定されていることを確認します。
GlideXMLUtil スクリプト可能項目のエンティティ拡張しきい値を最小化する (セキュリティセンター 1.3、1.5、および 2.0 で更新)	• 新しい修正:プロパティ *glide.xmlutil.max_entity_expansion* が 3000 以下に設定されていることを確認します。インスタンスが Washington 以降にある場合、sys_propertiesレコードが存在しない場合、デフォルトの暗黙的な値は 3000 です。インスタンスが Washington 以降にない場合、インスタンスアドミンは、名前が *glide.xmlutil.max_entity_expansion* 、値が 3000 のsys_propertiesレコードを作成することをお勧めします。 • 以前の修正:プロパティの *glide.xmlutil.max_entity_expansion* が 3000 以下に設定されていることを確認します。
発信 SSLv2/SSLv3 接続を無効化する (Security Center 1.3 で更新)	• 新しい修正:Glide プロパティ *glide.outbound.sslv3.disabled* が存在し、値が true に設定されていることを確認します。プロパティがsys_propertiesテーブルに含まれていない場合は、新しいレコードを追加してください。 • 以前の修正:プロパティ *glide.outbound.sslv3.disabled* が true に設定されていることを確認します。 **重要:** *glide.outbound.sslv3.disabled*プロパティの値は安全な上書きであり、一度変更すると変更できません。

自動翻訳

ドキュメント	更新回数
GlideRecord スコープフェンシングの従来の動作を無効にする (セキュリティセンター 1.3 の新機能、1.5 および 2.0 で更新)	• 新しい簡単な説明:GlideRecord スコープフェンシングの従来の動作を無効にする • 以前の簡単な説明:GlideRecord スコープフェンシングの従来の動作を有効にする
アップロードされる MIME タイプを制限する (セキュリティセンター 1.3 および 2.0 で更新)	• 新しい修正:プロパティ *glide.security.file.mime_type.validation* が存在し、true に設定されていることを確認します。プロパティがsys_propertiesテーブルに含まれていない場合は、新しいレコードを追加してください。 • 以前の修正:プロパティ *glide.security.file.mime_type.validation* が true に設定されていることを確認します。
入れ子になった式で Jelly JS 補間保護を有効にする (Security Center 2.0 で更新)	• 新しい修正:Glide プロパティ *glide.ui.jelly.js_interpolation.protect_nested_e* が存在し、値が true に設定されていることを確認します。プロパティがsys_propertiesテーブルに含まれていない場合は、新しいレコードを追加してください。 • 以前の修正:プロパティ *glide.ui.jelly.js_interpolation.protect_nested_e* が true に設定されていることを確認します。
LDAP 認証で SSL を有効にする (Security Center 1.5 および 2.0 で更新)	ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
UserCookie バージョン 3.1 を有効にする (Security Center 2.0 で更新)	• 新しい説明:UserCookie v3 は、property *glide.ui.secure.cookies.use_kmf is disabled* の場合にのみ生成されます。UserCookie v3 は、HMAC の秘密鍵をソースコードに保存し、すべての顧客に対して同一であるため、安全ではありません。これにより、悪意のあるアクターがこの1つの秘密鍵を使用してユーザーセッションを乗っ取ることができます。プロパティ *glide.ui.secure.cookies.use_kmf* を true に設定すると、UserCookie v3.1 が使用され、秘密鍵が KMF などのセキュリティストレージに保存されます。 • 以前の説明:UserCookie v3 は、プロパティ *glide.ui.secure.cookies.use_kmf* が無効になっている場合にのみ生成されます。UserCookie v3 は、HMAC の秘密鍵をソースコードに保存し、すべての顧客に対して同一であるため、安全ではありません。これにより、悪意のあるアクターがこの1つの秘

ドキュメント	更新回数
	密鍵を使用してユーザーセッションを乗っ取ることができます。 • 新しい修正:プロパティ *glide.ui.secure.cookies.use_kmf* が存在し、true に設定されていることを確認します。プロパティがsys_propertiesテーブルに含まれていない場合は、新しいレコードを追加してください。 • 以前の修正:プロパティ *glide.ui.secure.cookies.use_kmf* が true に設定されていることを確認します。つまり、UserCookie v3.1 が使用され、秘密鍵は KMF などのセキュリティストレージに保存されます。
パスワードリセットの OTP 期限を 1 時間に設定 (Security Center 2.0 で更新)	ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
ユーザーの代理操作をログ記録 (Security Center 1.3 および 2.0 で更新)	• 新しい修正:プロパティ *glide.sys.log_impersonation* が存在し、true に設定されていることを確認します。プロパティがsys_propertiesテーブルに含まれていない場合は、新しいレコードを追加してください。 • 以前の修正:プロパティ *glide.sys.log_impersonation* が true に設定されていることを確認します。
必須の JMS 接続ファクトリ (Security Center 1.3 の新機能、1.5 および 2.0 で更新)	ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
ダッシュボードの作成/削除にはアクセスチェックが必要であることを確認する (Security Center 1.3 の新機能、2.0 で更新)	• 新しい修正:Glide プロパティ *glide.processors.check_access_before_process* が存在し、値が true に設定されていることを確認します。プロパティがsys_propertiesテーブルに含まれていない場合は、新しいレコードを追加してください。 • 以前の修正: *glide.processors.check_access_before_process* の値が常に true であることを確認します。
非アクティブなセッションを積極的に無効化する (Security Center 1.3 の新機能、1.5 および 2.0 で更新)	• 新しい修正:Glide プロパティ *glide.active.session.timeout.invalidate.session* が存在し、値が true に設定されていることを確認します。プロパティがsys_properties

ドキュメント	更新回数
	テーブルに含まれていない場合は、新しいレコードを追加してください。 • 以前の修正:Glide プロパティ *glide.active.session.timeout.invalidate.session* を true に設定します。
HR ケース管理のエージェントワークスペースにセキュリティスコープを適用する (セキュリティセンター 1.5 の新機能、2.0 で更新)	ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
セキュリティスコープライセンスと許可プレイブックを強制する (Security Center 1.5 の新機能、2.0 で更新)	ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
ダウンロード可能な MIME タイプを制限する (セキュリティセンター 1.3 および 2.0 で更新)	• 新しい説明: プロパティ *glide.ui.attachment.force_download_all_mime_type* が true に設定されている場合、*glide.ui.attachment.download_mime_types* プロパティは上書きされ、ブラウザーによってレンダリングされるのではなく、すべての MIME タイプがダウンロードされます。たとえば、text/html をダウンロードすると、HTML ファイルはブラウザーでインライン表示されるのではなく、ファイルとしてクライアントに強制的にダウンロードされ、XSS 攻撃を防ぐことができます。XSS により、admin などの上位ロールへの特権エスカレーションが容易になり、横方向の移動が可能になる可能性があります。 • 以前の説明:プロパティ *glide.ui.attachment.force_download_all_mime_type* が true に設定されていない場合、*glide.ui.attachment.download_mime_types* プロパティは上書きされ、すべての MIME タイプがブラウザによってレンダリングされるのではなくダウンロードされます。たとえば、text/html をダウンロードすると、HTML ファイルはブラウザーでインライン表示されるのではなく、ファイルとしてクライアントに強制的にダウンロードされ、XSS 攻撃を防ぐことができます。XSS 機能により、admin などの上位ロールへの特権エスカレーションが容易になり、横方向の移動が容易になる可能性があります。 • 新しい修正:プロパティ *glide.ui.attachment.force_download_all_mime_type* が true に設定されていることを確認します。sys_propertiesテーブルにプロパティが存在しない場合、デフォルト値は false です。

ドキュメント	更新回数
	• 以前の修正:プロパティ *glide.ui.attachment.force_download_all_mime_type* が true に設定されていることを確認します。 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
制限されたダウンロード可能な MIME タイプを定義する (セキュリティセンター 1.3、1.5、および 2.0 で更新)	ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
感染したファイルのダウンロードを許可しない (セキュリティセンター 1.5 および 2.0 で更新)	• 新しい説明:プロパティ *com.glide.snap.infected_download_allowed* が true に設定されている場合でも、ウイルス対策サービスが停止しているか到達不能の場合でも、ユーザーはスキャンされていない添付ファイルをダウンロードできます。これは、ユーザーが悪意のあるファイルをダウンロードし、ユーザーのデスクトップを感染させるリスクがあることを意味します(デバイスに他のエンドポイント保護がない場合)。 • 以前の説明:*com.glide.snap.infected_download_allowed* が推奨値のFalseに設定されていない場合、スキャンされていない悪意のあるファイルをダウンロードする可能性があり、ユーザーのデスクトップに感染するリスクがあります。 • 新しい修正:プロパティ *com.glide.snap.infected_download_allowed* が false に設定されていることを確認します。 • 以前の修正:プロパティ *com.glide.snap.infected_download_allowed* が False に設定されていることを確認します。
GlideSystemUserSession スクリプト作成可能 API へのアクセスを制限する (セキュリティセンター 1.3 および 2.0 で更新)	• 新しい説明:*gs.addErrorMessageNoSanitizationMessaging()* および *gs.addInfoMessageNoSanitization()* は、スクリプト環境内でログ記録と通知に使用されます。どちらも、このプロパティが推奨値の false に設定されていない場合にサンドボックスで使用できます。サンドボックスは、非認証ユーザーでロールのないユーザーが利用できる、権限の低いスクリプト環境です。これらのメソッドはどちらも、サニタイズされていない入力をユーザーに表示するために使用できます。サニタイズされていない入力には、ユーザーのブラウザで実行される危険なコードが含まれている可能性があるため、サニタイズされていない入力をユーザー

ドキュメント	更新回数
	に表示することは危険です。これは、従来の反射型XSS攻撃に利用される可能性があります。反射型 XSS 攻撃は、セッションハイジャックを含む複数のシナリオで使用される可能性があります。 • 以前の説明:Glide スクリプトサンドボックス内のメッセージングは、ログ記録目的で使用されます。このサニタイズされていないエラー関数を呼び出すと、プラットフォームは反射型 XSS 攻撃にさらされます。XSS攻撃は、誰かのセッションCookieを盗むことで、簡単に権限昇格を可能にする可能性があります。`glide.sandbox.usersession.allow_unsanitized_` 推奨値の false に設定されていない場合、サニタイズされていないエラーメッセージ関数 `addErrorMessageNoSainitization` および `addInfoMessageNoSainitization` をスクリプトで使用できます。
サービス組織の作業指示管理クエリルールを有効にする (Security Center 1.5 の新機能、2.0 で更新)	• 新しい説明:true に設定すると、sn_query_ruleテーブルのルール/フィルターを使用して、クエリビジネスルールと読み取り ACL を介して、ログインしたユーザーに対するフィールドサービス管理 (FSM) 関連テーブル (作業指示書および作業指示タスク) への読み取りアクセス権が決定されます。false に設定されている場合、レコードはクエリルールに基づいてフィルタリングされません。クエリービジネスルールは、セキュリティ検証を追加します。具体的には、このプロパティは、アサインされたテリトリーまたはテリトリーメンバーシップに基づいて、エージェント、認定者、およびディスパッチャーのレコードをフィルタリングします。レコードを読み取るときは、最小特権の原則に従うことがベストプラクティスです。このプロパティが true に設定されていない場合、フィールドサービス管理 (FSM) テーブルからデータが漏洩するリスクが高まる可能性があります。 • 以前の説明:true に設定すると、sn_query_ruleテーブルのルール/フィルターを使用して、クエリビジネスルールと読み取り ACL を介して、ログインしたユーザーに対するフィールドサービス管理 (FSM) 関連テーブル (作業指示書および作業指示タスク) への読み取りアクセス権が決定されます。false に設定されている場合、レコードはクエリルールに基づいてフィルタリングされません。クエリービジネスルールは、セキュリティ検証を追加します。具体的には、このプロパティは、アサインされたテリトリーまたはテリトリーメンバーシップに基づいて、

ドキュメント	更新回数
	エージェント、認定者、およびディスパッチャーのレコードをフィルタリングします。レコードを読み取るときは、最小特権の原則に従うことがベストプラクティスです。
外部ユーザー登録用の電子メールドメインを制限する (セキュリティセンター 1.3、1.5、および 2.0 で更新)	• 新しい説明: *sn_ext_usr_reg.allowed_email_domains* プロパティは、ServiceNow インスタンスへの自己登録が許可されるメールアドレスを定義します。形式は、example@domain2.com などのメールが受け入れられる domain1.com,domain2.com などの受け入れ可能なメールドメインのカンマ区切りリストにする必要があります。受け入れ可能なドメインのリストで *sn_ext_usr_reg.allowed_email_domains* が設定されていない場合、メールアドレスを持つユーザーはインスタンスにアカウントを登録できます。定義されていない場合、悪意のある攻撃者が、インスタンスへの認証されたアクセスを取得するために、望ましくないドメインのメールアドレスを使用して登録を実行する可能性があります。 • 以前の説明: *sn_ext_usr_reg.allowed_email_domains* プロパティは、ServiceNow インスタンスへの自己登録が許可されるメールアドレスを定義します。受け入れ可能なドメインのリストで *sn_ext_usr_reg.allowed_email_domains* が設定されていない場合、メールアドレスを持つユーザーはインスタンスにアカウントを登録できます。定義されていない場合、悪意のある攻撃者が、インスタンスへの認証されたアクセスを取得するために、望ましくないドメインのメールアドレスを使用して登録を実行する可能性があります。
ドット連結フィールドにドメインセパレーションを適用する (セキュリティセンター 1.3、1.5、および 2.0 で更新)	• 新しい説明:このプロパティは、ドット連結フィールドにドメインセパレーション機能を確実に適用するために、結合クエリにドメインセパレーション条件を指定するかどうかを制御します。ドメインセパレーションを使用しているインスタンスで *glide.sys.domain.include_domain_condition_on_jo* が推奨値である true に設定されていない場合、特定のドメインと共有されない機密情報が公開される可能性があります。コンポーネントが安全でないクロスドメインクエリに依存している場合、インスタンスへの機能への影響は中程度になる可能性があります。イン

ドキュメント	更新回数
	スタンスは、有効にする前に準本番環境でテストする必要があります。 • 以前の説明:このプロパティは、ドット連結フィールドにドメインセパレーション機能を確実に適用するために、結合クエリにドメインセパレーション条件を指定するかどうかを制御します。ドメインセパレーションを使用しているインスタンスで *glide.sys.domain.include_domain_condition_on_jo* が推奨値である true に設定されていない場合、特定のドメインと共有されない機密情報が公開される可能性があります。
URL 許可リストのチェックを強制する (セキュリティセンター 1.3、1.5、および 2.0 で更新)	• 新しい修正:プロパティ *glide.security.url.whitelist.strict_check* が true に設定されているか、プロパティ *glide.security.url.whitelist* が値に設定されていることを確認します。 • 以前の修正:プロパティ *glide.security.url.whitelist.strict_check* が「true」に設定され、プロパティ *glide.security.url.whitelist* が値に設定されていることを確認します。
SOAP 要求のゲストユーザーを設定する (Security Center 1.3 および 2.0 で更新)	ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
バックグラウンドスクリプトへのアクセスを制限する (Security Center 1.3 および 2.0 で更新)	• 新しい説明:このプロパティは、[スクリプトバックグラウンド] モジュールにアクセスするために必要なロールを保持します。*glide.script_processor.admin* が推奨値とデフォルト値の admin に設定されていない場合、低い特権ロールを持つユーザーはインスタンスでバックグラウンドスクリプトを実行できます。これにより、ACL システムが完全にバイパスされ、テーブルへのフルアクセスが可能になります。 • 以前の説明:このプロパティは、[スクリプトバックグラウンド] モジュールにアクセスするために必要なロールを保持します。*glide.script_processor.admin*が推奨値の admin、security_admin、または maint に設定されていない場合、低い特権ロールを持つユーザーはインスタンスでバックグラウンドスクリプトを実行できます。これにより、ACL システムが完全にバイパスされ、テーブルへのフルアクセスが可能になります。 • 新しい修正:プロパティ *glide.script_processor.admin* がアド

自動翻訳

ドキュメント	更新回数
	ミンに設定されていることを確認します。これはインスタンスのデフォルト値です。 • 以前の修正:プロパティ *glide.script_processor.admin* が admin、security_admin、または maint ロールに設定されていることを確認します。
証明書チェーンとホスト名の検証 (Security Center 1.3 の新機能、2.0 で更新)	• 新しい説明:Glide プロパティ *com.glide.communications.httpclient.verify_hostn* が安全な値 true に設定されていない場合、ServiceNow インスタンスから開始された TLS 接続中にリモートホストによって提示されたホスト名と証明書チェーンは検証されません。これにより、TLS 接続のセキュリティが侵害され、2 者間の通信が傍受される中間者攻撃が可能になります。これにより、機密データが開示される可能性があります。 • 以前の説明: *com.glide.communications.httpclient.verify_hostn* が true に設定されていない場合、2 者間の通信が傍受される中間者攻撃が可能になります。このプロパティを安全でない値に設定すると、失効ステータスの確認によって証明書チェーン内のすべての証明書を評価する証明書検証プロセスが無効になります。http クライアントが潜在的に有害なホスト名に接続しないようにするには、このプロパティを true に設定します。
無効なパスワードリセット試行に対するロックアウト時間を制御する (Security Center 1.3 および 2.0 で更新)	• 新しい簡単な説明: *Control Lockout Time for Invalid Password Reset Attempts* • 以前の簡単な説明: *Minimize Reset Password Request Max Attempts Window Duration* • 新しい説明: *password_reset.request.max_attempt_window* プロパティは、*password_reset.request.max_attempt property*で設定された最大試行失敗回数を超えた場合に、ユーザーがパスワードのリセットまたは変更のために待機しなければならない分数を定義します。*password_reset.request.max_attempt_window*プロパティの時間が短いと、パスワードリセットの試行回数が増えるため、パスワードの総当たり攻撃に成功するリスクが高まります。デフォルトの 1440 分をお勧めします。 • 以前の説明: *password_reset.request.max_attempt_window*

自動翻訳

ドキュメント	更新回数
	が推奨値の 1440 以下に設定されていない場合、誤った認証試行の最大回数に達してもアカウントがロックされないため、アカウントの総当たり攻撃を実行できる可能性があります。 • 新しい修正:プロパティ *password_reset.request.max_attempt_window* が 1440 以上に設定されていることを確認します。 • 以前の修正:プロパティ *password_reset.request.max_attempt_window* が 1440 以下に設定されていることを確認します。 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
GlideRecord スコープフェンシングの従来の動作を無効にする (セキュリティセンター 1.3 の新機能、1.5 および 2.0 で更新)	• 新しい簡単な説明: *Disable GlideRecord Scope Fencing Legacy Behavior* • 以前の簡単な説明: *Enable GlideRecord Scope Fencing Legacy Behavior* • 新しい修正:Glide プロパティ *glide.record.legacy_cross_scope_access_policy_in* を false に設定します。sys_propertiesテーブルに存在しない場合、デフォルト値は true です。 • 以前の修正:Glide プロパティ *glide.record.legacy_cross_scope_access_policy_in* を false に設定します。
無効なパスワードリセットの試行回数を制限する (セキュリティセンター 1.3 で更新、2.0 で更新)	• 新しい簡単な説明: *Limit Invalid Password Reset Attempts* • 以前の簡単な説明: *Minimize Reset Password Request Max Attempt Allowance*

ベースラインバージョン **4.0** の更新されたハードニング設定

一部のハードニング設定は、ベースラインバージョン 4.0 のリリースで更新 セキュリティセンター 。

ベースラインバージョン 4.0 では、レコード間のスタイルと一貫性のために、簡単な説明がいくつか更新されています。さらに、多くのプロパティ関連スクリプトも更新され、sys_property テーブルからプロパティが削除された場合のデフォルト値の精度が向上しています。

ドキュメント	更新回数
SOAP 要求に認証を必須とする (Security Center 1.3、1.5、および 2.0 で更新)	• 新しい技術構成 名：*glide.basicauth.required.soap, glide.soap.require_ws_security* • 古い技術構成 名：*glide.basicauth.required.soap* • 新しい説明:glide プロパティ *glide.basicauth.required.soap* は、インスタンスへの SOAP 要求を行うためにベーシック認証が必要かどうかを制御します。*glide.basicauth.required.soap* が推奨値の true に設定されていない場合、SOAP 操作を実行する非認証ユーザーは soap.guest ユーザーにマップされます。これにより、非認証ユーザーが、インスタンスにログインしているユーザーであるかのように、インスタンスに対して操作を実行できます。*com.glide.soap.guest_user*内のユーザー定義に追加のロールが割り当てられると、追加の影響が生じる可能性があります。 • 以前の説明:glide プロパティ *glide.basicauth.required.soap* は、インスタンスへの SOAP 要求を行うために認証が必要かどうかを制御します。*glide.basicauth.required.soap* が推奨値の true に設定されていない場合、インスタンス上の SOAP 要求の認証は無効になります。これにより、管理者または保守レベルの操作への認証されていないアクセスが許可されます。これにより、インスタンス内のセキュリティコントロールが無効になります。 • 新しい修正:プロパティ *glide.basicauth.required.soap* 値が true に設定されていることを確認します。または、プロパティ *glide.soap.require_ws_security* を true に設定し、製品ドキュメントに従って WS セキュリティプロファイルを構成することで、WS セキュリティのインスタンスを構成します。 • 以前の修正:プロパティ *glide.basicauth.required.soap* が sys_propertiesテーブルに存在し、true に設定されていることを確認します。 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。

ドキュメント	更新回数
Jelly スクリプトをエスケープ (セキュリティセンター 1.3 および 1.5 で更新)	• 新しい説明:このプロパティは、出力ストリームに書き込まれる前に、含まれるすべての JS および HTML 文字列をエスケープし、いくつかの XSS の問題が発生するのを防ぎます。*glide.ui.escape_all_script* が推奨値の true に設定されていない場合、Jelly に挿入されたスクリプトのエスケープは無効になります。この軽減策がなければ、プラットフォームはさまざまなスクリプトインジェクション攻撃に対して広くオープンになります。攻撃者はインスタンス上で任意の Rhino スクリプトを実行する可能性があります。 • 以前の説明:次のプロパティは、<j:jelly に含まれるすべての JS 文字列と HTML 文字列をエスケープします>...</j:jelly> を実行して、いくつかの XSS の問題が発生するのを防ぎます。*glide.ui.escape_all_script* が推奨値の「true」に設定されていない場合、Jelly に挿入されたスクリプトのエスケープは無効になります。この軽減策がなければ、プラットフォームはさまざまなスクリプトインジェクション攻撃に対して広くオープンになります。攻撃者はインスタンス上で任意の Rhino スクリプトを実行する可能性があります。
CSRF 検証をバイパスする警告をユーザーが受け入れるのを防ぐ (Security Center 1.3 および 1.5 で更新)	• 新しい簡単な説明:CSRF 検証をバイパスする警告をユーザーが受け入れるのを防ぐ • 以前の簡単な説明:CSRF トークンの厳密な検証を強制する • 新しい説明:このプロパティは、悪意のある可能性がある要求をインスタンスに送信する警告をユーザーが受け入れることを防止します。この警告は、被害者の他のアクティブセッションの 1 つに属する Anti-CSRF トークンの一致が間違っているために POST 要求が失敗した場合に表示されます。*glide.security.csrf.strict.validation.mode* が推奨値の true に設定されていない場合、攻撃者は、被害者に属する別のアクティブセッションから漏洩した CSRF 対策トークンを利用して CSRF 攻撃を仕組むことができます。インスタンスへの POST 要求には、ユーザーの現在のセッションと一致する sysparm_ck または X-UserToken 内に CSRF 対策トークンが含まれています。CSRF 対策トークンがユーザーの他のアクティブ セッションの 1 つに関連付けられている場合、このプロパティが false に設定されている場合、POST 要求は security_interceptor.do への 302 リダイレクトを返し、ユーザーが使用できる [続行] ボタンを使用します。このボタンをクリッ

自動翻訳

ドキュメント	更新回数
	クすると、有効な CSRF 対策トークンが含まれる場合を除き、要求がインスタンスに再送信されます。このプロパティを true に設定すると、security_interceptor.do ページへの 302 リダイレクトに [続行] ボタンが表示されず、ユーザーは要求を再送信できません。CSRF 攻撃が成功すると、攻撃者は被害者が実行できるすべての操作を効果的に実行できます。 • 以前の説明:このプロパティは、CSRF トークンの再利用を防止する CSRF トークンの厳格な検証を有効にします。*glide.security.csrf.strict.validation.mode* が推奨値の true に設定されていない場合、CSRF トークンが再利用され、CSRF 攻撃の扉が開かれる可能性があります。 • 新しい CVSS スコア:3.7 • 以前の CVSS スコア:3.1
イベント管理 HTTP プロセッサで認証を必須とする (Security Center 1.3 の新機能、1.5 で更新、2.0 で削除)	• 新しい簡単な説明:イベント管理 HTTP プロセッサで認証を必須とする • 以前の簡単な説明:イベント管理 HTTP プロセッサで認証を必須とする
Anti-CSRF トークンを有効にする (Security Center 1.3 の新機能、1.5 で更新、2.0 で削除)	• 新しい説明: Cross-Site Request Forgery (CSRF) は、認証されたユーザーに、現在認証されている Web アプリケーションに要求を送信するように強制する攻撃です。CSRF 攻撃は、認証されたユーザーに対する Web アプリケーションの信頼を悪用します。このプロパティを使用すると、セキュアトークンを使用して受信要求を識別して検証できます。このトークンはクロスサイトリクエストフォージェリ攻撃を防ぐために使用されます。*glide.security.use_csrf_token* が推奨値の true に設定されていない場合は、CSRF が可能です。 • 以前の説明:Cross-Site Request Forgery (CSRF) は、認証されたユーザーに、現在認証されている Web アプリケーションに要求を送信するように強制する攻撃です。CSRF 攻撃は、認証されたユーザーに対する Web アプリケーションの信頼を悪用します。このプロパティを使用すると、セキュアトークンを使用して受信要求を識別して検証できます。このトークンはクロスサイトリクエストフォージェリ攻撃を防ぐために使用されま

自動翻訳

ドキュメント	更新回数
	す。推奨値の true に設定されていない場合 *glide.security.use_csrf_token* CSRF が可能です。
仮想エージェント内で HTML サニタイザーを有効にする (セキュリティセンター 1.3 および 1.5 で更新)	• 新しい簡単な説明： 仮想エージェント内で HTML サニタイザーを有効にする • 以前の簡単な説明：HTML サニタイザーを有効にする • 新しい説明:このプロパティは、HtmlSanitizerService が有効かどうかを制御します。*com.glide.cs.html.sanitizer.enabled* が true に設定されていない場合、VA Web クライアントで保存されたクロスサイトスクリプティング (XSS) 攻撃が発生する可能性があります。 • 以前の説明: このプロパティは、HTMLSanitezerService が有効かどうかを制御します。*com.glide.cs.html.sanitizer.enabled* が true に設定されていない場合、VA Web クライアントで保存されたクロスサイトスクリプティング (XSS) 攻撃が発生する可能性があります。
明示的な外部ロールに対して内部アクセスを拒否する (Security Center 1.3 および 1.5 で更新)	• 新しい技術構成名：*glide.security.explicit_roles.enable_interna* • 古い技術構成名：*glide.security.explicit_roles.enable_interna* • 新しい説明:このプロパティは、外部ユーザーに snc_internal ロールをアサインできないようにします。*glide.security.explicit_roles.enable_interna* 推奨値の true に設定されている場合、保守で保護された*glide.security.explicit_roles.internal_user_bl property*のパラメーターが適用され、信頼できないユーザークラスのリストに snc_external ロールが割り当てられます。値が false に設定されている場合、*glide.security.explicit_roles.internal_user_blac* プロパティは無視されます。このプロパティの構成を誤ると、外部ユーザー アカウントが内部情報にアクセスするリスクが高まります。 • 以前の説明:これにより、外部ユーザーにsnc_internalロールがアサインされなくなります。*glide.security.explicit_roles.enable_interna* が推奨値の true に設定されておら

ドキュメント	更新回数
	ず、*glide.security.explicit_roles.internal_user_*プロパティが信頼できないユーザークラスのリストに設定されていない場合は、指定されたロールに snc_external ロールの代わりに snc_internal ロールを割り当てることができます。リストが空の場合、デフォルトですべてのユーザーにsnc_internalロールが割り当てられます。プロパティには、少なくともデフォルトのロール csm_consumer_user customer_contact が含まれている必要があります。 これらのプロパティの構成を誤ると、外部ユーザー アカウントが内部情報にアクセスするリスクが高まります。 • 新しい修正:プロパティ *glide.security.explicit_roles.enable_internal_us* が true に設定されていることを確認します。 • 以前の修正:プロパティ *glide.security.explicit_roles.enable_internal_us* が true に設定されていること、およびプロパティ *glide.security.explicit_roles.internal_user_blac* に危険アイテム csm_consumer_user customer_contact が含まれていることを確認します。 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
WSDL 要求に認証を必須とする (セキュリティセンター 1.3 および 1.5 で更新)	• 新しい説明: *glide.basicauth.required.wsdl* が推奨値の true に設定されていない場合、WSDL 要求のベーシック認証が無効になります。WSDL は、インスタンステーブルスキーマなどの Web サービスを記述するために使用されるプロトコルであり、テーブル内のデータを共有するためのメカニズムではありません。このプロパティを true に設定すると、非認証ユーザーにテーブルスキーマを公開できます。 • 以前の説明: *glide.basicauth.required.wsdl* が推奨値の true に設定されていない場合、WSDL 要求のベーシック認証が無効になります。これにより、認証されていないユーザーへの情報が漏洩する可能性があります。 • 新しい CVSS スコア:5.3 • 以前の CVSS スコア:4.3

ドキュメント	更新回数
URL 許可リストのチェックを強制する (セキュリティセンター 1.3、1.5、および 2.0 で更新)	ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
制限されたダウンロード可能な MIME タイプを定義する (セキュリティセンター 1.3、1.5、および 2.0 で更新)	• 新しい簡単な説明：制限されたダウンロード可能な MIME タイプを定義する • 以前の簡単な説明：ダウンロード可能な MIME タイプを制限する • 新しい説明:text/html、image/svg、image/svg+xml、application/xml などの危険なアイテムが *glide.ui.attachment.download_mime_types* に含まれている場合、危険なファイルがブラウザーでインラインでレンダリングされ、クロスシットスクリプト攻撃 (XSS) につながる可能性があります。このプロパティは、ブラウザにインライン表示しない添付 MIME タイプのカンマ区切りリストです。たとえば、text/html を含めると、HTML ファイルはブラウザにインラインで表示されるのではなく、添付ファイルとしてクライアントに強制的にダウンロードされます。このリストを適切に管理することで、クロスサイトスクリプティング攻撃を防ぐことができます。 • 以前の説明: *glide.ui.attachment.download_mime_types* に text/html、image/svg、image/svg+xml、application/xml などの危険なアイテムが含まれている場合、危険なファイルはブラウザーでインラインでレンダリングされ、クロスシットスクリプティング攻撃 (XSS) につながる可能性があります。このプロパティは、ブラウザにインライン表示しない添付 MIME タイプのカンマ区切りリストです。たとえば、text/html を含めると、html ファイルはブラウザでインライン表示されず、添付ファイルとしてクライアントに強制的にダウンロードされます。このリストを適切に管理することで、クロスサイトスクリプティング攻撃を防ぐことができます。
リストビューでの HTML をエスケープ (セキュリティセンター 1.3 および 1.5 で更新)	• 新しい説明:このプロパティは、HTML フィールドのリストビューの表示をサニタイズするのに役立ちます。*glide.ui.escape_html_list_field*が推奨値の true に設定されていない場合、悪意のあるユーザーがフォームフィールド内に HTML コードを挿入して、さまざまなクライアント/ユーザーセッションで不要なスクリプトを実行できます。これは、攻撃者がセッション情報や機密データを盗むために利用される可能性があります。

ドキュメント	更新回数
	• 以前の説明:次のプロパティは、HTML フィールドのリストビューの表示をサニタイズするのに役立ちます。*glide.ui.escape_html_list_field*が推奨値の true に設定されていない場合、悪意のあるユーザーがフォームフィールド内に HTML コードを挿入して、さまざまなクライアント/ユーザーセッションで不要なスクリプトを実行できます。これは、攻撃者がセッション情報や機密データを盗むために利用される可能性があります。
外部ユーザー登録用の電子メールドメインを制限する (セキュリティセンター 1.3、1.5、および 2.0 で更新)	• 新しい簡単な説明：外部ユーザー登録用の電子メールドメインを制限する • 以前の簡単な説明:外部ユーザー登録用の電子メールドメインを制限する (プラグインの適用性:外部ユーザー登録) • 新しい説明: *sn_ext_usr_reg.allowed_email_domains* プロパティは、ServiceNow インスタンスへの自己登録が許可されるメールアドレスを定義します。受け入れ可能なドメインのリストで *sn_ext_usr_reg.allowed_email_domains* が設定されていない場合、メールアドレスを持つユーザーはインスタンスにアカウントを登録できます。定義されていない場合、悪意のある攻撃者が、インスタンスへの認証されたアクセスを取得するために、望ましくないドメインのメールアドレスを使用して登録を実行する可能性があります。 • 旧説明: *sn_ext_usr_reg.allowed_email_domains* 受け入れ可能なドメインのホワイトリストが設定されていない場合、悪意のある攻撃者が不要なドメインのメールアドレスを使用して登録を実行する可能性があります。 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
外部ユーザー登録向けに Captcha を有効にする (Security Center 1.3 および 1.5 で更新)	• 新しい簡単な説明:外部ユーザー登録向けに Captcha を有効にする • 以前の簡単な説明:外部ユーザー登録で Captcha を有効にする (プラグインの適用:外部ユーザー登録) • ルールスクリプト:検出精度を向上させるためにスクリプトが更新されました

ドキュメント	更新回数
外部ユーザー登録リンクの有効期限を最小化する (セキュリティセンター 1.3 および 1.5 で更新)	• 新しい簡単な説明：外部ユーザー登録リンクの有効期限を最小化する • 以前の簡単な説明：外部ユーザー登録リンクの有効期限を最小化する (プラグインの適用性：外部ユーザー登録) • ルールスクリプト:検出精度を向上させるためにスクリプトが更新されました
感染したファイルのダウンロードを許可しない (セキュリティセンター 1.5 および 2.0 で更新)	• 新しい簡単な説明：感染したファイルのダウンロードを許可しない • 以前の簡単な説明：感染したファイルのダウンロードを許可しない • 新しい修正: プロパティ *com.glide.snap.infected_download_allowed* が False に設定されていることを確認します。 • 以前の修正:プロパティ *com.glide.snap.infected_download_allowed* が True に設定されていることを確認します。 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
AttachmentCreator SOAP Web サービスでファイルの MIME タイプを検証する (Security Center 1.3 の新機能、1.5 で更新)	• 新しい説明: *glide.attachment.enforce_security_validation* が推奨値の true に設定されていない場合、添付ファイルの MIME タイプの検証が行われず、間違ったファイル拡張子を使用して危険なファイルがシステムにアップロードされる可能性があります。このプロパティを true に設定すると、ファイルは正しいファイルタイプ拡張子でアップロードされます。少なくともMIMEタイプの検証を使用して、ファイルのアップロードを検証することがセキュリティのベストプラクティスです。 • 以前の説明: *glide.attachment.enforce_security_validation* が推奨値の True に設定されていない場合、添付ファイルの MIME タイプの検証が行われず、間違ったファイル拡張子を使用して危険なファイルがシステムにアップロードされる可能性があります。このプロパティを true に設定すると、ファイルは正しいファイルタイプ拡張子でアップロードされます。少なくともMIMEタイプの検証を使用して、ファイルのアップロードを検証することがセキュリティのベストプラクティスです。

ドキュメント	更新回数
	• 新しい修正:プロパティ *glide.attachment.enforce_security_validation* が true に設定されていることを確認します。 • 以前の修正:プロパティ *glide.attachment.enforce_security_validation* が True に設定されていることを確認します。
MultiSSO のデバッグを無効にする (Security Center 1.3 および 1.5 で更新)	• 新しい簡単な説明：MultiSSO のデバッグを無効にする • 以前の簡単な説明：MultiSSO のデバッグを無効にする (プラグインの適用性：複数プロバイダーのシングルサインオン)
許可される ServiceNow 内部 IP アドレスを定義する (セキュリティセンター 1.3 および 1.5 で更新)	• 新しい技術構成名：*glide.ip.authenticate.strict* • 古い技術構成名：*glide.ip.authenticate.strict,glide.ip.authen* • 新しい説明:*glide.ip.authenticate.strict* が true に設定されている場合、ServiceNow の内部担当者とシステムは、必須 IP 範囲からインスタンスへの受信接続のみを行うことができます。この制限により、ServiceNow は重要な内部インフラストラクチャに対するインスタンスを可視化できなくなり、サポートスタッフや営業スタッフなど、より広範な ServiceNow 担当者が企業ネットワーク経由でアクセスできなくなります。「true」に設定すると、*glide.ip.authenticate.allow* プロパティを使用して内部 ServiceNow 受信接続が許可されます。true に設定されていない場合は、 *glide.ip.authenticate.allow* で定義されているより広範な ServiceNow 内部 IP 範囲を使用して、内部 ServiceNow 受信接続が許可されます。 • 以前の説明:*glide.ip.authenticate.strict* true に設定されている場合、*glide.ip.authenticate.allow.secured* で指定された IP 範囲のみがインスタンスへの受信接続を確立できます。このプロパティには、必須の ServiceNow 内部 IP 範囲 (セキュア VPN、DC) のみのリストが含まれています。*glide.ip.authenticate.allow.secured*が推奨値または順列「10.0.0.0/8, 37.98.232.0/21, 103. 23.64.0/22, 149.96.0.0/17, 149.96.0.0/16, 199.91.136.0/21, 148.139.0.0/16, 127.0.0.1" または新しい値リ

ドキュメント	更新回数
	スト "10.0.0.0/8, 37.98.232.0/21, 103.23.64.0/22, 149.96.0.0/17, 149.96.0.0/16, 199.91.136.0/21, 148.139.0.0/16, 127.0.0.1, 0:0:0:0:0:0:0:0, ::1" は IPv6 localhost を Utah に追加するため、SN データセンターおよびセキュア VPN 以外の信頼できないソースが機密にアクセスする可能性がありますインスタンスでのエンドポイントの監視。 • 新しい修正: プロパティ *glide.ip.authenticate.allow.secured* に信頼できる値のみが含まれていて、プロパティ *glide.ip.authenticate.strict* が true に設定されていることを確認します。 • 以前の修正:プロパティ *glide.ip.authenticate.allow.secured* に「10.0.0.0/8, 37.98.232.0/21, 103.23.64.0/22, 149.96.0.0/17, 149.96.0.0/16, 199.91.136.0/21, 148.139.0.0/16, 127.0.0.1, 0:0:0:0:0:0:0:1, ::1」の値のみが含まれ、プロパティ *glide.ip.authenticate.strict* が true に設定されていることを確認します。 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
XMLDocument2 ストリーミングパーサー内のエンティティ拡張を無効にする (セキュリティセンター 1.5 で更新)	• 新しい簡単な説明:XMLDocument2 ストリーミングパーサー内のエンティティ拡張を無効にする • 以前の簡単な説明:エンティティ拡張を無効にする • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
ドット連結フィールドにドメインセパレーションを適用する (セキュリティセンター 1.3、1.5、および 2.0 で更新)	• 新しい簡単な説明：ドット連結フィールドにドメインセパレーションを適用する • 以前の簡単な説明：ドット連結フィールドにドメインセパレーションを適用する (プラグインの適用性：ドメインセパレーション) • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
CMDB モデルの権限を制限する (セキュリティセンター 1.3 および 1.5 で更新)	ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
モバイルアプリケーションのバックグラウンド処理中にペーストボードをクリアすることを必須とする (Security Center 1.3 の新機能、1.5 で更新)	• 新しい説明: *glide.sg.clear_pasteboard_when_backgrounded* プロパティは、ServiceNow モバイルアプリからコピーされたテキストを、アプリがバッ

ドキュメント	更新回数
	クグラウンドモードになった後もクリップボードとペーストボードに保持するかどうかを制御します。推奨値の true に設定されていない場合、機密情報が Android または iOS クリップボードに開示され、デバイス上の他のアプリケーションに公開される可能性があります。 • 以前の説明:このプロパティ *glide.sg.clear_pasteboard_when_backgrounded* 、Se モバイルアプリからコピーされたテキストが、アプリにフォーカスがなくなった後もクリップボード/ペーストボードに保持されるかどうかを制御します。推奨値の true に設定されていない場合、機密情報が Android または iOS クリップボードに開示され、デバイス上の他のアプリケーションに公開される可能性があります。
アカウント復旧の有効化 (セキュリティセンター 1.3 および 1.5 で更新)	• 新しい簡単な説明：アカウント復旧の有効化 • 以前の簡単な説明：アカウント復旧の有効化 (プラグインの適用性：複数プロバイダーのシングルサインオン)
SQL エラーメッセージを無効にする (Security Center 1.3 および 1.5 で更新)	• 新しい説明: *glide.db.loguser* が推奨値の false に設定されていない場合、機密性の高いサーバー側のエラーメッセージがエンドユーザーに表示される可能性があります。エラーメッセージには、スタックトレースやデータベースの構造に関する情報が含まれており、前提条件が存在する場合、SQL インジェクションを成功させるために必要な知識を攻撃者に提供する可能性があります。多層防御のため、これらのエラーメッセージはエンドユーザーに表示されません。 • 以前の説明: *glide.db.loguser* が推奨値の false に設定されていない場合、機密性の高いサーバー側のエラーメッセージがエンドユーザーに表示される可能性があります。
相対リンクを強制する (セキュリティセンター 1.3 および 1.5 で更新)	• 新しい説明: *glide.cms.catalog_uri_relative* プロパティは、/ess/catalog.do の URI パラメーターからの相対リンクを強制します。*glide.cms.catalog_uri_relative* が推奨値の true に設定されていない場合、URL は enforceRelativeURL(url) 関数でサニタイズされません。絶対 URL は、パラメーターまたはフィールド値の一部として使用するとセキュリティリスクを引き起こす可能性があり、ソースページが攻撃者が制御

ドキュメント	更新回数
	する Web サイトにリダイレクトされます。このプロパティは、サービスポータルに置き換えられた従来のコンテンツ管理システム (CMS) に影響します。 • 以前の説明: *glide.cms.catalog_uri_relative* プロパティは、/ess/catalog.do の URI パラメーターからの相対リンクを強制します。*glide.cms.catalog_uri_relative* が推奨値の true に設定されていない場合、URL は enforceRelativeURL(url) 関数でサニタイズされません。絶対 URL は、パラメーターまたはフィールド値の一部として使用するとセキュリティリスクを引き起こす可能性があり、ソースページが攻撃者が制御する Web サイトにリダイレクトされます。
GlideXMLUtil スクリプト可能項目のエンティティ拡張しきい値を最小化する (セキュリティセンター 1.3、1.5、および 2.0 で更新)	• 新しい簡単な説明：GlideXMLUtil スクリプト可能項目のエンティティ拡張しきい値を最小化する • 以前の簡単な説明：エンティティ拡張しきい値を最小化する • 新しい説明:このプロパティは、XML パーサー内のエンティティ展開の最大量を制御します。*glide.xmlutil.max_entity_expansion* が推奨値の 3000 以下に設定されていない場合、GlideXMLUtil 解析スクリプト可能はサービス拒否攻撃に対して脆弱になる可能性があります。 • 以前の説明:このプロパティは、XML パーサー内のエンティティ展開の最大量を制御します。*glide.xmlutil.max_entity_expansion* が推奨値の 3000 以下に設定されていない場合、XML パーサーはサービス拒否攻撃に対して脆弱になる可能性があります。 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
GlideRecord スコープフェンシングの従来の動作を無効にする (セキュリティセンター 1.3 の新機能、1.5 および 2.0 で更新)	• 新しい説明:GlideRecord は、そのレベルのアクセス権で構成されていないテーブルへのクロススコープ作成/更新アクセスを提供しました。このスコープ指定のアクセス動作にパッチが適用されたときに、顧客のアプリケーションが壊れないようにするために、プロパティ *glide.record.legacy_cross_scope_access_policy_i* が作成されました。true の場合、クロススコープアクセスは従来の動作にフォールバッ

自動翻訳

ドキュメント	更新回数
	クします (安全でない)。このプロパティは、スコープフェンシングを無効にし、スコープ対象アプリがグローバルスクリプトインターフェイスにアクセスできるようにします。セキュリティのベストプラクティスは、スコープフェンシングの制限を設けることです。スコーピングにより、アプリケーションは最小特権の原則に従って、明示的なアクセス権を持つリソースまたはスコープ内のリソースにのみアクセスできます。この機能を無効にすると、機密性、可用性、および整合性に影響を与える可能性があります。 • 以前の説明:従来の動作では、作成/更新アクセスが許可されていないテーブルへのアクセス権が提供されていました。このスコープ指定のアクセス動作にパッチを適用したときに、従来の顧客のアプリケーションが壊れないようにするために、プロパティ *glide.record.legacy_cross_scope_access_policy_in* が作成されました。true の場合、クロススコープアクセスは従来の動作にフォールバックします (安全でない)。このプロパティは、スコープフェンシングを無効にし、スコープ対象アプリがグローバルスクリプトインターフェイスにアクセスできるようにします。セキュリティのベストプラクティスは、スコープフェンシングの制限を設けることです。スコーピングにより、アプリケーションは最小特権の原則に従って、明示的なアクセス権を持つリソースまたはスコープ内のリソースにのみアクセスできます。この機能を無効にすると、機密性、可用性、および整合性に影響を与える可能性があります。
MultiSSO プラグインの更新バージョンを有効にする (Security Center 1.3 および 1.5 で更新)	• 新しい簡単な説明：更新された複数 SSO プラグインのバージョンを有効にする • 以前の簡単な説明：更新された複数 SSO プラグインのバージョンを有効にする (プラグイン適用：複数プロバイダーのシングルサインオン)
LDAP 認証で SSL を有効にする (Security Center 1.5 および 2.0 で更新)	検出精度を向上させるためにスクリプトが更新されました。
API 要求でパスワードリセットを強制する (Security Center 1.5 で更新)	検出精度を向上させるためにスクリプトが更新されました。
ログイン時にパスワードポリシーを適用しない (セキュリティセンター 1.5 で更新、2.0 で削除)	• 新しい説明:プロパティ *glide.apply.password_policy.on_login* を False に設定すると、ログイン時にパスワードの複雑さが強制されなくなります。プロパティを True に設定すると、パスワード

自動翻訳

ドキュメント	更新回数
	の複雑さが強制され、組織のポリシーコンプライアンスの問題が発生します。 ASVS 4.03 v2.1.9の推奨事項によると: 許可される文字の種類を制限するパスワード作成ルールがないことを確認します。大文字、小文字、数字、特殊文字を必須とするべきではありません。(C6) ASVS の推奨事項では、パスワードの複雑さを強制する代わりに、パスワードの長さに最低 12 文字を強制します。 参照: OWASP ASVS v4.0 認証 • 以前の説明: プロパティ *glide.apply.password_policy.on_login* を False に設定すると、ログイン時にパスワードの複雑さが強制されません。プロパティを True に設定すると、パスワードの複雑さが強制され、組織のポリシーコンプライアンスの問題が発生します。 ASVS 4.03 v2.1.9の推奨事項によると: 許可される文字の種類を制限するパスワード作成ルールがないことを確認します。大文字、小文字、数字、特殊文字を必須とするべきではありません。(C6) ASVS の推奨事項では、パスワードの複雑さを強制する代わりに、パスワードの長さに最低 12 文字を強制します。 参照: OWASP ASVS v4.0 認証
アクティブな SAML 構成でデモ認証を使用しない (セキュリティセンター 1.5 で更新)	• 新しい簡単な説明：アクティブな SAML 構成でデモ認証を使用しない • 以前の簡単な説明：アクティブな SAML 構成でデモ認証を使用しない (プラグイン適用：複数プロバイダーのシングルサインオン)
SAML の「notBefore」または「notOnOrAfter」の制約期間を最小化する (セキュリティセンター 1.3 および 1.5 で更新)	• 新しい簡単な説明：SAML の「notBefore」または「notOnOrAfter」の制約期間を最小化する • 以前の簡単な説明：SAML の「notBefore」または「notOnOrAfter」の制約期間を最小化する (プラグイン適用：複数プロバイダーのシングルサインオン)

ドキュメント	更新回数
期限切れの Anti-CSRF トークンをブロック (セキュリティセンター 1.5 で更新)	• 新しい簡単な説明：期限切れの Anti-CSRF トークンをブロック • 以前の簡単な説明：期限切れの CSRF トークンをブロック
カスタマーサービスアプリケーションでのゲストウォークアップエクスペリエンスに Captcha を必須とする (セキュリティセンター 1.3 の新機能、1.5 で更新)	• 新しい簡単な説明：カスタマーサービスアプリケーションでのゲストウォークアップエクスペリエンスに Captcha を必須とする • 以前の簡単な説明：カスタマーサービスアプリケーションでのゲストウォークアップエクスペリエンスに Captcha を必須とする (プラグイン適用：カスタマーサービスのゲストウォークアップエクスペリエンス)
HR アプリの ACL 評価で代理操作をチェックする (セキュリティセンター 1.3 の新機能、1.5 で更新)	• 新しい簡単な説明：HR アプリの ACL 評価で代理操作をチェックする • 以前の簡単な説明：HR アプリの ACL 評価で代理操作をチェックする (プラグイン適用：ヒューマンリソース (HR) スコープ対象アプリ)
私用メールからの HR ケースの更新を制限する (セキュリティセンター 1.3 の新機能、1.5 で更新)	• 新しい簡単な説明：私用メールからの HR ケースの更新を制限する • 以前の簡単な説明：私用メールからの HR ケースの更新を制限する (プラグイン適用：ヒューマンリソース (HR) スコープ対象アプリ) • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
MID 監査ログを有効化する (セキュリティセンター 1.3 の新機能、1.5 で更新)	• 新しい簡単な説明: MID 監査ログを有効化する • 以前の簡単な説明: MID 監査ログを有効化する (プラグイン適用：MID サーバー)
認証情報エイリアスの使用を強制する (セキュリティセンター 1.3 の新機能、1.5 で更新)	• 新しい簡単な説明：認証情報エイリアスの使用を強制する • 以前の簡単な説明：認証情報エイリアスの使用を強制する (プラグイン適用：MID サーバー)

ドキュメント	更新回数
必須の JMS 接続ファクトリ (Security Center 1.3 の新機能、1.5 および 2.0 で更新)	• 新しい簡単な説明：必須の JMS Connection Factory • 以前の簡単な説明：必須の JMS Connection Factory (プラグイン適用：MID サーバー) • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
トレーニングおよび予測フローの添付ファイルサイズの制限 (Security Center 1.3 の新機能、1.5 で更新)	• 新しい簡単な説明：トレーニングと予測フローの添付ファイルのサイズを制限する • 以前の簡単な説明：トレーニングと予測フローの添付ファイルのサイズを制限する (プラグイン適用：プラットフォームドキュメントインテリジェンス (DocIntel))
アーカイブテーブル ACL のチェックを確認する (Security Center 1.3 の新機能、1.5 で更新)	ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
セッション監査イベントのログ記録 (Security Center 1.3 の新機能、1.5 で更新)	• 新しい説明:Glide プロパティ *glide.authenticate.session_access.log_audit_ever* が true に設定されている場合、セッション監査イベントがsys_session_access_auditテーブルに作成されます。悪意のあるアクターの調査を支援するために、セッションにアクセスしたユーザーに関する情報をログに記録することをお勧めします。ログに記録される情報には、ユーザー、セッション ID (非機密)、IP アドレス、ロール、ポリシーが含まれます。 • 以前の説明:Glide プロパティ *glide.authenticate.session_access.log_audit_ever* が true に設定されている場合、セッション監査イベントがsys_session_access_auditテーブルに作成されます。悪意のあるアクターの調査を支援するために、セッションアクセスに関する一般的な情報をログに記録することをお勧めします。ログに記録される情報には、ユーザー、セッション ID (非機密)、IP アドレス、ロール、ポリシーが含まれます。
Information Request Playbook に対してスコープ付き ACL アクセスを強制する (Security Center 1.3 の新機能、1.5 で更新)	• 新しい簡単な説明:情報要求プレイブックに対してスコープされた ACL アクセスを強制する • 以前の簡単な説明:情報要求プレイブックにスコープされた ACL アクセスを強制する • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。

ドキュメント	更新回数
非アクティブなセッションを積極的に無効化する (Security Center 1.3 の新機能、1.5 および 2.0 で更新)	• 新しい説明:Glide プロパティ *glide.active.session.timeout.invalidate.session* は、Tomcat コンテナがセッションを無効にする前に、タイムアウトしたセッションを事前に無効にするかどうかを制御します。このプロパティが true に設定されていない場合、タイムアウトしたセッションが無効にならない短い間隔 (キューサイズに応じて 60+ 秒) が生じることがあります。セッションがハイジャックされた場合、攻撃者はこの短い期間にセッションを利用できる可能性があります。 • 以前の説明:Glide プロパティ *glide.active.session.timeout.invalidate.session* は、タイムアウトセッションが Tomcat コンテナより先に事前に無効化されるかどうかを制御します。このプロパティが true に設定されていない場合、タイムアウトしたセッションが無効にならない短い間隔 (キューサイズに応じて 60+ 秒) が生じることがあります。セッションがハイジャックされた場合、攻撃者はこの短い期間にセッションを利用できる可能性があります。
HTTP 応答の本文サイズを制限する (セキュリティセンター 1.3 の新機能、1.5 で更新)	• 新しい簡単な説明：HTTP 応答の本文サイズを制限する • 以前の簡単な説明：応答の本文サイズが原因で、HTTP 応答が OutofMemory 例外をトリガーしないようにしてください

自動翻訳

ベースラインバージョン **2.0** の更新されたハードニング設定

一部のハードニング設定は、ベースラインバージョン 2.0 のリリースで更新 セキュリティセンター 。

ドキュメント	更新回数
同時インタラクティブセッションの量を最小化する (Security Center 1.3 で更新)	• 新しい簡単な説明：同時インタラクティブセッション数の最小化 • 以前の簡単な説明：Glide 認証：同時インタラクティブセッション数の最大化
証明書の信頼を強制する (Security Center 1.3 で更新、 2.0 で削除、7.0 で追加)	• 新しい簡単な説明：証明書の信用を強制する • 以前の簡単な説明： 証明書の信用

ドキュメント	更新回数
パスワードリセット SMS の複雑さを最大化する (セキュリティセンター 1.3 で更新)	• 新しい簡単な説明：パスワードリセット SMS の複雑さを最大化する • 以前の簡単な説明：パスワードリセット SMS の複雑さ
高セキュリティプラグインを有効にする (Security Center 1.3 で更新)	• 新しい簡単な説明： High Security プラグインを有効にする • 以前の簡単な説明： High Security プラグイン
厳格なセッション Cookie のセキュリティを強制する (セキュリティセンター 1.3 で更新)	• 新しい簡単な説明：厳格なセッション Cookie のセキュリティを強制する • 以前の簡単な説明：安全なセッション Cookie
アクティブな SAML 構成でデモ認証を使用しない (セキュリティセンター 1.5 で更新)	• 新しい簡単な説明:アクティブな SAML 構成でデモ認証を使用しない (プラグイン適用:複数プロバイダーのシングルサインオン) • 以前の簡単な説明:アクティブな SAML 構成でデモ認証を使用しない
XMLDocument2 ストリーミングパーサー内のエンティティ拡張を無効にする (セキュリティセンター 1.5 で更新)	ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
許可された Java パッケージを制限する (セキュリティセンター 1.3 で更新)	• 新しい簡単な説明：許可された Java パッケージを制限する • 以前の簡単な説明：Java パッケージ許可リスト
モバイルアプリ UI の難読化を必須とする (セキュリティセンター 1.3 で更新)	• 新しい簡単な説明：モバイルアプリ UI の難読化を必須とする • 以前の簡単な説明：モバイルアプリ UI 難読化
お気に入りへのパブリックアクセスを無効にする (セキュリティセンター 1.3 および 2.0 で更新)	• 新しい簡単な説明：お気に入りへのパブリックアクセスを無効にする • 以前の簡単な説明：お気に入りへのパブリックアクセス
JavaScript をエスケープ (Security Center 1.3 で更新)	• 新しい説明:glide プロパティ `glide.html.escape_script` は、HTML フィールドのサニタイズに役立ちます。`glide.html.escape_script` が推奨値の true に設定されていない場合、埋め込み JavaScript を削除することで、バックエンド

ドキュメント	更新回数
	Java コンテキストからの HTML フィールド (出力エンコーディング) の入力はサニタイズされません。HTML フィールドの Javascript は、XSS を保存して反映させる可能性があります。XSS 機能により、admin などの上位ロールへの特権エスカレーションが容易になり、横方向の移動が容易になる可能性があります。 • 以前の説明:glide プロパティ *glide.html.escape_script* は html フィールドのサニタイズに役立ちます。*glide.html.escape_script* が推奨値の true に設定されていない場合、埋め込み JavaScript を削除して、バックエンド Java コンテキストから HTML フィールド (出力エンコーディング) の入力がサニタイズされません。HTML フィールドの Javascript は、XSS を保存して反映させる可能性があります。XSS 機能により、admin などの上位ロールへの特権エスカレーションが容易になり、横方向の移動が容易になる可能性があります。
サードパーティ Web サイトの埋め込みを防ぐために Xframe オプションを設定する (Security Center 1.3 で更新)	• 新しい簡単な説明： サードパーティ Web サイトの埋め込みを防ぐために Xframe オプションを設定する • 以前の簡単な説明：Xframe オプション • 新しい説明: *com.glide.cs.embed.xframe_options* が推奨値の DENY または SAMEORIGIN に設定されていない場合、Web アプリケーションのコンテンツは ALLOW-FROM URI を使用してサードパーティサイトに埋め込まれる可能性があります。信頼できないサードパーティサイトを許可すると、クリックジャッキングなどの攻撃が可能になる可能性があります。 • 以前の説明: *com.glide.cs.embed.xframe_options* が推奨値の DENY または SAMEORIGIN に設定されていない場合、ALLOW-FROM URI を使用して Web アプリケーションのコンテンツをサードパーティサイトに埋め込む可能性があります。信頼できないサードパーティサイトを許可すると、クリックジャッキングなどの攻撃が可能になる可能性があります。 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。

ドキュメント	更新回数
リストビューでの HTML をエスケープ (セキュリティセンター 1.3 および 1.5 で更新)	• 新しい簡単な説明：リストビューで HTML をエスケープ • 以前の簡単な説明：HTML をエスケープ
クラシックモバイルアプリ UI の難読化を必須とする (セキュリティセンター 1.3 で更新)	• 新しい簡単な説明：クラシックモバイルアプリ UI の難読化を必須とする • 以前の簡単な説明：クラシックモバイルアプリ UI の難読化
空の ACL でデフォルトで拒否する (Security Center 1.3 で更新)	• 新しい簡単な説明：空の ACL でデフォルトで拒否する • 以前の簡単な説明：セキュリティマネージャーのデフォルト拒否 • 新しい説明: *glide.sm.default_mode* が推奨値の拒否に設定されていない場合、リソースに対して ACL が定義されていない場合でも、インスタンスの従来のセキュリティマネージャーはリソースへのアクセスを許可します。テーブルレベルのワイルドカード ACL のみが定義されています。明示的な ACL が設定されていないものはすべて操作の影響を受けやすくなるように設定すると、 • 以前の説明: *glide.sm.default_mode* が推奨値の拒否に設定されていない場合、そのリソースに対して ACL が定義されていないとき、またはワイルドカードテーブルレベルの ACL のみが定義されているときに、従来のセキュリティマネージャーによるリソースへのアクセスが許可されます。明示的な ACL が設定されていないものはすべて操作の影響を受けやすくなるように設定すると、
パスワードリセット要求の再試行ウィンドウの持続期間を最大化する (セキュリティセンター 1.3 で更新)	• 新しい簡単な説明：パスワードリセット要求の再試行ウィンドウの持続期間を最大化する • 以前の簡単な説明：パスワードリセット要求の再試行ウィンドウ
XSD 要求に認証を必須とする (Security Center 1.3 で更新)	• 新しい簡単な説明:XSD 要求に対する認証を必須とする • 以前の簡単な説明:XSD 要求認証 • 新しい修正:プロパティ *glide.basicauth.required.xsd* が sys_propertiesテーブルに存在し、true に設定されていることを確認します。

ドキュメント	更新回数
	• 以前の修正:プロパティ *glide.basicauth.required.xsd* が true に設定されていることを確認します。 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
Jelly スクリプトをエスケープ (セキュリティセンター 1.3 および 1.5 で更新)	• 新しい簡単な説明：Jelly スクリプトをエスケープ • 以前の簡単な説明：Jelly をエスケープ
受信トランザクションを再確認する (Security Center 1.3 で更新)	• 新しい修正:プロパティ *glide.security.strict.updates* が sys_propertiesテーブルに存在し、true に設定されていることを確認します。 • 以前の修正:プロパティ *glide.security.strict.updates* が true に設定されていることを確認します。 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
静的コンテンツでダウンロード可能なファイルの種類を制限する (セキュリティセンター 1.3 で更新)	• 新しい簡単な説明：静的コンテンツでダウンロード可能なファイルタイプを制限する • 以前の簡単な説明：静的コンテンツからのダウンロードするファイルタイプの制限
PDF 要求に認証を必須とする (セキュリティセンター 1.3 で更新)	• 新しい簡単な説明：PDF 要求に認証を必須とする • 以前の簡単な説明：PDF 要求認証 • 新しい修正:プロパティ *glide.basicauth.required.pdf* が sys_propertiesテーブルに存在し、true に設定されていることを確認します。 • 以前の修正:プロパティ *glide.basicauth.required.pdf* が true に設定されていることを確認します。 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。

自動翻訳

ドキュメント	更新回数
アップロードされる MIME タイプを制限する (セキュリティセンター 1.3 および 2.0 で更新)	• 新しい簡単な説明:アップロードされる MIME タイプを制限する • 以前の簡単な説明:MIME タイプのアップロード制限 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
レガシー JQuery 動作を無効にする (セキュリティセンター 1.3 で更新)	• 新しい簡単な説明：レガシー JQuery 動作を無効にする • 以前の簡単な説明：レガシー JQuery 動作
パスワードリセット要求のロック解除ウィンドウの持続期間を最大化する (セキュリティセンター 1.3 で更新)	• 新しい簡単な説明：パスワードリセット要求のロック解除ウィンドウの持続期間を最大化する • 以前の簡単な説明：パスワードリセット要求のロック解除ウィンドウ
MultiSSO のデバッグを無効にする (Security Center 1.3 および 1.5 で更新)	• 新しい簡単な説明:MultiSSO のデバッグを無効にする (プラグインの適用:複数プロバイダーのシングルサインオン) • 以前の簡単な説明:MultiSSO のデバッグを無効にする • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
本番インスタンスの動作を強制する (セキュリティセンター 1.3 および 1.5 で更新)	• 新しい簡単な説明：本番インスタンスの動作を強制する • 以前の簡単な説明：本番インスタンスの動作
無効なパスワードリセットの試行回数を制限する (セキュリティセンター 1.3 で更新、2.0 で更新)	• 新しい簡単な説明：パスワードリセット要求の最大試行回数を最小化する • 以前の簡単な説明：パスワードリセット要求の最大試行回数
csv 要求に認証を必須とする (Security Center 1.3 で更新)	• 新しい簡単な説明：CSV 要求に認証を必須とする • 以前の簡単な説明：CSV 要求認証 • 新しい修正:プロパティ `glide.basicauth.required.csv` が sys_propertiesテーブルに存在し、true に設定されていることを確認します。

自動翻訳

ドキュメント	更新回数
	• 以前の修正:プロパティ *glide.basicauth.required.csv* が true に設定されていることを確認します。 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
パスワードリセット要求の成功ウィンドウの持続期間を最小化する (セキュリティセンター 1.3 で更新)	• 新しい簡単な説明：パスワードリセット要求の成功ウィンドウの持続期間を最小化する • 以前の簡単な説明：パスワードリセット要求の成功ウィンドウ
SOAP 要求の厳格なセキュリティを強制する (Security Center 1.3 で更新)	• 新しい簡単な説明：SOAP 要求の厳格なセキュリティを強制する • 以前の簡単な説明：SOAP 要求の厳格なセキュリティ
SOAP 要求に認証を必須とする (Security Center 1.3、1.5、および 2.0 で更新)	• 新しい簡単な説明：SOAP 要求に認証を必須とする • 以前の簡単な説明：SOAP 要求認証 • 新しい説明:glide プロパティ *glide.basicauth.required.soap* は、インスタンスへの SOAP 要求を行うために認証が必要かどうかを制御します。*glide.basicauth.required.soap* が推奨値の true に設定されていない場合、インスタンス上の SOAP 要求の認証は無効になります。これにより、管理者または保守レベルの操作への認証されていないアクセスが許可されます。これにより、インスタンス内のセキュリティコントロールが無効になります。 • 以前の説明:glide プロパティ *glide.basicauth.required.soap* は、インスタンスへの SOAP 要求を行うために認証が必要かどうかを制御します。glide.basicauth.required.soap が推奨値の true に設定されていない場合、インスタンスで SOAP 要求の認証が無効になります。これにより、管理者または保守レベルの操作への認証されていないアクセスが許可されます。これにより、インスタンス内のすべてのセキュリティコントロールが無効になります。 • 新しい修正:プロパティ *glide.basicauth.required.soap* が sys_propertiesテーブルに存在し、true に設定されていることを確認します。

ドキュメント	更新回数
	• 以前の修正:プロパティ *glide.basicauth.required.soap* が true に設定されていることを確認します。 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
allowlistDisable エンティティ拡張のある XMLdoc2 エンティティ検証を必須とする (セキュリティセンター 1.3 で更新)	• 新しい簡単な説明：allowlistDisable エンティティ拡張のある XMLdoc2 エンティティ検証を必須とする • 以前の簡単な説明：allowlistDisable エンティティ拡張のある XMLdoc2 エンティティ検証
ドット連結フィールドにドメインセパレーションを適用する (セキュリティセンター 1.3、1.5、および 2.0 で更新)	• 新しい簡単な説明:ドット連結フィールドにドメインセパレーションを適用する (プラグイン適用:ドメインセパレーション) • 以前の簡単な説明：ドメインセパレーションを適用 • 新しい説明:このプロパティは、ドット連結フィールドにドメインセパレーション機能を確実に適用するために、結合クエリにドメインセパレーション条件を指定するかどうかを制御します。ドメインセパレーションを使用しているインスタンスで *glide.sys.domain.include_domain_condition_on_joi* が推奨値である true に設定されていない場合、特定のドメインと共有されない機密情報が公開される可能性があります。 • 以前の説明:このプロパティは、ドット連結フィールドにドメインセパレーション機能を確実に適用するために、結合クエリにドメインセパレーション条件を指定するかどうかを制御します。*glide.sys.domain.include_domain_condition_on* 推奨値の true に設定されていない場合、特定のドメインと共有されるべきではない機密情報が開示される可能性があります。 • 新しい修正:ドメインセパレーションプラグインがアクティブな場合に、プロパティ *glide.sys.domain.include_domain_condition_on_joi* が true に設定されていることを確認します。 • 以前の修正:プロパティ *glide.sys.domain.include_domain_condition_on_joi* が true に設定されていることを確認します。 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。

ドキュメント	更新回数
JSONP 要求を信頼できる URL に制限する (Security Center 1.3 で更新)	• 新しい簡単な説明：JSONP 要求を信頼できる URL に制限する • 以前の簡単な説明：JSONP 要求包含リスト • 新しい説明:このプロパティは、JSONP 要求を許可/拒否するための angularJS $http サービスの信頼できる URL を指定します。プロパティが必要です。これは顧客にとって破壊的変更となる可能性があるため、信頼できる URL を追加する方法が必要です。*angular.jsonp.inclusion_list.enabled* が推奨値の「true」に設定されていない場合、JSONP 要求は任意の URL に対して許可されます。 • 以前の説明:このプロパティは、JSONP 要求を許可/拒否するための angularJS $http サービスの信頼できる URL を指定します。プロパティが必要です。これは顧客にとって破壊的変更となる可能性があるため、信頼できる URL を追加する方法が必要です。*angular.jsonp.inclusion_list.enabled*) が推奨値の true に設定されていない場合、任意の URL に対して jsonp 要求が許可されます。
1 日あたりのパスワードリセット SMS の最大数を最小化する (セキュリティセンター 1.3 で更新)	• 新しい簡単な説明：1 日あたりのパスワードリセット SMS の最大数を最小化する • 以前の簡単な説明：1 日あたりのパスワードリセット SMS の最大数
パスワードリセット検証遅延期間を最大化する (セキュリティセンター 1.3 で更新)	• 新しい簡単な説明：パスワードリセット検証遅延期間を最大化する • 以前の簡単な説明：パスワードリセットの検証遅延 • 新しい説明: *password_reset.verification.delay* が推奨値の 1000 以上に設定されていない場合、ログインは総当たり攻撃の影響を受けやすくなります。このミリ秒単位の遅延により、悪意のあるアクターが自動化ツール (「ボット」) を使用してユーザーの識別または検証の詳細を推測しようとする能力が制限されます。 • 以前の説明: *password_reset.verification.delay* が推奨値の 1000 以上に設定されていない場合、ログインは総当たり攻撃の影響を受けやすくなります。このミリ秒単位の遅延によ

ドキュメント	更新回数
	り、ハッカーが自動化ツール (「ボット」) を使用してユーザーの識別または検証の詳細を推測しようとする能力が制限されます。
データブローカー REST API に認証を必須とする (Security Center 1.3 で更新)	• 新しい簡単な説明：データブローカー REST API に対する認証を必須とする • 以前の簡単な説明：データブローカー Rest API 認証 • 新しい説明: *glide.basicauth.required.databrokerrestapiproces* が推奨値の true に設定されていない場合、すべての受信データブローカー REST API 要求にベーシック認証は必要ありません。これにより、インスタンスから認証されていない情報が開示される可能性があります。 • 以前の説明:Utah リリース以降、*glide.basicauth.required.databrokerrestapiproces* が推奨値の「true」に設定されていない場合、すべての受信データブローカー REST API 要求にベーシック認証は必要ありません。これにより、インスタンスから認証されていない情報が開示される可能性があります。 • 新しい修正:プロパティ *glide.basicauth.required.databrokerrestapiproces* がsys_propertiesテーブルに存在し、true に設定されていることを確認します。 • 以前の修正:Utah リリース以降を実行しているインスタンスでプロパティ *glide.basicauth.required.databrokerrestapiproces* が true に設定されていることを確認します。 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
JSONv2 要求に認証を必須とする (Security Center 1.3 で更新)	• 新しい簡単な説明：JSONv2 要求に対する認証を必須とする • 以前の簡単な説明：JSONv2 要求認証 • 新しい修正:プロパティ *glide.basicauth.required.jsonv2* が sys_propertiesテーブルに存在し、true に設定されていることを確認します。 • 以前の修正:プロパティ *glide.basicauth.required.jsonv2* が true に設定されていることを確認します。 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。

ドキュメント	更新回数
埋め込み HTML で Javascript タグを無効にする (セキュリティセンター 1.3 で更新)	• 新しい簡単な説明：埋め込み HTML で JavaScript タグを無効にする • 以前の簡単な説明：埋め込み HTML で Javascript タグを許可する • 新しい修正:プロパティ *glide.ui.security.codetag.allow_script* がsys_propertiesテーブルに存在し、false に設定されていることを確認します。 • 以前の修正:プロパティ *glide.ui.security.codetag.allow_script* が false に設定されていることを確認します。 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
Security Jump Start プラグイン (ACL ルール) を有効化する (Security Center 1.3 で更新)	• 新しい簡単な説明:セキュリティのジャンプスタートプラグイン (ACL ルール) を有効にする • 以前の簡単な説明：Security Jump Start プラグイン (ACL ルール)
SOAP 要求のゲストユーザーを設定する (Security Center 1.3 および 2.0 で更新)	• 新しい簡単な説明：SOAP 要求のゲストユーザーを設定する • 以前の簡単な説明：SOAP 要求のゲストユーザー
XML 外部エンティティを制限する (セキュリティセンター 1.3 および 2.0 で更新)	• 新しい簡単な説明：XML 外部エンティティを制限する • 以前の簡単な説明：XML エンティティ検証 URL 許可リスト
ACL を有効にしてライブプロファイルの詳細を制御する (Security Center 1.3 で更新)	• 新しい簡単な説明:ACL を有効にしてライブプロファイルの詳細をコントロールする • 以前の簡単な説明:ACL を有効にしてライブプロファイルの詳細をコントロールする
カスタムジャーナルエントリへのアクセスを制限する (セキュリティセンター 1.3 で更新、2.0 で削除)	• 新しい簡単な説明：カスタムジャーナルエントリへのアクセスを制限する • 以前の簡単な説明：安全なカスタムジャーナルエントリ • 新しい説明: *glide.live_feed.custom_journal.acl_check_enabled* が推奨値の true に設定されていない場合、すべてのユーザーがライフフィード機能内の

自動翻訳

ドキュメント	更新回数
	すべてのジャーナルエントリを表示できるようになります。プロパティを true に設定すると、推奨機能であるカスタムジャーナルフィールドの ACL が優先されます。 • 以前の説明: *glide.live_feed.custom_journal.acl_check_enabled* が推奨値のtrueに設定されていない場合、すべてのユーザーがすべてのジャーナルエントリを表示できます。プロパティを true に設定すると、推奨機能であるカスタムジャーナルフィールドの ACL が優先されます。
パスワードリセットの OTP 期限を 1 時間に設定 (Security Center 2.0 で更新)	• 新しい説明:このプロパティ *glide.pwd_reset.onetime.token.validity* 、パスワードリセットメール内のリンクを、その *glide.pwd_reset.onetime.token.validity property*で指定された時間の経過後に期限切れにすることができます。パスワードリセットトークンの有効期限は、通常のユーザーエクスペリエンスに応じてできるだけ短く保つ必要があります。パスワードリセットトークンの有効期間が長いと、悪意のあるアクターがアカウントを乗っ取りやすくなります。 • 以前の説明:このプロパティ *glide.pwd_reset.onetime.token.validity* 、パスワードリセットメール内のリンクを、その *glide.pwd_reset.onetime.token.validity* プロパティで指定された時間の経過後に期限切れにすることができます。パスワードリセットトークンの有効期限は、通常のユーザーエクスペリエンスに応じてできるだけ短く保つ必要があります。パスワードリセットトークンの有効期間が長いと、ハッカーがアカウントの乗っ取りを実行するのに役立ちます。
委託開発者の読み取りアクセスを制限する (Security Center 1.3 で更新)	• 新しい簡単な説明：委託開発者の読み取りアクセスを制限する • 以前の簡単な説明： 委任開発者の読み取りアクセス許可リスト
許可される ServiceNow 内部 IP アドレスを定義する (セキュリティセンター 1.3 および 1.5 で更新)	• 新しい簡単な説明：許可される ServiceNow 内部 IP アドレスを定義する • 以前の簡単な説明：IP アドレスのアクセス許可リスト

ドキュメント	更新回数
SOAP コンテンツタイプの検証 (セキュリティセンター 1.3 で更新)	• 新しい簡単な説明：SOAP コンテンツタイプの検証 • 以前の簡単な説明：SOAP コンテンツタイプのチェック • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
Excel 要求に認証を必須とする (Security Center 1.3 で更新)	• 新しい簡単な説明：Excel 要求に対する認証を必須とする • 以前の簡単な説明：Excel 要求認証 • 新しい修正:プロパティ `glide.basicauth.required.excel` が sys_propertiesテーブルに存在し、true に設定されていることを確認します。 • 以前の修正:プロパティ `glide.basicauth.required.excel` が true に設定されていることを確認します。 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
API 要求に認証を必須とする (Security Center 1.3 で更新)	• 新しい簡単な説明：API 要求に認証を必須とする • 以前の簡単な説明：API 要求認証 • 新しい修正:プロパティ `glide.basicauth.required.api` が sys_propertiesテーブルに存在し、true に設定されていることを確認します。 • 以前の修正:プロパティ `glide.basicauth.required.api` が true に設定されていることを確認します。 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
GlideXMLUtil スクリプト可能項目のエンティティ拡張しきい値を最小化する (セキュリティセンター 1.3、1.5、および 2.0 で更新)	• 新しい簡単な説明：エンティティ拡張しきい値を最小化する • 以前の簡単な説明：エンティティ拡張しきい値の設定
パスワードリセット/変更プロセス中にユーザーに通知する (Security Center 1.5 で削除)	• 新しい簡単な説明：パスワードリセット/変更プロセス中にユーザーに通知する • 以前の簡単な説明：パスワードリセット/変更通知プロセス

ドキュメント	更新回数
	• 新しい修正: パスワードの変更またはリセット時にパスワードリセットプロセスがユーザーに通知されるようにします。 • 以前の修正:パスワードの変更またはリセット時にパスワードリセットプロセスがユーザーに通知されるようにします。
レガシー AngularJS 動作を無効にする (Security Center 1.3 で更新)	• 新しい簡単な説明： レガシー AngularJS 動作を無効にする • 以前の簡単な説明：レガシー AngularJS 動作
失敗したログインのロック解除タイムアウト期間を最大化する (セキュリティセンター 1.3 で更新)	• 新しい簡単な説明：失敗したログインのロック解除タイムアウト期間を最大化する • 以前の簡単な説明：ログイン失敗後のタイムアウトのロック解除を管理
HTTP のみの Cookie フラグを有効にする (Security Center 1.3 で更新)	• 新しい簡単な説明：HTTP のみの Cookie フラグを有効にする • 以前の簡単な説明：HTTP のみの Cookie フラグ • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
スコープ対象の管理アプリケーション ACL を有効にする (Security Center 1.3 で更新)	• 新しい簡単な説明：アドミンアプリケーション ACL 範囲を有効にする • 以前の簡単な説明：スコープ対象の管理アプリケーション ACL の管理 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
UserCookie バージョン 3.1 を有効にする (Security Center 2.0 で更新)	• 新しい説明:UserCookie v3 は、プロパティ *glide.ui.secure.cookies.use_kmf* が無効になっている場合にのみ生成されます。UserCookie v3 は、HMAC の秘密鍵をソースコードに保存し、すべての顧客に対して同一であるため、安全ではありません。これにより、悪意のあるアクターがこの1つの秘密鍵を使用してユーザーセッションを乗っ取ることができます。 • 以前の説明:UserCookie v3 は、プロパティglide.ui.secure.cookies.use_kmfが無効になっている場合にのみ生成されます。UserCookie v3 は、HMAC の秘密鍵をソースコードに保存し、すべての顧客に対して同一であるため、安全ではありません。こ

自動翻訳

ドキュメント	更新回数
	れにより、ハッカーはこの1つの秘密鍵を使用してユーザーセッションを乗っ取ることができます。
XML 要求に対する認証を必須とする (セキュリティセンター 1.3 で更新)	• 新しい簡単な説明：XML 要求に対する認証を必須とする • 以前の簡単な説明：XML 要求認証 • 新しい修正:プロパティ `glide.basicauth.required.xml` が sys_propertiesテーブルに存在し、true に設定されていることを確認します。 • 以前の修正:プロパティ `glide.basicauth.required.xml` が true に設定されていることを確認します。 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
外部ユーザー登録リンクの有効期限を最小化する (セキュリティセンター 1.3 および 1.5 で更新)	• 新しい簡単な説明：外部ユーザー登録リンクの有効期限を最小化する • 以前の簡単な説明：外部ユーザー登録リンクの有効期限
	• 新しい簡単な説明：受信メールの画像を添付ファイルに変換 • 以前の簡単な説明：受信メール HTML の変換
SMTP 受信者の数量を最小化する (セキュリティセンター 1.3 で更新)	• 新しい簡単な説明：SMTP 受信者数を最小化する • 以前の簡単な説明：最大 SMTP 受信者数
MultiSSO プラグインの更新バージョンを有効にする (Security Center 1.3 および 1.5 で更新)	• 新しい簡単な説明:更新された複数 SSO プラグインのバージョンを有効にする (プラグイン適用:複数プロバイダーのシングルサインオン) • 以前の簡単な説明：マルチ SSO プラグインの更新されたバージョンが有効 • 新しい CVSS スコア:7.1 • 以前の CVSS スコア:5
生のデータベースクエリの実行を無効にする (Security Center 1.3 で更新、2.0 で削除)	• 新しい簡単な説明：生のデータベースクエリの実行を無効にする • 以前の簡単な説明：操作レベルのアクセス管理要件

ドキュメント	更新回数
	• 新しい説明:このプロパティを使用すると、ユーザーはデータベースに対して生の SQL クエリを実行して、GlideRecord の制限外のテーブルとデータにアクセスを許可できます。`glide.db.allow_unsafe_dbi_execute_sql` が推奨値の false に設定されていない場合、Glide スクリプト可能項目から `dbi.executeStatement()` を呼び出すことができます。 • 以前の説明:このプロパティを使用すると、ユーザーはデータベースに対して生の SQL クエリを実行して、GlideRecord の制限外のテーブルとデータへのアクセス権を付与できます。`glide.db.allow_unsafe_dbi_execute_sql` が推奨値の false に設定されていない場合、Glide スクリプト可能項目から `dbi.executeStatement()` を呼び出すことができます。
XML マークアップをエスケープ (セキュリティセンター 1.3 での更新)	• 新しい簡単な説明：XML マークアップをエスケープ • 以前の簡単な説明：XML のエスケープ • 新しい修正:プロパティ `glide.ui.escape_text` がsys_propertiesテーブルに存在し、true に設定されていることを確認します。 • 以前の修正:プロパティ `glide.ui.escape_text` が true に設定されていることを確認します。 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
RSS 要求に認証を必須とする (Security Center 1.3 で更新)	• 新しい簡単な説明：RSS 要求に認証を必須とする • 以前の簡単な説明：RSS 要求認証 • 新しい修正:プロパティ `glide.basicauth.required.rss` がsys_propertiesテーブルに存在し、true に設定されていることを確認します。 • 以前の修正:プロパティ `glide.basicauth.required.rss` が true に設定されていることを確認します。 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。

ドキュメント	更新回数
添付ファイルの最大許容サイズ (Security Center 1.3 で更新)	• 新しい簡単な説明：添付ファイルの許可されるサイズを最小化する • 以前の簡単な説明：添付ファイルの許可されるサイズを最大化する
相対リンクを強制する (セキュリティセンター 1.3 および 1.5 で更新)	• 新しい説明: *glide.cms.catalog_uri_relative* プロパティは、/ess/catalog.do の URI パラメーターからの相対リンクを強制します。*glide.cms.catalog_uri_relative* が推奨値の true に設定されていない場合、URL は enforceRelativeURL(url) 関数でサニタイズされません。絶対 URL は、パラメーターまたはフィールド値の一部として使用するとセキュリティリスクを引き起こす可能性があり、ソースページが攻撃者が制御する Web サイトにリダイレクトされます。 • 以前の説明: *glide.cms.catalog_uri_relative* プロパティを使用して、/ess/catalog.do の URI パラメーターからの相対リンクを強制します。*glide.cms.catalog_uri_relative* が推奨値の true に設定されていない場合、enforceRelativeURL(url) 関数で URL がサニタイズされない可能性があります。
登録および検証の際の SMS コード通知を有効化する (セキュリティセンター 1.3 で更新)	• 新しい簡単な説明：登録および検証の際の SMS コード通知を有効化する • 以前の簡単な説明：登録および検証の SMS コード通知
キャッシュ制御 HTTP ヘッダー値 (Security Center 1.3 で更新、1.5 で削除)	• 新しい簡単な説明：HTTP ヘッダー値のキャッシュコントロール • 以前の簡単な説明：HTTP ヘッダー値のキャッシュコントロール • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
明示的な外部ロールに対して内部アクセスを拒否する (Security Center 1.3 および 1.5 で更新)	• 新しい簡単な説明:明示的な外部ロールに対して内部アクセスを拒否する • 以前の簡単な説明:明示的なロールの内部拒否リストを有効にする • 新しい技術構成名:glide.security.explicit_roles.enable_internal_user_blacklis • 古い技術構成名:glide.security.explicit_roles.enable_internal_user_blacklis

自動翻訳

ドキュメント	更新回数
	• 新しい説明:これにより、外部ユーザーにsnc_internalロールが割り当てられなくなります。*glide.security.explicit_roles.enable_interna* が推奨値の true に設定されておらず、*glide.security.explicit_roles.internal_user_* プロパティが信頼できないユーザークラスのリストに設定されていない場合は、指定されたロールに snc_external ロールの代わりに snc_internal ロールを割り当てることができます。リストが空の場合、デフォルトですべてのユーザーにsnc_internalロールが割り当てられます。プロパティには、少なくともデフォルトのロール csm_consumer_user、customer_contact が含まれている必要があります。これらのプロパティの構成を誤ると、外部ユーザー アカウントが内部情報にアクセスするリスクが高まります。 • 以前の説明:このプロパティは、外部ユーザーにsnc_internalロールをアサインできないようにします。*glide.security.explicit_roles.enable_interna* 推奨値の true に設定されている場合は、snc_externalロールを割り当てることができる *glide.security.explicit_roles.internal_user_blac property*が有効になります。値が false に設定されている場合、*glide.security.explicit_roles.internal_user_blac* プロパティが無効になります。 • 新しい修正:プロパティ *glide.security.explicit_roles.enable_internal_us* が true に設定されていること、およびプロパティ *glide.security.explicit_roles.internal_user_blac* に危険アイテム csm_consumer_user customer_contactが含まれていることを確認します。 • 以前の修正:プロパティ *glide.security.explicit_roles.enable_internal_us* が true に設定されていることを確認します。 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
ワンタイム帯域外検証のライフタイム持続期間を最小化する (Security Center 1.3 で更新)	• 新しい簡単な説明：ワンタイム帯域外検証のライフタイム持続期間を最小化する • 以前の簡単な説明：短い 1 回限りの帯域外検証ツールの期限

ドキュメント	更新回数
スクリプト要求に認証を必須とする (Security Center 1.3 で更新)	• 新しい簡単な説明:スクリプト要求に認証を必須とする • 以前の簡単な説明:スクリプト要求認証 • 新しい修正:プロパティ *glide.basicauth.required.scriptedprocessor* がsys_propertiesテーブルに存在し、true に設定されていることを確認します。 • 以前の修正:プロパティ *glide.basicauth.required.scriptedprocessor* が true に設定されていることを確認します。 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
同時インタラクティブセッションを制限する (セキュリティセンター 1.3 で更新)	• 新しい簡単な説明：同時インタラクティブセッションを制限する • 以前の簡単な説明：Glide 認証：同時インタラクティブセッションの制限 • 新しい説明:このプロパティは、同時セッション制限 (*com.glide.limit.concurrent.sessions*) プラグインで使用するためのものです。プラグインがアクティブで、プロパティが false に設定されている場合、ユーザーはインスタンス上で任意の数の同時インタラクティブセッションを使用することができます。開いているセッションの数が多いほど、セッションハイジャックが発生する可能性が高くなります。 • 以前の説明:このプロパティは、同時セッション制限 (*com.glide.limit.concurrent.sessions*) プラグインで使用するためのものです。プラグインがアクティブで、プロパティが false に設定されている場合、ユーザーはインスタンス上で任意の数の同時インタラクティブセッションを使用することができます。開いているセッションの数が多いほど、セッションハイジャックが発生する可能性が高くなります。
CSRF 検証をバイパスする警告をユーザーが受け入れるのを防ぐ (Security Center 1.3 および 1.5 で更新)	• 新しい簡単な説明：CSRF トークンの厳密な検証を強制する • 以前の簡単な説明：厳格な CSRF 検証 • 新しい説明:このプロパティは、CSRF トークンの再利用を防止する CSRF トークンの厳密な検証を有効にします。*glide.security.csrf.strict.validation.mode* が推奨値の true に設定されていない場

ドキュメント	更新回数
	合、CSRF トークンが再利用され、CSRF 攻撃の扉が開かれる可能性があります。 • 以前の説明:このプロパティは、CSRF トークンの再利用を防止する CSRF トークンの厳格な検証を有効にします。*glide.security.csrf.strict.validation.mode* が推奨値の true に設定されていない場合、CSRF トークンが再利用され、CSRF 攻撃の扉が開かれる可能性があります。
セッションアクティビティのタイムアウト時間を最小化する (セキュリティセンター 1.3 で更新)	• 新しい簡単な説明：セッションアクティビティのタイムアウト時間を最小化する • 以前の簡単な説明：セッションアクティビティのタイムアウト
HTML サニタイザーを有効にする (セキュリティセンター 1.3 で更新)	• 新しい簡単な説明：HTML サニタイザーを有効にする • 以前の簡単な説明：HTML サニタイザー
バックグラウンドスクリプトへのアクセスを制限する (Security Center 1.3 および 2.0 で更新)	• 新しい説明:このプロパティは、[スクリプトバックグラウンド] モジュールにアクセスするために必要なロールを保持します。*glide.script_processor.admin*が推奨値の admin、security_admin、または maint に設定されていない場合、低い特権ロールを持つユーザーはインスタンスでバックグラウンドスクリプトを実行できます。これにより、ACL システムが完全にバイパスされ、テーブルへのフルアクセスが可能になります。 • 以前の説明:このプロパティは、[スクリプトバックグラウンド] モジュールにアクセスするために必要なロールを保持します。*glide.script_processor.admin*が推奨値のアドミンに設定されていない場合、低い特権ロールを持つすべてのユーザーがインスタンスでバックグラウンドスクリプトを実行できます。これにより、ACL システムが完全にバイパスされ、テーブルへのフルアクセスが可能になります • 新しい修正:プロパティ *glide.script_processor.admin* が admin、security_admin、または maint ロールに設定されていることを確認します。

ドキュメント	更新回数
	• 以前の修正:プロパティ *glide.script_processor.admin* が [アドミン] に設定されていることを確認します。 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
埋め込み HTML コードを無効化する (セキュリティセンター 1.3 で更新)	• 新しい簡単な説明：埋め込み HTML コードを無効にする • 以前の簡単な説明：埋め込み HTML コードを無効にする
絶対的なセッションタイムアウト時間を最小化する (セキュリティセンター 1.3 で更新)	• 新しい簡単な説明：セッションタイムアウトの絶対時間を最小化する • 以前の簡単な説明：絶対的なセッションタイムアウト
クライアント呼び出し可能スクリプトインクルードにはデフォルトで認証を必須とする (Security Center 1.3 で更新)	• 新しい簡単な説明：クライアント呼び出し可能スクリプトインクルードにはデフォルトで認証を必須とする • 以前の簡単な説明：クライアント呼び出し可能スクリプトインクルードのプライバシー
GlideSystemUserSession スクリプト作成可能 API へのアクセスを制限する (セキュリティセンター 1.3 および 2.0 で更新)	• 新しい簡単な説明:GlideSystemUserSession スクリプト作成可能 API へのアクセスを制限する • 以前の簡単な説明:GlideSystemUserSession スクリプト可能 API へのアクセス
HTML サニタイゼーションを強制する (Security Center 1.3 で更新)	• 新しい簡単な説明：HTML のサニタイズを強制する • 以前の簡単な説明：サニタイズされていない HTML をチェック • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
絶対的なセッションタイムアウト時間を最小化する (セキュリティセンター 1.3 で更新)	• 新しい簡単な説明：セッションタイムアウトの絶対時間を最小化する • 以前の簡単な説明：絶対的なセッションタイムアウト

ドキュメント	更新回数
ロールベースのマルチファクター認証を有効にする (Security Center 1.3 で更新)	• 新しい簡単な説明：ロールベースのマルチファクター認証を有効にする • 以前の簡単な説明：ロールベースのマルチファクター認証
SAML の「notBefore」または「notOnOrAfter」の制約期間を最小化する (セキュリティセンター 1.3 および 1.5 で更新)	• 新しい簡単な説明:SAML の「notBefore」または「notOnOrAfter」の制約期間を最小化する (プラグイン適用:複数プロバイダーのシングルサインオン) • 以前の簡単な説明：SAML の「notBefore」または「notOnOrAfter」制約 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
外部ユーザー登録用の電子メールドメインを制限する (セキュリティセンター 1.3、1.5、および 2.0 で更新)	• 新しい簡単な説明:外部ユーザー登録用の電子メールドメインを制限する (プラグインの適用性:外部ユーザー登録) • 以前の簡単な説明：外部ユーザー登録メールドメインの許可リスト • 新しい修正:プロパティ *sn_ext_usr_reg.allowed_email_domains* が空の値に設定されていないことを確認します。 • 以前の修正:プロパティ *sn_ext_usr_reg.allowed_email_domains* が空の値に設定されていないことを確認します。
パスワードリセット SMS の一時停止ウィンドウの持続期間を最大化する (セキュリティセンター 1.3 で更新)	• 新しい簡単な説明：パスワードリセット SMS の一時停止ウィンドウの持続期間を最大化する • 以前の簡単な説明：パスワードリセット SMS の一時停止ウィンドウ • 新しい修正:プロパティ *password_reset.sms.pause_window* が 2 以上に設定されていることを確認します。 • 以前の修正:プロパティ *password_reset.sms.pause_window* が 2 に設定されていることを確認します。 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。

ドキュメント	更新回数
発信 SSLv2/SSLv3 接続を無効化する (Security Center 1.3 で更新)	• 新しい簡単な説明：発信 SSLv2/SSLv3 接続を無効にする • 以前の簡単な説明：SSLv2/SSLv3 の無効化
アンロード要求に認証を必須とする (Security Center 1.3 で更新)	• 新しい簡単な説明：アンロード要求に認証を必須とする • 以前の簡単な説明：アンロード要求認証 • 新しい修正:プロパティ `glide.basicauth.required.unl` が sys_properties_tableに存在し、true に設定されていることを確認します。 • 以前の修正:プロパティ `glide.basicauth.required.unl` が true に設定されていることを確認します。 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
スパムメールのスコアリングとフィルタリングを有効化する (Security Center 1.3 で更新)	• 新しい簡単な説明： スパムメールのスコアリングとフィルタリングを有効にする • 以前の簡単な説明： スパムメールのスコアリングとフィルタリング
LDAP の初期識別名の設定を解除する (Security Center 1.3 で更新、2.0 で削除)	• 新しい簡単な説明：LDAP の初期識別名の設定を解除 • 以前の簡単な説明：LDAP の初期識別名
Anti-CSRF トークンを有効にする (Security Center 1.3 の新機能、1.5 で更新、2.0 で削除)	• 新しい簡単な説明：Anti-CSRF トークンを有効にする • 以前の簡単な説明：Anti-CSRF トークン
AJAXGlideRecord ACL チェックを必須とする (Security Center 1.3 で更新)	• 新しい簡単な説明：AJAXGlideRecord ACL チェックを必須とする • 以前の簡単な説明：AJAXGlideRecord ACL チェックの有効化
ユーザーの代理操作をログ記録 (Security Center 1.3 および 2.0 で更新)	ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。検出精度を向上させるためにスクリプトが更新されました。
感染したファイルのダウンロードを許可しない (セキュリティセンター 1.5 および 2.0 で更新)	ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。

ドキュメント	更新回数
外部ユーザー登録向けに Captcha を有効にする (Security Center 1.3 および 1.5 で更新)	• 新しい簡単な説明:外部ユーザー登録で Captcha を有効にする (プラグイン適用:外部ユーザー登録) • 以前の簡単な説明:外部ユーザー登録向けに Captcha を有効にする
SQL エラーメッセージを無効にする (Security Center 1.3 および 1.5 で更新)	• 新しい簡単な説明： SQL エラーメッセージを無効にする • 以前の簡単な説明：SQL エラーメッセージを無効にする
パスワードリセット要求の有効期限を最小化する (セキュリティセンター 1.3 で更新)	• 新しい簡単な説明：パスワードリセット要求の有効期限を最小化する • 以前の簡単な説明：パスワードリセット要求の有効期限 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
無効なパスワードリセット試行に対するロックアウト時間を制御する (Security Center 1.3 および 2.0 で更新)	• 新しい簡単な説明：パスワードリセット要求の試行ウィンドウの最大持続期間を最小化する • 以前の簡単な説明：パスワードリセット要求の最大試行ウィンドウ
ダウンロード可能な MIME タイプを制限する (セキュリティセンター 1.3 および 2.0 で更新)	• 新しい簡単な説明：ダウンロード可能な MIME タイプを制限する • 以前の簡単な説明：ダウンロード可能な MIME タイプの拒否リスト
Excel 計算式をエスケープ (Security Center 1.3 で更新)	• 新しい簡単な説明：Excel 式をエスケープ • 以前の簡単な説明：Excel 式をエスケープ
コンテキストセキュリティプラグインを有効にする (Security Center 1.3 で更新)	• 新しい簡単な説明：コンテキスト依存セキュリティプラグインを有効にする • 以前の簡単な説明：コンテキスト依存セキュリティプラグイン
アカウント復旧の有効化 (セキュリティセンター 1.3 および 1.5 で更新)	• 新しい簡単な説明:アカウント復旧の有効化 (プラグイン適用:複数プロバイダーのシングルサインオン) • 以前の簡単な説明：アカウント復旧

ドキュメント	更新回数
	• 新しい説明:このプロパティは、シングルサインオンをバイパスする機能を特別に指定されたアドミニストレーターにバインドするアカウント復旧機能を制御します。*glide.sso.acr.enabled* が推奨値の true に設定されていない場合、インスタンスでシングルサインオンが有効になっていても、ローカルインタラクティブログイン (ユーザー名またはパスワードベース) は有効なままになります。ローカルのインタラクティブログインを排除することで、インスタンスへの不正アクセスの可能性が減少します。 • 以前の説明:このプロパティは、アカウント復旧機能を制御します。*glide.sso.acr.enabled*が推奨値の true に設定されていない場合、userId によるアカウント復旧は実行できません。 • 新しい CVSS スコア:6.5 • 以前の CVSS スコア:9.1 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
インポート要求に認証を必須とする (Security Center 1.3 で更新)	• 新しい簡単な説明：インポート要求に対する認証を必須とする • 以前の簡単な説明：インポート要求認証 • 新しい修正:プロパティ *glide.basicauth.required.importprocessor* がsys_propertiesテーブルに存在し、true に設定されていることを確認します。 • 以前の修正:プロパティ *glide.basicauth.required.importprocessor* が true に設定されていることを確認します。 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
SNC アクセスコントロールプラグインを有効化する (セキュリティセンター 1.3 で更新)	• 新しい簡単な説明：SNC アクセスコントロールプラグインを有効化する • 以前の簡単な説明：SNC アクセスコントロールプラグイン
全ノードを対象とした同時セッション制限 (Security Center 1.3 で更新)	• 新しい簡単な説明：全ノードを対象とした同時セッション制限 • 以前の簡単な説明：Glide 認証：全ノードを対象とした同時セッション制限

ドキュメント	更新回数
XML 出力要求に対する認証を必須とする (Security Center 1.3 で更新)	• 新しい簡単な説明：XML 出力要求に対する認証を必須とする • 以前の簡単な説明：XML 出力認証 • 新しい修正:プロパティ *glide.basicauth.required.xmloutputprocessor* がsys_propertiesテーブルに存在し、true に設定されていることを確認します。 • 以前の修正:プロパティ *glide.basicauth.required.xmloutputprocessor* が true に設定されていることを確認します。 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
Scratchpad のスクリプトをエスケープ (セキュリティセンター 1.3 で更新)	• 新しい簡単な説明：Scratchpad のスクリプトをエスケープ • 以前の簡単な説明：Scratchpad をエスケープ • 新しい説明:スクラッチパッドは、ブラウザでアクセスできるサーバー上の情報を設定する簡単な方法です。アドミニストレーターは、任意のレコードの任意のデータなど、あらゆるものをスクリプト化して配置できます。*glide.ui.escape_scratchpad*が推奨値の true に設定されていない場合、クロスサイトスクリプティングの脆弱性のような悪意のあるスクリプトが実行される可能性があります。 • 以前の説明:スクラッチパッドは、ブラウザでアクセスできるサーバー上の情報を設定する簡単な方法です。アドミニストレーターは、任意のレコードの任意のデータなど、あらゆるものをスクリプト化して配置できます。*glide.ui.escape_scratchpad* が推奨値の true に設定されていない場合、クロスサイトスクリプティングの脆弱性のような悪意のあるスクリプトが実行される可能性があります。
WSDL 要求に認証を必須とする (セキュリティセンター 1.3 および 1.5 で更新)	• 新しい簡単な説明：WSDL 要求に対する認証を必須とする • 以前の簡単な説明：WSDL 要求認証 • 新しい修正:プロパティ *glide.basicauth.required.wsdl* が sys_propertiesテーブルに存在し、true に設定されていることを確認します。

ドキュメント	更新回数
	• 以前の修正:プロパティ *glide.basicauth.required.wsdl* が true に設定されていることを確認します。 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
スキーマ要求に認証を必須とする (Security Center 1.3 で更新)	• 新しい簡単な説明：スキーマ要求に対する認証を必須とする • 以前の簡単な説明：スキーマ要求認証 • 新しい修正:プロパティ *glide.basicauth.required.schema* が sys_propertiesテーブルに存在し、true に設定されていることを確認します。 • 以前の修正:プロパティ *glide.basicauth.required.schema* が true に設定されていることを確認します。 • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
ダウンロード可能な MIME タイプを制限する (セキュリティセンター 1.3 および 2.0 で更新)	• 新しい簡単な説明：ダウンロード可能な MIME タイプを制限する • 以前の簡単な説明：ダウンロード可能な MIME タイプ
スクリプトサンドボックスで優先度「低」のユーザーのロガーを無効にする (Security Center 1.3 で更新)	• 新しい簡単な説明:スクリプトサンドボックスで優先度「低」のユーザーのロガーを無効化する • 以前の簡単な説明:Glide セキュリティロガーがサンドボックスにログインしない • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
X-Frame-Options：SAMEORIGIN セキュリティヘッダーを実装 (セキュリティセンター 1.3 で更新)	• 新しい簡単な説明：X-Frame-Options: SAMEORIGIN セキュリティヘッダーを実装 • 以前の簡単な説明：X-Frame-Options: SAMEORIGIN • ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
パフォーマンス監視のアクセスを制限する (Security Center 1.3 で更新)	• 新しい簡単な説明:パフォーマンスモニタリングのアクセスを制限する • 以前の簡単な説明:パフォーマンスモニタリング ACL

ドキュメント	更新回数
インポートプロセッサの詳細な SQL エラーメッセージをオフにする (Security Center 1.3 で更新)	ルールスクリプト:スクリプトが更新され、検出の精度が向上しました。
パスワードリセットの SMS 有効期限を最小化する (セキュリティセンター 1.3 で更新)	• 新しい簡単な説明：パスワードリセットの SMS 有効期限を最小化する • 以前の簡単な説明：パスワードリセット SMS の有効期限
受信メールからユーザーを作成するのを無効化する (セキュリティセンター 1.3 で更新)	• 新しい簡単な説明：受信メールからユーザーを作成するのを無効にする • 以前の簡単な説明：ドメインでメールを制限する • 新しい説明:アドミニストレーターは、受信メールからユーザーを自動的に作成するようにメールプロパティを設定できます。このプロパティを安全でない値に設定すると、インスタンスは受信メールから自動的にユーザーを作成します。作成された各ユーザーは、ハードコードされた同じデフォルトのパスワードを持つため、ブルートフォースによる認証のバイパスが容易になります。 • 以前の説明:アドミニストレーターは、受信メールからユーザーを自動的に作成するようにメールプロパティを設定できます。このプロパティを安全でない値に設定すると、インスタンスは受信メールから自動的にユーザーを作成します。作成された各ユーザーは、ハードコードされた同じデフォルトのパスワードを持つため、ブルートフォースによる認証のバイパスが容易になります。 • 新しい修正:プロパティ *glide.pop3readerjob.create_caller* が false に設定されていることを確認します。 • 以前の修正:プロパティ *glide.pop3readerjob.create_caller* が false に設定されていることを確認します

削除されたハードニング設定

一部のハードニング設定がセキュリティセンターのベースラインから削除されます。

ベースラインバージョン **6.0** の削除されたハードニング設定

セキュリティセンターベースラインバージョン 6.0 のリリースでは、一部のハードニング設定が削除されています。

- HR ライフサイクルイベントデータに ACL を強制する
- セキュリティスコープライセンスと許可プレイブックを適用

- HR コアデータに ACL を強制する
- グローバル管理者がスコープ対象アプリのアクセス制限をバイパスしないように制限する
- レガシー AngularJS 動作を無効にする
- GlideRecordSandbox でフィールドレベルの ACL を強制する
- 公共機関デジタルサービスのスコープセキュリティを強制する
- 静的コンテンツでダウンロード可能なファイルの種類を制限する
- 情報要求プレイブックに対してスコープされた ACL アクセスを強制する
- SAML の「notBefore」または「notOnOrAfter」の制約期間を最小化する
- 委託開発者の読み取りアクセスを制限する
- すべてのメソッドにパスワードポリシーを強制する
- サービスアプリケーション情報にセキュリティスコープを適用する
- クラシックモバイルアプリ UI の難読化を必須とする
- LDAP 認証で SSL を有効にする
- HR ケース管理のエージェントワークスペースのセキュリティスコープを適用
- トレーニングと予測フローの添付ファイルのサイズを制限する
- ようこそページから認証情報を削除する
- HR 仮想エージェントデータに ACL を強制する
- HR アプリの ACL 評価で代理操作をチェックする
- 許可された Java パッケージを制限する
- 厳格なコード署名チェックを強制する

ベースラインバージョン **5.0** の削除されたハードニング設定

Security Center ベースラインバージョン 5.0 のリリースでは、一部のハードニング設定が削除されています。

- イベント管理 HTTP プロセッサーで認証を必須とする
- カスタムジャーナルエントリへのアクセスを制限する
- ログイン時にパスワードポリシーを適用しない
- 生のデータベースクエリの実行を無効にする
- HTML サニタイズをログに記録
- Anti-CSRF トークンを有効にする
- 認証信用を強制する
- LDAP の初期識別名の設定を解除
- パスワードの最小長を設定

ベースラインバージョン **4.0** の削除されたハードニング設定

一部のハードニング設定は、ベースラインバージョン 4.0 のリリースで削除 セキュリティセンター 。

- LDAP 初期パスワード
- Mobile Offline ロール
- Zero Trust アクセスポリシーを使用して重要なデータへのアクセスを制限する (プラグイン適用: 適応認証)
- 読み取り専用テーブル 書き込みの許可リスト
- 許可された JDBC プローブ操作 (プラグイン適用:MID サーバー)
- 受信メールからユーザーを作成する許可ドメインを設定する
- 複雑な「デフォルト」パスワードを設定する
- GraphQL 許可リストのアクセス可能なプロパティ
- クロスオリジンメッセージング許可リスト
- GraphQL エンドポイントのトレーニングおよび予測フローの添付ファイルのサイズを制限する (プラグイン適用:プラットフォームドキュメントインテリジェンス (DocIntel))
- コンテンツロール許可リストの編集 (プラグイン適用:Communities)
- クエリサイズが原因でデータベースクエリが OutofMemory 例外をトリガーしないようにする
- スクリプト実行のロール許可リスト
- ダウンロード可能なファイルタイプの許可リスト
- 読み取り専用テーブルの削除許可リスト
- ロールレス ACL の認証の適用
- LDAP ワンタイムトークンの有効期限を最小化する
- 信頼できる IP アドレスのみに認証を許可する
- モバイルパスワードリセット用 URL を設定
- サービスアカウントのパスワードの複雑さ (プラグイン適用:サービスブリッジ)
- パスワードリセット/変更プロセス中にユーザーに通知する
- アプリケーションスコープの制限を強制する
- アクセス制御要件 (プラグイン適用:Communities)
- レコード履歴アクセスロール許可リスト
- 添付ファイルのロールアクセスの制限
- 信頼できる URL の事前定義されたリストからの PDF のみを許可
- LDAP サーバーの停止中にワンタイムパスワードのメール送信を防止する
- 受信メールの画像を添付ファイルに変換
- 無関係な明示的なロールアクセス制御条件の確認

ベースラインバージョン **2.0** の削除されたハードニング設定

一部のハードニング設定は、ベースラインバージョン 2.0 のリリースで削除 セキュリティセンター 。

- アプリケーション構成データとスクリプトのコード署名を有効にする
- Glide KMF エンクリプターを有効にする
- インスタンスレベルのエンクリプターの使用を無効にする

- 明示的なロールの内部拒否リストを有効にする
- コード署名要件のために MID ECC キューへの直接挿入をブロック
- あらゆるアウトバウンド HTTP 要求フィールドをログ記録

アクセス制御

アクセス制御カテゴリは、権限モデルに基づいて要求を許可および拒否することで、権限のないアクセスからリソースを保護するプロセスを監査します。これには、リソースにアクセスするエンティティがそれを行うための有効な認証情報を保持していることの確認、明確に定義されたロールまたは権限のセットの作成および保護、ロールまたは権限のコントロールが再現や改ざんから保護されていることの確認が含まれます。

アクセス制御は、特定のリソースへのアクセスを許可するか拒否するかを決定します。リソースの使用が許可されているユーザーのみに、そのリソースへのアクセスが許可されます。

CSRF 対策トークンの検証時間 (Security Center 1.3 の新機能)

glide.security.csrf_previous.time_limit プロパティは、保護トークンが期限切れになるまでの時間 (秒数) を指定します。

ユーザーセッションが期限切れになると、[失効したトークンの再使用を許可] プロパティが有効になっている場合を除いて、このプロパティで指定された期間内に保護トークンも期限切れになります。このトークンはクロスサイトリクエストフォージェリ攻撃を防ぐために使用されます。

詳細情報

属性	説明
構成名	*glide.security.csrf_previous.time_limit*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	整数
推奨値	86400 秒または 1 日
デフォルト値	86400 秒または 1 日
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：5.3 • CVSS スコア：中 • セキュリティリスクの詳細：このプロパティを推奨値に設定しないと、クロスサイトリクエストフォージェリ攻撃を防ぐために使用されるトークン検証が無効になります。
依存関係と前提条件	なし
機能への影響度	このプロパティは、保護トークンが有効な期間を秒単位で決定します。[失効したトークンの再使用を許可] プロパティが有効になっている場合を除いて、ユーザーセッションが期限切れになると、このプロパティで指定された期間内に保護トークンが期限切れになります。このトークンは

属性	説明
	クロスサイトリクエストフォージェリ攻撃を防ぎます。デフォルト値は 86,400 秒または 1 日です。

ドット連結フィールドにドメインセパレーションを適用する (セキュリティセンター 1.3、1.5、および **2.0** で更新)

glide.sys.domain.include_domain_condition_on_join プロパティは、ドット連結フィールドのドメインセパレーション機能を確実に適用するために、結合クエリにドメインセパレーション条件を指定するかどうかを制御します。

このプロパティは、ドット連結フィールドにドメインセパレーション機能を確実に適用するために、結合クエリにドメインセパレーション条件を指定するかどうかを制御します。ドメインセパレーションを使用しているインスタンスで *glide.sys.domain.include_domain_condition_on_join* が推奨値である true に設定されていない場合、特定のドメインと共有されない機密情報が公開される可能性があります。コンポーネントが安全でないクロスドメインクエリに依存している場合、インスタンスへの機能への影響は中程度になる可能性があります。インスタンスは、有効にする前に準本番環境でテストする必要があります。

⚠ **警告:** これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

詳細情報

属性	説明
プロパティ名	*glide.sys.domain.include_domain_condition_on_join*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	ドメインセパレーションがインストールされている場合は true。それ以外の場合、このプロパティは存在しません。
デフォルト値	false
カテゴリ	アクセス制御
目的	ドット連結フィールドのドメインセパレーション機能を確実に適用するために、結合クエリにドメインセパレーション条件を指定するかどうかを制御します。
セキュリティリスク	• 重大度スコア：6.5 • CVSS スコア：中 • セキュリティリスクの詳細：*glide.sys.domain.include_domain_condition_on_join* が推奨値である **true** に設定されていない場合、特定のドメインと共有されない機密情報が公開される可能性があります。
参照	サービスプロバイダーのドメインセパレーション

委任開発者のアクセスをブロック

この設定は、スクリプトを使用してユーザーロールを更新する委任開発者のアクセスに影響します。設定が準拠している場合、開発者は user_admin ロールなしで sys_user_has_role テーブルのレコードを更新したり挿入したりすることはできません。

このプロパティの値は、インスタンス内の機能への予期しないアクセスを委任開発者が許可できるかどうかに影響します。プロパティにロールが含まれている場合は、そのロールのみがスクリプトのモジュールを実行できます。

詳細情報

属性	説明
プロパティ名	*com.glide.sys.security.delegateddev.block_grant_roles*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	アクセス制御
目的	このプロパティの値は、インスタンス内の機能への予期しないアクセスを委任開発者が許可できるかどうかに影響します。
タイプ	切り替えスイッチ
デフォルト値	true
推奨値	true
セキュリティ依存関係	なし
セキュリティリスク評価	6.7
機能への影響	delegated_developer ロールを持つユーザーが sys_user_has_roleテーブルのレコードを変更しようとすると、このプロパティによって操作に対する追加のセキュリティチェックが有効になります。ユーザーが sys_user_has_role テーブルを作成または更新しようとしている場合、user_admin ロールがそのユーザーに付与されていることが追加のセキュリティチェックによって検証されます。ユーザーに user_admin ロールがない場合、アクセスは拒否されます。プロパティが false の場合、これらの追加チェックは検証されません。
セキュリティリスク	(中) 適切な承認がないと、無許可のユーザーがインスタンスの機密コンテンツ/データにアクセスする可能性があります。
参照	アクセス制御

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

期限切れの **Anti-CSRF** トークンをブロック (セキュリティセンター **1.5** で更新)

クロスサイトリクエストフォージェリ攻撃を防ぐために期限切れの CSRF トークンをブロックします。

概要

クロスサイトリクエストフォージェリは、認証されたユーザーに代わって不正なコマンドを実行する悪意のあるエクスプロイトの一種です。

構成詳細

属性	説明
概要	期限切れのセキュアトークンの使用をコントロールして、受信要求を識別して検証します。以前に期限切れになったトークンを使用して受信要求を検証しないようにするには、**false** に設定します。
構成名	*glide.security.csrf_previous.allow*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	**false**
デフォルト値	**true**
カテゴリ	アクセス制御
セキュリティリスク	重大度スコア：6.5
	CVSS スコアに応じた重大度：中
	セキュリティリスクの詳細：認証された機能を保護するために強力な CSRF 対策メカニズムを適用し、効果的な自動化対策または CSRF 対策で認証されていない機能を保護します。
依存関係と前提条件	なし
参照	Anti-CSRF トークンを有効にする (Security Center 1.3 の新機能、1.5 で更新、2.0 で削除)、 クロスサイトリクエストフォージェリ 。

実行前の UI アクションの条件のチェック

glide.security.strict.actions プロパティを使用して、実行前にフォームおよびリスト内の UI アクションの条件を確認できるようにします。このプロパティを true に設定すると、実行前にテーブル UI アクションに検証レイヤーが追加されます。

詳細情報

属性	説明
プロパティ名	*glide.security.strict.actions*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	アクセス制御

属性	説明
目的	実行前にテーブル UI アクションに検証レイヤーを確実にさらに追加すること
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
セキュリティリスク評価	3.3
機能への影響	この修正では、インスタンスのターゲットテーブル/ページで UI アクションをチェックするための検証レイヤーが追加されるだけです。アクセス制御が顧客インスタンスで適切に設定されているかぎり、ここに影響はありません。
セキュリティリスク	(低) 2 つのゾーンの間でトランザクションが発生した場合は常に、アクセス要求がチェックされます。この操作により、エンドユーザーに対してフォームがレンダリングされる前に UI アクションが検証されます。
参照	高セキュリティプラグイン

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

イベント管理アサイン先グループアドミンロールの構成 (Security Center 1.5 の新機能)

evt_mgmt.connector_assignment_group_admin_roles プロパティを使用して、コネクタインスタンスのアサイン先グループフィールドへのアドミンアクセスをどのロールに許可するかを設定します。

evt_mgmt.connector_assignment_group_admin_roles プロパティには、コネクタインスタンスのアサイン先グループフィールドへのアドミンアクセス権を持つロール名を示すカンマ区切りの文字列が含まれます。このリストのデフォルトのロールを変更すると、権限のないユーザーがインスタンス上のイベント統合を変更できる可能性があります。ロールへの不正なアクセスを防ぐには、*evt_mgmt.connector_assignment_group_admin_roles* を admin,evt_mgmt_admin,sn_sow_srm.srm_admin の値に設定します。推奨値文字列内の追加のロールを確認して、そのロールが含まれていることを確認します。

詳細情報

属性	説明
構成名	*evt_mgmt.connector_assignment_group_admin_roles*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	文字列
推奨値	admin,evt_mgmt_admin,sn_sow_srm.srm_admin
デフォルト値	admin,evt_mgmt_admin,sn_sow_srm.srm_admin
カテゴリ	アクセス制御

属性	説明
セキュリティリスク	• 重大度スコア：3.1 • CVSS スコア：低 • セキュリティリスクの詳細：デフォルトのロールを変更すると、権限のないユーザーがインスタンス上のイベント統合を変更できる可能性があります。
依存関係と前提条件	なし
参照	グループの作成

サービスポータルウィジェット許可リストを構成する (Security Center 2.0 の新機能)

テーブルのアクセス制御リスト (ACL) が機密情報を公開しないように、glide.service_portal.widget.allow_list プロパティを安全に構成する方法について説明します。

glide.service_portal.widget.allow_list プロパティは、インスタンス内の任意のテーブルにアクセスできるウィジェットを識別します。ただし、これらのテーブルのアクセス制御リスト (ACL) は引き続き適用されます。ACL が正しく構成されていないか存在しない場合、このリストのウィジェットによってこれらのテーブルへのアクセスが可能になり、機密情報が公開される可能性があります。このプロパティは、ウィジェットが SNCACLWidgetUtil を使用し、*glide.service_portal.widget.enforce_public_check* プロパティが有効になっている (true に設定されている) 場合にのみ有効です。

詳細情報

属性	説明
構成名	*glide.service_portal.widget.allow_list*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	アレイ
推奨値	空
デフォルト値	空:一部の顧客のケースでは、いくつかの値が存在する可能性があります。
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：3.7 • CVSS スコア：低 • セキュリティリスクの詳細:このプロパティを推奨値に設定しないと、ウィジェットがインスタンス内の任意のテーブルにアクセスできる可能性があります。
依存関係と前提条件	*glide.service_portal.widget.allow_list*設定を適用するには、*glide.service_portal.widget.enforce_public_check*プロパティを true に設定する必要があります。

属性	説明
機能への影響	ウィジェットが公開に設定されていて、プロパティの値に含まれている場合、このプロパティを使用すると、顧客はテーブルの情報にアクセスできます。

サービスポータルウィジェットテーブル許可リストを構成する (Security Center 2.0 の新機能)

追加のチェックと特定の Glide プロパティ設定に応じて、サービスポータルウィジェットを介して非認証ユーザーがアクセスできるテーブルをリストすることで、*glide.service_portal.widget.table_allow_list* プロパティがどのようにセキュリティを強化するかについて説明します。

glide.service_portal.widget.table_allow_list プロパティには、SNCACLWidgetUtil スクリプトの追加のセキュリティチェックを利用するサービスポータルウィジェットを介して、非認証ユーザーがアクセスできるテーブルのリストが含まれています。このプロパティは、glide プロパティ *glide.service_portal.widget.enforce_public_check* が true に設定されている場合にのみ適用されます。このプロパティに不要なテーブルを含めると、機密情報が開示される可能性があります。ただし、テーブル ACL は以前と同様に引き続き評価されます。

詳細情報

属性	説明
構成名	*glide.service_portal.widget.table_allow_list*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	false
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：3.7 • CVSS スコア：低 • セキュリティリスクの詳細:このプロパティが安全な値に設定されていない場合、不要なテーブルが含まれ、機密情報が開示される可能性があります。
依存関係と前提条件	*glide.service_portal.widget.table_allow_list*設定を有効にするには、*glide.service_portal.widget.enforce_public_check* プロパティを true に設定する必要があります。
機能への影響	テーブルリストは、ウィジェットがデータを取得できるテーブルへのアクセスを制御します。

明示的な外部ロールに対して内部アクセスを拒否する (Security Center 1.3 および 1.5 で更新)

システムプロパティを使用して、外部ユーザーにsnc_internalロールをアサインできるかどうかを決定します。

glide.security.explicit_roles.enable_internal_user_blacklist システムプロパティを使用して、外部ユーザーに *snc_internal* ロールがアサインされないようにします。このプロパティが **true** に設定されている場合、メンテナンスで保護された *glide.security.explicit_roles.internal_user_blacklist* プロパティのパラメーターが適用されます。このプロパティは、信頼できないユーザークラスのリストに *snc_external* ロールを割り当てます。*glide.security.explicit_roles.enable_internal_user_blacklist*が **false** に設定されている場合、*glide.security.explicit_roles.internal_user_blacklist* プロパティは無視されます。

注: 明示的なロールがインストールされていないインスタンスは影響を受けません。Paris リリース以降、明示的なロールの新規インストールでは、プロパティにはデフォルト値 true が指定されます。

詳細情報

属性	説明
構成名	*glide.security.explicit_roles.enable_internal_user_blacklist*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
フォールバック値	false
カテゴリ	セッション管理
セキュリティリスク	• 重大度スコア:5.4 • CVSS スコア：中 • このプロパティの構成を誤ると、外部ユーザー アカウントが内部情報にアクセスするリスクが高まります。
依存関係と前提条件	なし

要求アイテムへの不正アクセスを拒否する **(Security Center 1.3** で更新)

glide.sc.req_for.roles.default プロパティは、*retrieveAddress* API のデフォルトの動作を定義します。

このプロパティは、*glide.sc.req_for.roles* に値がない場合にのみ機能します。*glide.sc.req_for.roles* に値がある場合、このプロパティは重要ではなく、定義されたロールのみを持つユーザーに API へのアクセス権が付与されます。

詳細情報

属性	説明
プロパティ名	*glide.sc.req_for.roles.default*

構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	アクセス制御
目的	プロパティでロールが指定されていない場合、特権のないログインユーザーが、クライアント呼び出し可能スクリプトインクルード *ScriptServiceCatalogGetLocation* を呼び出して、システム内の他のユーザーのアドレスを取得できます。このプロパティは、この API が特権のないユーザーに公開されるのを防ぎます。
推奨値	拒否
デフォルト値	拒否
構成タイプ	選択リスト (許可 | 拒否)
セキュリティリスク	(中) *glide.sc.req_for.roles.default* が推奨値 **deny** (許可) に設定されておらず、*glide.sc.req_for.roles* の値が空の場合、どのユーザーも他のユーザーのアイテムを要求できるようになり、不正なリソースアクセスが許可されます。
参照	クライアント呼び出し可能スクリプトインクルード

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

高リスク **UI** ページの推奨事項の表示

リスクの高い UI ページの推奨事項を表示することで、認証エラーや意図しない情報漏えいの可能性を減らします。

次の場合に、 **glide.script.ui_page.customer_scoped.security_msgs_enabled** システムプロパティを使用して、UI ページを構成するユーザーにセキュリティの推奨事項を表示するかどうかを決定します。

- ACL がありません
- GlideRecordSecure の代わりに GlideRecord/GlideDBQuery API が使用される
- ページは「sys_public」テーブルで公開として構成されています

プロパティを有効にすると、前述の条件が満たされたときの推奨事項により、認証エラーの可能性が減少し、意図しない情報漏えいを防ぐことができます。

これらの推奨事項を表示するには、システムプロパティ **glide.script.ui_page.customer_scoped.security_msgs_enabled** を **true** に設定します。

詳細情報

属性	説明
技術的な構成名	glide.script.ui_page.customer_scoped.security_msgs_enabled
プラグインの適用性	なし
セキュリティリスク	このシステムプロパティを **false** に設定すると、認証エラーや意図しない情報が開示される可能性が高くなります。
共通脆弱性スコアリングシステム (CVSS) スコア	5.3
共通脆弱性スコアリングシステム (CVSS) 評価	中
機能への影響	UI ページを構成するユーザーにセキュリティに関する推奨事項を表示します。
依存関係と前提条件	なし
データタイプ	ブール
ベースシステム値	true
フォールバック値	true
推奨値	true

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

ロックアウトされたユーザーの受信メールを無効にする

glide.pop3.process_locked_out プロパティを使用して、ロックアウトされたアクティブなユーザーの受信メールアクションを制御します。

ロックアウトされたユーザーの受信メールを無効にするには、このプロパティを **false** に設定します。

注: 信頼できないドメインのユーザーからのメールによる受信メールアクションのトリガーを許可する前に、それらのユーザーを許可する場合のセキュリティへの影響とユーザーがロックアウトされている理由を検討してください。

詳細情報

属性	説明
プロパティ名	*glide.pop3.process_locked_out*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	アクセス制御
目的	このプロパティは、ロックアウトされているユーザーの受信メールアクションを制御します。
データタイプ	ブーリアン
推奨値	false

属性	説明
デフォルト値	false
セキュリティリスク	(高) このプロパティを **true** に設定した場合、ロックされたアカウントを持つユーザーが受信メールを受信するため、情報が公開される可能性があります。
セキュリティリスク評価	7.5

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

受信トランザクションを再確認する (Security Center 1.3 で更新)

glide.security.strict.updates プロパティを使用して、フォームの送信時に受信トランザクションのセキュリティのダブルチェックを有効にします。このプロパティを **true** に設定すると、フォームがブラウザーでレンダリングされる前に、テーブル検証レイヤーがさらに追加されます。

プロパティ *glide.security.strict.updates* がsys_propertiesテーブルに存在し、true に設定されていることを確認します。

詳細情報

⚠ **警告:** これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

属性	説明
プロパティ名	*glide.security.strict.updates*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	アクセス制御
目的	ブラウザーでフォームを表示する前に、ユーザー権限の検証レイヤーを確実に追加すること
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
セキュリティリスク評価	8.1
機能への影響	この修正では、インスタンスのターゲットテーブル/ページでユーザー権限をチェックするための検証レイヤーが追加されます。アクセス制御が顧客インスタンスで適切に設定されているかぎり、影響はありません。
セキュリティリスク	(高) 2 つのゾーンの間でトランザクションが発生した場合は常に、アクセス要求をチェックする必要があります。この操作では、フォームが要求されたとき、およびフォームがレンダリングされる前に、権限がチェックされます。
参照	

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

スコープ対象の管理アプリケーション ACL を有効にする (Security Center 1.3 で更新)

glide.security.scoped_administration.honor_global_acl は、アプリケーション管理アプリがグローバルアクセス制御リスト (ACL) ルールを継承できるかどうかを決定します。

このプロパティは、スコープ対象の管理アプリケーション ACL がレコードスコープに対して定義されていない場合に特に役立ちます。

アプリケーションへの権限を持つ低い特権のユーザーが機密レコードにアクセスすることを防ぐには、*glide.security.scoped_administration.honor_global_acl* を true に設定します 。

詳細情報

属性	説明
プロパティ名	*glide.security.scoped_administration.honor_global_acl*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	アクセス制御
目的	スコープ対象の管理アプリケーションで ACL アクセスルールを制御します。
推奨値	true
デフォルト値	true
構成タイプ	ブーリアン
セキュリティリスク	(低) プロパティ値が true で、レコードスコープにスコープ対象のアドミンアプリケーション ACL が定義されていない場合は、グローバル ACL が優先されます。false に設定すると、レコードスコープにスコープ対象の管理アプリケーション ACL が定義されていない場合、ACL チェックは無視されます。
セキュリティリスク評価	3.8
参照	アプリケーション管理アプリのアクセス制御ルール

プラグインのアクティブ化の詳細については、「プラグインを有効にする 」を参照してください。

サービス組織の作業指示管理クエリルールを有効にする (Security Center 1.5 の新機能、2.0 で更新)

sn_fsm.use_query_rules プロパティを使用して、フィールドサービス管理 (FSM) テーブルにルールとフィルターを適用します。

true に設定すると、sn_query_ruleテーブルのルール/フィルターを使用して、クエリビジネスルールと読み取り ACL を介して、ログインしたユーザーに対するフィールドサービス管理 (FSM) 関連テーブル (作業指示書および作業指示タスク) への読み取りアクセス権が決定されます。false に設定されている場合、レコードはクエリルールに基づいてフィルタリングされません。クエリービジネスルールは、セキュリティ検証を追加します。具体的には、このプロパティは、アサインされたテリトリーまたはテリトリーメンバーシップに基づいて、エージェント、認定者、およびディスパッチャーのレコードをフィルタリングします。レコードを読み取るときは、最小特権の原則に従うことがベ

ストプラクティスです。このプロパティが true に設定されていない場合、フィールドサービス管理 (FSM) テーブルからデータが漏洩するリスクが高まる可能性があります。

詳細情報

属性	説明
構成名	*sn_fsm.use_query_rules*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	false
デフォルト値	false
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：7.3 • CVSS スコア：中 • セキュリティリスクの詳細：このプロパティが true に設定されている場合、sn_query_rule のルールとフィルターを使用し、最小特権の原則を実装して、フィールドサービス管理 (FSM)テーブルへの読み取りアクセスが決定されます。false に設定されている場合、レコードはクエリビジネスルールに基づいてフィルタリングされず、フィールドサービス管理 (FSM) テーブルからのデータ漏洩のリスクが高まる可能性があります。
依存関係と前提条件	なし
機能への影響度	true に設定されている場合、sn_query_rule テーブルのルールとフィルターを使用して、フィールドサービス管理 (FSM) 関連テーブルへの読み取りアクセスが決定されます。たとえば、クエリビジネスルールと読み取り ACL を通じて、ログインユーザーに作業指示 (WO) と作業指示テーブル (WOT) を送信します。false に設定されている場合、レコードはクエリルールに基づいてフィルタリングされません。 このプロパティを有効にすると、データが保護され、すべてのデータ (wm_task および wm_order) がユーザーに表示されなくなります。
参照	フィールドサービス管理 (FSM)の参照

ACL を有効にしてライブプロファイルの詳細を制御する (Security Center 1.3 で更新)

glide.live_profile.details プロパティを使用して、ユーザーがライブプロファイルのすべての詳細フィールド (会社名や電話番号など) を表示できるようにするかどうかを指定します。

glide.live_profile.details プロパティの設定に応じて、以下が発生します。

- 値が [表示] に設定されている場合、ユーザープロファイルに対して作成された ACL に関係なく、ライブプロファイル情報へのアクセスが許可されます。
- 値が [ACL] に設定されている場合、ユーザープロファイルに対して作成された ACL に応じて、ライブプロファイル情報へのアクセスが制限されます。
- 値が [非表示] に設定されている場合、ユーザープロファイルに対して作成された ACL に関係なく、ライブプロファイル情報へのアクセスが制限されます。

詳細情報

属性	説明
プロパティ名	*glide.live_profile.details*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	アクセス制御
目的	目的は、許可されたユーザーのみがライブプロファイルの詳細 (会社名、電話番号など) にアクセスできるようにすることです。
データタイプ	選択リスト
推奨値	ACL
デフォルト値	ACL
セキュリティリスク評価	4.3
機能への影響	プロパティが有効になっていない場合、無許可のユーザーが他のすべてのユーザーのライブプロファイルの詳細にアクセスできます。
セキュリティリスク	(中) API 要求は常にテーブル ACL を優先する必要があります。無許可のユーザーがライブプロファイルの詳細にアクセスするのを防ぐために、制限を適用する必要があります。

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

簡易リストウィジェットでエンコードクエリの **ACL** を有効にする **(Security Center 2.0** の新機能)

glide.service_portal.enable_acls_for_encoded_query_in_list プロパティを安全な値に設定して、ユーザーが簡易リストウィジェットのクエリ条件でアクセス制御リスト (ACL) 評価をバイパスしないようにする方法について説明します。

glide.service_portal.enable_acls_for_encoded_query_in_list プロパティが安全な値である true に設定されていない場合、ユーザーは簡易リストウィジェットのクエリ条件でアクセス制御リスト (ACL) の評価をバイパスできる可能性があります。プロパティが false に設定されている場合は、以前の動作に戻り、enforce_aclチェックボックスの値に基づいてエンコードされたクエリの ACL チェックが強制されます。

クエリ内の ACL を評価して、ユーザーがクエリ対象のフィールドにアクセスできることを確認し、不正なデータ漏洩を防ぐことがベストプラクティスです。

glide プロパティ *glide.service_portal.enable_acls_for_encoded_query_in_list* が true に設定されていることを確認します。プロパティがsys_propertiesテーブルに存在しない場合、デフォルト値は true です。

詳細情報

属性	説明
構成名	*com.glide.script.fencing.cross_scope_access.shared_table_*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	文字列
推奨値	true
デフォルト値	true
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：7.3 • CVSS スコア：中 • セキュリティリスクの詳細:このプロパティを推奨値の true に設定していません。
依存関係と前提条件	なし
機能への影響	簡易リストウィジェットは、ユーザーのロールと基礎となる ACL によっては、データを表示しないことがあります。さらに、簡易リストクエリに、現在のユーザーがアクセスできないプロパティを持つフィルター条件が含まれている場合、ユーザーにセキュリティ警告が表示される可能性があります。

iframe 間のクロスオリジン通信で URL 許可リストを有効にする

glide.ui.concourse.onmessage_enforce_same_origin プロパティを使用して、iframe 間のクロスオリジン通信を有効にします。

Openframe は、*glide.ui.concourse.onmessage_enforce_same_origin_whitelist* プロパティで指定された信頼できるドメインからのメッセージのみ処理できます。詳細については、「iframe 間のクロスオリジン通信で URL 許可リストを有効にする」を参照してください。

詳細情報

属性	説明
プロパティ名	*glide.ui.concourse.onmessage_enforce_same_origin*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブール
カテゴリ	アクセス制御
目的	信頼できるドメインの包含リストを有効にし、OpenFrame の iframe 間で通信できるようにすること

属性	説明
推奨値	true
デフォルト値	true
セキュリティリスク評価	4.2
機能への影響	意図したドメインを包含リストに含めない場合、ServiceNow AI Platform インスタンス内に他のページを埋め込む機能が制限されることがあります。
セキュリティリスク	(高) Web ページに適切な作成元検証を実行しないイベントハンドラーが含まれている場合、任意の作成元からの Web ページまたはスクリプトは、その Web ページと通信できます。また、イベントハンドラーによって実行される機能を開始することもできます。
参照	OpenFrame の概要

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する 」を参照してください。

Anti-CSRF トークンを有効にする **(Security Center 1.3** の新機能、1.5 で更新、2.0 で削除)

glide.security.use_csrf_token プロパティを使用して、受信要求の識別および検証に保護トークンが使用されるようにします。このトークンは、これらの攻撃を防止するために使用されます。

Cross-Site Request Forgery (CSRF) は、認証されたユーザーに、現在認証されている Web アプリケーションに要求を送信するように強制する攻撃です。CSRF 攻撃は、認証されたユーザーに対する Web アプリケーションの信頼を悪用します。このプロパティを使用すると、セキュアトークンを使用して受信要求を識別して検証できます。このトークンはクロスサイトリクエストフォージェリ攻撃を防ぐために使用されます。*glide.security.use_csrf_token* が推奨値の true に設定されていない場合は、CSRF が可能です。

詳細情報

属性	説明
プロパティ名	*glide.security.use_csrf_token*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	アクセス制御
目的	CSRF 攻撃の可能性からアプリケーションを保護すること
セキュリティリス	8.1

ク評価	
推奨値	true
デフォルト値	true
機能への影響	この修正により、インスタンスユーザーがインスタンスへの書き込み要求を送信する前に、追加の検証手順が有効になります。すべての書き込み要求には、CSRF トークン (つまりユーザーセッションに関連付けられた検証/CSRF ID) が含まれています。ユーザーセッションが期限切れになると、セキュアトークンも期限切れになります。
セキュリティリスク	(高) クロスサイト要求偽造は、インスタンスデータの完全性を侵害する重大なセキュリティリスクです。攻撃者は、インスタンスユーザーの信頼を悪用して CSRF 攻撃を開始することができます。ソーシャルエンジニアリング攻撃を利用して、ユーザーがインスタンスで攻撃者の代わりに誤った要求を送信する可能性があります。

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

コンテキストセキュリティプラグインを有効にする (Security Center 1.3 で更新)

コンテキストセキュリティプラグイン (*com.glide.role_management*) のアクティブ化によりコンテキストセキュリティが有効になり、作成、読み取り、書き込み、および削除機能を使用してレコード/情報が保護されます。

インストールしてアクティブ化すると、(簡易セキュリティマネージャーによって作成された) dictionary ロールはテストされなくなります。代わりに、ServiceNow AI Platform はフィールドとテーブルの ACL ルールを検索します。また、シンプルなセキュリティマネージャーによって実装される従来のロールベースの辞書ルールではなく、ACL ルールを使用してデータを保護します。辞書フォームを設定して辞書エントリにロールを追加しても、権限は変更されません。

詳細情報

属性	説明
プラグイン ID	*com.glide.role_management*
構成タイプ	[システム定義] > [プラグイン]
カテゴリ	アクセス制御
目的	シンプルなセキュリティマネージャーとは異なり、コンテキストセキュリティマネージャーはシステムテーブル階層を認識しています。階層内の表示場所に基づいて、1 つのフィールドに複数の異なるセキュリティルールを設定できます。
推奨値	アクティブ
デフォルト値	これはプラグインであり、Glide プロパティではないため、デフォルト値はありません。
セキュリティリスク評価	8.1

属性	説明
機能への影響	この修正では、アクセスコントロールの機能レベルが適用されます。これにより、アプリケーションは ACL テーブルのみに基づいてアクセス制限を決定できます。
セキュリティリスク	(高) CRUD 操作を実行する前に、サーバー側から機能レベルのアクセス制御を適用し、インスタンスユーザーに適切なレベルのアクセスを確保する必要があります。
参照	コンテキスト依存セキュリティマネージャー

サービスポータルフォームでクロススコープ権限チェックを有効にする (Security Center 7.0 の新機能)

システムプロパティを使用して、サービスポータルフォームウィジェットでクロススコープ権限チェックを適用し、スコープ間でのフォームやテーブルデータの不正取得を防止します。

Yokohama 以降のリリースでは、クエリは、指定されたsys_id情報を読み取る前に、テーブルに対するクロススコープ権限チェックを強制します。

glide.service_portal.enforce_cross_scope_check_in_form プロパティが推奨値の **true** に設定されている場合、クロススコープ権限チェックがテーブルに適用されます。**false** に設定すると、クロススコープ特権チェックは適用されません。

*glide.service_portal.enforce_cross_scope_check_in_form*システムプロパティを **true** に設定するか、プロパティがシステムプロパティ [sys_properties] テーブルに存在しないことを確認します。sys_propertiesテーブルにレコードが存在しない場合、値はデフォルトで **true** に設定されます。

詳細情報

属性	説明
構成名	*glide.service_portal.enforce_cross_scope_check_in_form*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：3.1 • CVSS スコア：低 • セキュリティリスクの詳細:クロススコープ権限チェックが行われていない場合、スコープ間でフォームやテーブルデータが不正に取得される可能性があります。
依存関係と前提条件	なし

HR ライフサイクルイベントデータに ACL を強制する (Security Center 2.0 の新機能)

*glide.enforce_security_scope.sn_hr_le property*がセキュリティで保護された値に設定されていることを確認して、ヒューマンリソース (HR) ライフサイクルイベントアプリケーションのデータへの不正アクセスを防止する方法について説明します。

glide.enforce_security_scope.sn_hr_le プロパティは、「sn_hr_le」スコープのみが考慮されるように、複数の HR テーブルのアクセス制御リスト (ACL) を制限します。*glide.enforce_security_scope.sn_hr_le* が推奨値の true に設定されていない場合、ヒューマンリソース (HR):ライフサイクルイベントアプリケーションのデータは他のすべてのスコープの ACL に公開され、無許可のユーザーが機密データにアクセスする可能性があります。たとえば、HR データへのアクセス権を取得する IT アドミニストレーターなどです。

⚠ 警告: これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

詳細情報

属性	説明
構成名	*glide.enforce_security_scope.sn_hr_le*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	false
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：6.5 • CVSS スコア：中 • セキュリティリスクの詳細:このプロパティを値 true に設定しないと、ヒューマンリソース (HR):ライフサイクルイベントアプリケーションデータが他のすべてのスコープから ACL に公開される可能性があります。これにより、権限のないユーザーが機密データにアクセスする可能性があります。
依存関係と前提条件	なし

HR コアデータに ACL を強制する (Security Center 2.0 の新機能)

ヒューマンリソース (HR) スコープ対象のアプリ:Core (com.sn_hr_core) プラグインが他のすべてのスコープのアクセス制御リスト (ACL) に機密データを公開しないように、*glide.enforce_security_scope.sn_hr_core* プロパティを構成する方法について説明します。

glide.enforce_security_scope.sn_hr_core プロパティは、sys_attachment や sys_email などの複数のグローバルデータテーブルのアクセス制御リスト (ACL) を、sn_hr_coreスコープのみを考慮するように制限します。このプロパティが推奨値の true に設定されていない場合、ヒューマンリソース (HR) スコープ対象のアプリ:Core プラグインのデータは他のすべてのスコープの ACL に公開されます。たとえば、これにより、IT アドミニストレーターがヒューマンリソース (HR) データにアクセスできる可能性があります。

⚠ 警告: これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

詳細情報

属性	説明
構成名	*glide.enforce_security_scope.sn_hr_core*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：6.5 • CVSS スコア：中 • セキュリティリスクの詳細:このプロパティが安全な値である true に設定されていない場合、ヒューマンリソース (HR) スコープ対象のアプリ:Core プラグインのデータは他のすべてのスコープの ACL に公開されます。
依存関係と前提条件	なし
参照	ケースとナレッジ管理 のアクティブ化

HR 仮想エージェントデータに ACL を強制する (Security Center 2.0 の新機能)

*glide.enforce_security_scope.sn_hr_va*プロパティを安全な値に設定して、仮想エージェント対話スコープ対象のアプリケーションからのデータ漏洩を防ぐ方法について説明します。

*glide.enforce_security_scope.sn_hr_va*プロパティは、sn_hr_vaスコープのみを考慮するように、複数のヒューマンリソース (HR) テーブルのアクセス制御リスト (ACL) を制限します。このプロパティが推奨値の true に設定されていない場合、ヒューマンリソース (HR):仮想エージェント対話スコープ対象のアプリケーションのデータは他のすべてのスコープの ACL に公開されます。たとえば、これにより IT アドミニストレーターが HR データにアクセスできるようになります。

詳細情報

属性	説明
構成名	*glide.enforce_security_scope.sn_hr_va*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	false
カテゴリ	アクセス制御

自動翻訳

属性	説明
セキュリティリスク	• 重大度スコア：6.5 • CVSS スコア：中 • セキュリティリスクの詳細:このプロパティを安全な値 true に設定しないと、ヒューマンリソース (HR):仮想エージェント対話スコープ対象のアプリケーションのデータが他のすべてのスコープの ACL に公開される可能性があります。
依存関係と前提条件	なし

アプリケーションデータに対してのみアプリケーション固有の **ACL** を適用する

アプリケーションデータに対してのみアプリケーション固有のアクセス制御リスト (ACL) を適用することで、アプリケーションデータへの無許可のアクセスや望ましくないアクセスを回避します。

アプリケーション外のプライマリテーブルに存在するアプリケーションデータの動作を制御します。これらのプロパティの値が **true** の場合、これらのテーブルに存在するアプリケーションデータへのアクセスについて、アプリケーション固有の ACL のみが評価されます。すべてのアプリケーションがこの構成で動作するように設計されているわけではなく、この目的でシステムプロパティ [sys_properties] レコードを使用するように設計されているわけでもありません。

これらのシステムプロパティは、 **glide.enforce_security_scope.<scope>** 命名形式を使用します。たとえば、従業員センター Core (sn_hr_sp) スコープには **glide.enforce_security_scope.sn_hr_sp** プロパティを使用します。次のアプリケーションスコープには、このプロパティが含まれています。

- sn_doc
- sn_egd_act
- sn_egd_core
- sn_egd_goals
- sn_em
- sn_gsm
- sn_gsm_info_req
- sn_gsm_lic_prmt
- sn_gsm_lic_prmt_ex
- sn_gsm_soc_bnfts
- sn_hc_professional
- sn_hr_agent_ws
- sn_hr_ai_agents
- sn_hr_awa
- sn_hr_core
- sn_hr_ef
- sn_hr_er
- sn_hr_gen_ai
- sn_hr_hc
- sn_hr_le
- sn_hr_le_ent
- sn_hr_mii_base
- sn_hr_na_galileo
- sn_hr_pad
- sn_hr_pj
- sn_hr_sp
- sn_hr_va
- sn_hr_ws
- sn_imt_health_test
- sn_imt_tracing
- sn_imt_vaccine
- sn_ja
- sn_jny
- sn_lg_contracts
- sn_lg_matter
- sn_lg_ops
- sn_opp_market
- sn_professional
- sn_svc_appl_info
- sn_svc_appl_pgm_mg
- sn_talent_aia
- sn_uni_req
- sn_uni_task

システムプロパティ [sys_properties] テーブルの *glide.enforce_security_scope* プロパティとともにインストールされた各アプリケーション (*glide.enforce_security_scope.sn_hr_core* など) について、プロパティ値が **true** に設定されていることを確認します。

注: これらのプロパティは、特定のアプリケーションのスコープ指定されたアドミニストレーターのみが変更できます。指定されたアプリケーションとそれぞれのプロパティのsys_propertiesレコードが存在しない場合は、作成する必要があります。

このスクリプトを使用して、インスタンスで更新または作成する必要があるプロパティを見つけることができます。

```
var properties = [
    'glide.enforce_security_scope.sn_uni_task',
    'glide.enforce_security_scope.sn_uni_req',
    'glide.enforce_security_scope.sn_svc_appl_info',
    'glide.enforce_security_scope.sn_professional',
    'glide.enforce_security_scope.sn_opp_market',
    'glide.enforce_security_scope.sn_lg_ops',
    'glide.enforce_security_scope.sn_lg_matter',
    'glide.enforce_security_scope.sn_lg_contracts',
    'glide.enforce_security_scope.sn_jny',
    'glide.enforce_security_scope.sn_ja',
    'glide.enforce_security_scope.sn_imt_vaccine',
    'glide.enforce_security_scope.sn_imt_tracing',
    'glide.enforce_security_scope.sn_imt_health_test',
    'glide.enforce_security_scope.sn_hr_ws',
    'glide.enforce_security_scope.sn_hr_va',
    'glide.enforce_security_scope.sn_hr_sp',
    'glide.enforce_security_scope.sn_hr_pj',
    'glide.enforce_security_scope.sn_hr_pad',
    'glide.enforce_security_scope.sn_hr_mii_base',
    'glide.enforce_security_scope.sn_hr_le',
    'glide.enforce_security_scope.sn_hr_le_ent',
    'glide.enforce_security_scope.sn_hr_hc',
    'glide.enforce_security_scope.sn_hr_gen_ai',
    'glide.enforce_security_scope.sn_hr_er',
    'glide.enforce_security_scope.sn_hr_ef',
    'glide.enforce_security_scope.sn_hr_core',
    'glide.enforce_security_scope.sn_hr_awa',
    'glide.enforce_security_scope.sn_hr_agent_ws',
    'glide.enforce_security_scope.sn_hc_professional',
    'glide.enforce_security_scope.sn_gsm_soc_bnfts',
    'glide.enforce_security_scope.sn_gsm_lic_prmt_ex',
    'glide.enforce_security_scope.sn_gsm_lic_prmt',
    'glide.enforce_security_scope.sn_gsm_info_req',
    'glide.enforce_security_scope.sn_gsm',
    'glide.enforce_security_scope.sn_em',
    'glide.enforce_security_scope.sn_egd_goals',
    'glide.enforce_security_scope.sn_egd_core',
    'glide.enforce_security_scope.sn_egd_act',
    'glide.enforce_security_scope.sn_doc',
    'glide.enforce_security_scope.sn_talent_aia',
    'glide.enforce_security_scope.sn_hr_na_galileo',
    'glide.enforce_security_scope.sn_svc_appl_pgm_mg',
    'glide.enforce_security_scope.sn_hr_ai_agents',
    'glide.enforce_security_scope.sn_hr_mii_base'
];

var pm = new GlidePluginManager();

for (var i = 0; i < properties.length; i++) {
```

```
    var property = properties[i];
    var application = property.split('.')[2];
    var propertyValue = gs.getProperty(property, 'false');

    if (pm.isActive(application) && propertyValue.toLowerCase() != 'true') {
       gs.print(property);
    }
 }
```

詳細情報

属性	説明
構成名	*glide.enforce_security_scope.<scope>*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：4.1 • CVSS スコア：中 • セキュリティリスクの詳細:これらのプロパティの値が **true** でない場合でも、プライマリテーブルの ACL はアクセスについて評価されるため、アプリケーションデータへの不正なアクセスや望ましくないアクセスが許可される可能性があります。
依存関係と前提条件	なし

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

アプリケーションスコープの制限を強制する (Security Center 1.3 の新機能、1.5 で削除)

glide.record.legacy_cross_scope_access_policy_in_script プロパティを使用して、スコープ対象のアプリの権限を制御します。

glide.record.legacy_cross_scope_access_policy_in_script プロパティが true に設定されている場合、スコープ対象アプリは、グローバルアプリのみが使用できるはずの API を呼び出すことができます。このプロパティは、これらのスコープ対象アプリの開発者を作成および更新する際に意図されているアクセス制御をバイパスします。

詳細情報

属性	説明
構成名	*glide.record.legacy_cross_scope_access_policy_in_script*
構成タイプ	システムプロパティ (/sys_properties_list.do)

属性	説明
データタイプ	ブーリアン
推奨値	false
デフォルト値	true (sys_properties テーブルにプロパティが存在しない場合)
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：3.5 • CVSS スコア：低 • セキュリティリスクの詳細：このプロパティが推奨値に設定されていない場合、スコープ対象アプリとそれらのスコープ対象アプリの委任開発者は、インシデントなどのグローバルテーブルのレコードを作成および更新できます。
依存関係と前提条件	なし

カタログ変数検索の読み取りロールを強制する (Security Center 7.0 の新機能)

システムプロパティを使用して、読み込みロールが空のカタログ変数のみが検索用にインデックス付けされるようにします。

プロパティ *glide.ais.ingestion.ignore_catalog_variables_read_roles* が推奨値の **false** に設定されている場合、読み込みロールが空のカタログ変数のみが検索用にインデックス付けされます。このプロパティが **true** に設定されている場合、変数に指定された読み込みロールに関係なく、すべての変数が検索用にインデックス化されます。

glide.ais.ingestion.ignore_catalog_variables_read_roles システムプロパティがシステムプロパティ [sys_properties] テーブルに存在しないか、存在していて **false** に設定されていることを確認します。

詳細情報

属性	説明
構成名	*glide.ais.ingestion.ignore_catalog_variables_read_roles*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	false
デフォルト値	false
カテゴリ	アクセス制御
機能への影響	このプロパティは、読み取りアクセスに特定のロールを必要とするカタログ変数からの検索可能コンテンツのインデックス作成に影響します。

属性	説明
セキュリティリスク	• 重大度スコア:2.6 • CVSS スコア：低 • セキュリティリスクの詳細:ユーザーは、読み取りロールを持たない変数を検索して情報漏えいを引き起こす可能性があります。
依存関係	なし
参照	サービスカタログ変数

ダッシュボードの共有にセキュリティルールを強制する (セキュリティセンター **1.3** の新機能)

glide.cms.dashboards.sharing_with_secure_search プロパティを使用して、ユーザーがダッシュボードを共有できるかどうかを制御します。

glide.cms.dashboards.sharing_with_secure_search プロパティが **true** に設定されていない場合、ユーザーはアクセス権を持っていないダッシュボードグループとロールを共有できます。このプロパティを有効にした場合、ダッシュボード共有プロセス中に sys_user、sys_user_role、および sys_user_group テーブルを検索するときにアクセス制御リスト (ACL) が適用されます。ダッシュボードを過度に共有すると、ユーザー、グループ、またはロールが、表示する権限がないデータにアクセスし、機密情報が侵害される可能性があります。したがって、*glide.cms.dashboards.sharing_with_secure_search* を true に設定して、適切な権限を持つユーザーとのみダッシュボードが共有されるようにすることをお勧めします。

詳細情報

属性	説明
構成名	*glide.cms.dashboards.sharing_with_secure_search*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	false
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：3.5 • CVSS スコア：低 • セキュリティリスクの詳細：このプロパティを推奨値の true に設定しないと、sys_user、sys_user_role、および sys_user_group テーブルの検索時にアクセス制御リストは適用されません。これにより、権限のないユーザーとダッシュボードが共有され、機密情報が公開される可能性があります。
依存関係と前提条件	なし
参照	アクセス制御

属性	説明
機能への影響度	このプロパティは、ダッシュボードを共有すると表示されるユーザーのリスト、ユーザーグループ、およびロールにセキュリティルールを適用します。

公共機関デジタルサービスのスコープセキュリティを強制する (Security Center 1.3 の新機能)

glide.enforce_security_scope.sn_gsm プロパティを使用して、公共機関デジタルサービスアプリケーションからのアプリケーションデータへのアクセス方法を制御します。

この ServiceNow 公共機関デジタルサービスアプリケーションを使用すると、市民、企業、および機関にデジタルサービスを提供する公共部門アプリケーションを開発できます。

glide.enforce_security_scope.sn_gsm が false に設定されている場合、公共機関デジタルサービスアプリのグローバルテーブル内に存在するアプリケーションデータへのアクセスは、それらのグローバルテーブルのアクセス制御リスト (ACL) に基づいて実行できます。このプロパティが true に設定されている場合、グローバルテーブルに存在するデータへのアクセスは、公共機関デジタルサービスアプリケーションに直接付属する ACL に基づいてのみ評価されます。このプロパティを false に設定すると、過剰な許容度の ACL から情報が漏洩する可能性が生じます。

このセキュリティリスクを修復するには、*glide.enforce_security_scope.sn_gsm* を true に設定します。

詳細情報

属性	説明
構成名	*glide.enforce_security_scope.sn_gsm*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：4.2 • CVSS スコア：中 • セキュリティリスクの詳細：このプロパティを推奨値に設定しないと、過剰な許容度の ACL により情報が開示される可能性があります。
依存関係と前提条件	なし
参照	公共機関デジタルサービスの構成

Information Request Playbook に対してスコープ付き ACL アクセスを強制する (Security Center 1.3 の新機能、1.5 で更新)

glide.enforce_security_scope.sn_gsm_info_req プロパティを使用して、Information Request Playbook 機能のプレイブックデータへのアクセスを制御します。

Information Request Playbook アプリケーションを使用すると、公共部門のエンドユーザーは、公開記録要求を送信および追跡できます。さらに、これらの要求を処理および解決するための事前定義されたプロセスを政府機関に提供します。*glide.enforce_security_scope.sn_gsm_info_req* が true に設定されていない場合、Information Request Playbook アプリケーションの プレイブックデータに予期しないアクセスが許可される可能性があります。アクセスを許可するときに sn_gsm_info_req スコープからの ACL のみを考慮するには、このプロパティを true に設定します。

詳細情報

属性	説明
構成名	*glide.enforce_security_scope.sn_gsm_info_req*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：7.3 • CVSS スコア：中 • セキュリティリスクの詳細：このプロパティが false に設定されている場合、スコープマスターテーブルのプレイブックデータへのアクセスを許可するときに、すべてのスコープの ACL が考慮されます。これにより、Information Request Playbook データが公開されます。
依存関係と前提条件	なし
参照	• Information Request Playbook の使用 • 情報要求サービスチャネルの設定

厳格な特権昇格を強制する **(Security Center 1.3** の新機能)

glide.security.strict_elevate_privilege プロパティを使用して、ユーザーにロールの機能を付与するために、特権としてマークされたロールを手動で昇格させる必要があるかどうかを制御します。

このプロパティを true に設定すると、特権ユーザーのロールの昇格時にセキュリティ検証のレイヤーが追加されます。

詳細情報

属性	説明
構成名	*glide.security.strict_elevate_privilege*
構成タイプ	システムプロパティ (/sys_properties_list.do)

属性	説明
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：6.7 • CVSS スコア：中 • セキュリティリスクの詳細：*glide.security.strict_elevate_privilege* が false に設定されている場合、特権としてマークされたロールは、アドミンユーザーの新しいセッション時に自動的に昇格され、手動で昇格させる必要はありません (security_admin を除く)。
依存関係と前提条件	なし
機能への影響度	このプロパティでは、必要に応じて権限を昇格するには、必ず「admin」ロールユーザーである必要があります。

セキュリティスコープライセンスと許可プレイブックを強制する **(Security Center 1.5** の新機能、2.0 で更新)

このプロパティを使用して、スコープへのアクセスを決定する際に、ライセンスと許可プラグイン内のアクセス制御リスト (ACL) のみを使用するか、すべてのスコープの ACL を考慮するかを決定します。

glide.enforce_security_scope.sn_gsm_lic_prmt プロパティが推奨値である true に設定されている場合、ライセンスと許可プラグイン内の ACL のみがスコープへのアクセスを決定するために使用されます。この設定が false に構成されている場合、すべてのスコープの ACL にアクセスが許可されるため、スコープマスターテーブル内のライセンスと許可プレイブックデータが公開されます。データの公開を減らすには、*glide.enforce_security_scope.sn_gsm_lic_prmt* を推奨値である true に設定します。

詳細情報

属性	説明
構成名	*glide.enforce_security_scope.sn_gsm_lic_prmt*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：2.7 • CVSS スコア：低

属性	説明
	• セキュリティリスクの詳細：この設定を推奨値である true に構成すると、アクセスを許可するために *sn_gsm_lic_prmt* スコープの ACL のみが考慮されるため、スコープマスターテーブル内のライセンスと許可プレイブックデータが保護されます。これを false に設定すると、アクセスを許可するためにすべてのコープの ACL が考慮されるため、スコープマスターテーブル内のライセンスと許可プレイブックデータが公開されます。たとえば、この設定が false の場合、IT アドミニストレーターはライセンスと許可プレイブックデータにアクセスできます。
依存関係と前提条件	なし
参照	• ライセンスと許可プレイブック の使用 • アプリケーションスコープ

HR ケース管理のエージェントワークスペースにセキュリティスコープを適用する (セキュリティセンター 1.5 の新機能、2.0 で更新)

HR ケース管理のエージェントワークスペースプラグインを構成して、スコープマスター テーブル内のデータに適切な権限を持つユーザーのみがアクセスできるようにし、最小特権の原則を適用します。

glide.enforce_security_scope.sn_hr_agent_ws プラグインが推奨値 true に設定されている場合、HR ケース管理のエージェントワークスペースプラグイン内のアクセス制御リスト (ACL) のみがリソースへのアクセスを決定するために使用されます。この設定を false に設定すると、すべてのスコープの ACL にアクセスが許可されるため、スコープマスターテーブル内の HR ケース管理のエージェントワークスペースデータが公開されます。たとえば、この設定が false に設定されている場合、IT アドミニストレーターは、HR ケース管理のエージェントワークスペースにアクセスできます。これを防ぐには、*glide.enforce_security_scope.sn_hr_agent_ws* を推奨値である true に設定します。これにより、ユーザーは権限のあるリソースのみにアクセスできるため、最小特権の原則が確実に存在します。

詳細情報

属性	説明
構成名	*glide.enforce_security_scope.sn_hr_agent_ws*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：2.7 • CVSS スコア：低

自動翻訳

属性	説明
	• セキュリティリスクの詳細：この設定を false に設定すると、すべてのスコープの ACL にアクセスが許可されるため、スコープマスターテーブル内の HR ケース管理のエージェントワークスペースデータが公開されます。
依存関係と前提条件	HR ケース管理のエージェントワークスペース
機能への影響度	この設定を true に構成すると、スコープ付き ACL が存在しないテーブルに対してグローバル ACL が強制的に実行されます。
参照	• https://owasp.org/www-project-proactive-controls/#div-numbering • エージェントワークスペース へのコンポーネントの追加

サービスアプリケーション情報にセキュリティスコープを適用する **(Security Center 2.0** の新機能)

glide.enforce_security_scope.sn_svc_appl プロパティを使用して、マススコープテーブルのデータがセキュリティで保護されていることを確認します。

glide.enforce_security_scope.sn_svc_appl_info プロパティが true に設定されている場合、スコープ内のリソースへのアクセスは、サービスアプリケーション情報プラグイン (sn_svc_appl_info) からのアクセス制御リスト (ACL) によってのみ決定されます。これにより、アクセス許可がsn_svc_appl_infoスコープ内で定義されているものに制限されるため、マススコープテーブル内のデータのセキュリティが確保されます。

安全でない値 false に設定すると、sys_attachment などのマススコープテーブルのデータへのアクセスを許可するときに、すべてのスコープの ACL が考慮されます。これにより、サービスアプリケーション情報データに対する権限を持たないユーザーによって機密情報に不正にアクセスされる可能性があります。

詳細情報

属性	説明
構成名	*glide.enforce_security_scope.sn_svc_appl_info*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	false
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：7.3 • CVSS スコア：中 • セキュリティリスクの詳細:このプロパティが安全でない値 false に設定されている場合、サービスアプリケー

属性	説明
	ション情報データに対する権限を持たないユーザーによって機密データに不正にアクセスされる可能性があります。
依存関係と前提条件	*glide.enforce_security_scope.sn_svc_appl_info*プロパティを有効にするには、サービス申請者情報プラグイン (*com.sn_svc_appl_info*) を有効にする必要があります。

GlideRecordSandbox でのフィールドレベル ACL の適用

インスタンスの GlideRecordSandbox でフィールドレベルの ACL を管理します。

glide.sandbox.fields.check_acl プロパティを使用して、GlideRecordSandbox でフィールドレベルの ACL を適用します。このプロパティが適用される例としては、*sysparm_query* のようにユーザーがスクリプトを提供できる場合が挙げられます。このプロパティが推奨値の **true** に設定されていない場合、ACL 制限がバイパスされ、権限のないユーザーによって、*sys_user.user_password* などの機密データが侵害される可能性があります。

⚠ 警告: このプロパティの値は、DB オーバーライドなしです。変更またはオーバーライドすることはできません。

詳細情報

属性	説明
構成名	*glide.sandbox.fields.check_acl*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：7.5 • CVSS スコア：高 • セキュリティリスクの詳細：このプロパティを **false** に設定すると、ACL 制限がバイパスされ、機密データが公開される可能性があります。
依存関係と前提条件	なし

認証情報エイリアスの使用を強制する (セキュリティセンター **1.3** の新機能、1.5 で更新**)**

MID サーバーのプロパティを設定して資格情報を不正使用から保護する方法について説明します。

管理、計測、検出 (MID) サーバーは、ローカルネットワーク内で Windows サービスまたは UNIX デーモンとして動作する Java アプリケーションです。MID サーバーのプロパティは [ecc_agent_property] テーブルにリストされます。インスタンスでアクセスするには、 **MID** サー

バー > プロパティ. 認証情報エイリアスを使用すると、アドミニストレーターはディスカバリースケジュールで特定の認証情報を使用できます。認証情報エイリアスを使用すると、ディスカバリーテーブルで使用を許可される認証情報をより細かくコントロールできます。このセキュリティの脆弱性を修復するには、*alias_filtering_behavior* を strict に設定して、 昇格された権限によって認証情報が不必要に公開されないようにします。詳細は、「MID サーバープロパティ 」を参照してください。

詳細情報

属性	説明
構成名	*alias_filtering_behavior*
構成タイプ	MID サーバープロパティ (/ecc_agent_property_list.do)
データタイプ	文字列
推奨値	strict
デフォルト値	loose
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：2 • CVSS スコア：低 • セキュリティリスクの詳細：このハードニング設定が「厳格 (strict)」に設定されていない場合、ディスカバリーテーブルのエイリアスに関係なくすべての認証情報が使用されるため、不正アクセスの可能性が高まります。
依存関係と前提条件	なし
参照	• MID サーバー • ディスカバリーの認証情報エイリアス

GroupBy ACL の適用

groupby 列の ACL チェックを実施するようにインスタンスを構成します。

glide.security.groupby_acl_check システムプロパティを使用して、groupby 列の ACL チェックを実行するようにインスタンスを設定します。このプロパティが推奨値の **true** に設定されている場合、groupby 列の ACL がデフォルトで優先されます。テーブルの *groupby_acl_check* 属性は、 *glide.security.groupby_acl_check* プロパティよりも優先されます。プロパティが **false** に設定されている場合は、groupby 列の ACL チェックを行う必要があるテーブルの *groupby_acl_check* 属性が **true** に設定されていることを確認してください。

プロパティ *glide.security.groupby_acl_check* が **true** に設定されていることを確認します。

詳細情報

属性	説明
構成名	*glide.security.groupby_acl_check*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：3.7 • CVSS スコア：低 • セキュリティリスクの詳細：このプロパティを **false** に設定すると、groupby 列の ACL チェックが無効になり、情報漏洩につながる可能性があります。
依存関係と前提条件	なし

アーカイブテーブル **ACL** のチェックを確認する **(Security Center 1.3** の新機能、1.5 で更新)

glide.security.enable_archive_table_acls プロパティは 、元のテーブル (アーカイブテーブルの作成元のテーブル) のアクセス制御リスト (ACL) が false と評価されるかどうかを制御します。

元のテーブルの ACL はその値に関係なく評価されるため、*glide.security.enable_archive_table_acls* プロパティを false に設定してはいけません。アーカイブテーブルの ACL を追加しないことで、追加の ACL を回避できます。

詳細情報

属性	説明
構成名	*glide.security.enable_archive_table_acls*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：3 • CVSS スコア：低 • セキュリティリスクの詳細：このプロパティが false に設定されている場合、アーカイブされたテーブルに追加

自動翻訳

属性	説明
	された ACL は無視されます。これは直感的ではないアクションであるため、認証のバイパスにつながる可能性があります。
依存関係と前提条件	なし
機能への影響	このプロパティが true に設定されている場合、アーカイブテーブルのアクティブな読み込み ACL はすべて受け入れられます。アクティブな読み取り ACL が存在しないか、プロパティが false に設定されている場合、元のテーブル (データのアーカイブ元のテーブル) の ACL がアーカイブテーブルに適用されます。 **注:** アーカイブテーブルでは読み取り ACL のみがサポートされます。アーカイブテーブルに対するその他の操作は、アクセスハンドラーによって内部的に制御されます。

ダッシュボードの作成/削除にはアクセスチェックが必要であることを確認する **(Security Center 1.3** の新機能、2.0 で更新)

glide.processors.check_access_before_process システムプロパティを使用すると、ユーザーがログインしているときにダッシュボードを作成または削除するためのアクセス制御リスト (ACL) を適用できます。

glide.processors.check_access_before_process システムプロパティを **[true]** に設定します。プロパティがシステムプロパティ [sys_properties] テーブルに表示されない場合、フォールバック値は **true** です。

詳細情報

属性	説明
構成名	*glide.processors.check_access_before_process*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
フォールバック値	true
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：6.3 • CVSS スコア：中 • セキュリティリスクの詳細：このプロパティを false に設定して無効にすると、ダッシュボードで ACL バイパスが有効になります。これにより、低い権限を持つすべての認証済みユーザーがダッシュボードを削除および追加できるようになります。

属性	説明
依存関係と前提条件	なし
機能への影響度	このプロパティは、ユーザーに必要なアクセス権がない場合に、新しい sys_dashboards を作成したり、既存のダッシュボードを削除したりする機能を制御します。値が false に設定されている場合、不適切なロールを持つユーザーでも、sys_dashboard エントリを追加および削除できます (ただし、GlideRecord レイヤーは既存の ACL を再チェックする必要があります)。true の値を指定すると、必要なアクセス権を持たないユーザーの追加および削除の操作が制限されます。

機密データテーブルとフィールドをデータ生成から除外する (Security Center 7.0 の新機能)

システムプロパティを使用して、既存のデータに基づいて偽のデータセットを生成するために使用されるデータ生成からテーブルとフィールドを除外します。これらの除外リストに追加されたテーブルとフィールドは、データ生成機能には使用できません。

glide.data.generation.excluded.tables システムプロパティに含まれるテーブルは、メタデータテーブルに加えてデータ生成から除外されます。

*glide.data.generation.excluded.fields.<TABLE-NAME>*のカンマ区切りリストに含まれるフィールドは、該当する場合、これらのフィールドに加えてデータ生成から除外されます。

- 番号
- ロール
- sys_class_name
- sys_created_by
- sys_id
- sys_mod_count
- sys_tags
- sys_updated_by

プロパティ *glide.data.generation.excluded.tables*に含まれるテーブルのリストを確認します。データ生成から除外する必要があるテーブルをテーブルのカンマ区切りリストに追加します。さらに、メタデータテーブルはデータ生成で無視されます。

glide.data.generation.excluded.fields.<TABLE-NAME> という形式のプロパティを調べて、各テーブルのフィールドのリストを確認します。指定したテーブルの機密フィールドをカンマ区切りの値リストとして追加します。

詳細情報

属性	説明
構成名	• *glide.data.generation.excluded.tables* • *glide.data.generation.excluded.fields.**
構成タイプ	システムプロパティ (/sys_properties_list.do)

属性	説明
データタイプ	• テーブル名のカンマ区切りリスト • 指定されたテーブルのフィールドのカンマ区切りリスト
推奨値	• データ生成から除外する必要があるテーブル名のカンマ区切りリスト • 適用可能なテーブルごとのデータ生成から除外する必要があるフィールド名のカンマ区切りリスト
デフォルト値	"",""
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア:2.6 • CVSS スコア：低 • セキュリティリスクの詳細:データ生成機能で使用されるテーブルのデータには、機能の意図したとおりに実際の値または偽の値を入力できます。表示または複製すべきではないテーブル内の機密値が、他のインスタンスユーザーに公開される可能性があります。
依存関係と前提条件	データ生成プラグインは使用中です

CSRF 検証をバイパスする警告をユーザーが受け入れるのを防ぐ (Security Center 1.3 および 1.5 で更新)

glide.security.csrf.strict.validation.mode プロパティを使用して、CSRF トークンの厳格検証を有効にします。CSRF トークンが一致しない場合、要求は再送信されません。

このプロパティは、悪意のある可能性のある要求をインスタンスに送信する可能性のある警告をユーザーが受け入れることを防ぎます。この警告は、被害者の他のアクティブセッションの 1 つに属する Anti-CSRF トークンの一致が間違っているために POST 要求が失敗した場合に表示されます。*glide.security.csrf.strict.validation.mode* が推奨値の true に設定されていない場合、攻撃者は、被害者に属する別のアクティブセッションから漏洩した CSRF 対策トークンを利用して CSRF 攻撃を仕組むことができます。インスタンスへの POST 要求には、ユーザーの現在のセッションと一致する「sysparm_ck」または「X-UserToken」内に CSRF 対策トークンが含まれています。

CSRF 対策トークンがユーザーの他のアクティブ セッションの 1 つに関連付けられている場合、このプロパティが false に設定されている場合、POST 要求は security_interceptor.do への 302 リダイレクトを返し、ユーザーが使用できる [続行] ボタンを使用します。このボタンをクリックすると、有効な CSRF 対策トークンが含まれる場合を除き、要求がインスタンスに再送信されます。このプロパティを true に設定すると、security_interceptor.do ページへの 302 リダイレクトに [続行] ボタンが表示されず、ユーザーは要求を再送信できません。CSRF 攻撃が成功すると、攻撃者は被害者が実行できるすべての操作を効果的に実行できます。

詳細情報

属性	説明
プロパティ名	*glide.security.csrf.strict.validation.mode*

属性	説明
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	アクセス制御
目的	CSRF トークンの厳格検証を適用し、再利用を防止すること
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
セキュリティリスク評価	(中) クロスサイト要求偽造は、インスタンスデータの完全性を侵害する重大なセキュリティリスクです。攻撃者は、インスタンスユーザーの信頼を悪用して、どのインスタンスユーザーにでも CSRF 攻撃を開始することができます。ソーシャルエンジニアリング攻撃を利用して、ユーザーが攻撃者の代わりにインスタンスに対して誤った要求を送信する可能性があります。
セキュリティリスク評価	3.7
機能への影響	この修正により、インスタンスユーザーがインスタンスへの書き込み要求を送信する前に、追加の検証手順が有効になります。現在の CSRF トークンが以前に使用されたかどうかをチェックします。[はい] の場合、以降の書き込み要求は送信されません。
セキュリティリスク	(中) クロスサイト要求偽造は、インスタンスデータの完全性を侵害する重大なセキュリティリスクです。攻撃者は、インスタンスユーザーの信頼を悪用して、どのインスタンスユーザーにでも CSRF 攻撃を開始することができます。ソーシャルエンジニアリング攻撃を利用して、ユーザーが攻撃者の代わりにインスタンスに対して誤った要求を送信する可能性があります。

ラップされたキーをアップロードするには 顧客指定のキーの構成とアップロード に戻ります。

委託開発者の読み取りアクセスを制限する (Security Center 1.3 で更新)

com.glide.dd_allow_global_access_tables に wf_activity、wf_activity_definition、wf_workflow、wf_workflow_version、sp_portal、sp_widget、sp_page の推奨値が含まれていない場合、これらのテーブルは委託開発者によって読み取られる可能性があります。これにより、機密情報への委任開発者の読み取りアクセスが提供される可能性があります。

詳細情報

属性	説明
プロパティ名	*com.glide.dd_allow_global_access_tables*
構成タイプ	システムプロパティ (/sys_properties_list.do)

データタイプ	文字列
推奨値	wf_activity、wf_activity_definition、wf_workflow、wf_workflow_version、sp_portal、sp_widget、sp_p
デフォルト値	wf_activity、wf_activity_definition、wf_workflow、wf_workflow_version、sp_portal、sp_widget、sp_p
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：2.7 • CVSS スコア：低 • セキュリティリスクの詳細：*com.glide.dd_allow_global_access_tables* が wf_activity、wf_activity_definition、wf_workflow、wf_workflow_version、sp_portal、sp_widget、sp_ に設定されていることを確認してください。

AJAXGlideRecord ACL チェックを必須とする (Security Center 1.3 で更新)

glide.script.secure.ajaxgliderecord プロパティを使用して、クライアントスクリプト内で GlideAjax API を使用してサーバー側のレコード (テーブルなど) にアクセスする場合にアクセス制御リスト (ACL) ルール検証を実行します。

クライアントスクリプトから、AJAXGlideRecord (GlideAjax -クライアント サーバー側の Glide レコードなどの構文を使用して、API を使用します。多くの展開で、これは強力で便利なツールです。

アクセス制御リスト (ACL) を GlideAjax API 呼び出しに適用することを選択すると、現在接続しているユーザーがアクセスできるデータのみをクエリできます。たとえば、cmn_location テーブルを読み取る権限のない ESS ユーザーがログインすると、そのテーブルに対する GlideAjax API 呼び出しは失敗します。

ServiceNow AI Platform が GlideAjax ACL 呼び出しのチェックなしで実行されている場合、API は、現在ログインしているユーザーが他の方法ではアクセスできない情報を返す可能性があります。

データのクエリを実行するときに GlideRecordSecure を使用して、最高レベルのセキュリティを確保します。GlideRecord は構成による ACL の適用に依存しますが、GlideRecordSecure はより厳格なセキュリティ制御を適用します。GlideRecordSecure は、機密データを処理するための、より安全ですぐに使えるソリューションを提供します。

⚠ **警告:** これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

詳細情報

属性	説明
プロパティ名	*glide.script.secure.ajaxgliderecord*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	アクセス制御

自動翻訳

属性	説明
目的	クライアント側 API を介してレコードにアクセスする場合でも、セキュリティ ACL が確実にチェックおよび検証されるようすること
推奨値	true
デフォルト値	true
セキュリティリスク評価	8.1
機能への影響	この修正では、AJAXGlideRecord API 呼び出しを使用して要求が行われた場合に、サーバー側レコードとの ACL 関係が適用されます。ACL 構成が適切に設定されていない場合、影響を受ける可能性があります。影響度とその識別方法の詳細については、「クライアント側 GlideRecord (AJAXGlideRecord) トランザクションの監査とレビューに関する記事 [KB0550828] (HI ナレッジベース で入手可能)」を参照してください。
セキュリティリスク	(高) クライアントスクリプトを使用すると、GlideAjax API を介してサーバーから任意のデータをクエリーできます。サーバー側のリソースには適切な承認なしでアクセスできるため、ACL 検証を使用すると、アプリケーションで設定された承認に基づいて要求を検証するのに役立ちます。
ワークアラウンド	GlideAjax (AJAXGlideRecord) API で使用されるスクリプトインクルード、プロセッサー、およびその他のエンティティに対して適切な ACL が作成され、適切な承認の下で実行されるようにします。 canRead ()、canWrite ()、canCreate ()、canDelete () などのメソッドを実装して、GlideRecord を使用してテーブルレコードにアクセスする前に、ユーザーの承認を実行します。 もう 1 つの方法は、GlideRecordSecure を使用することです。このクラスは GlideRecord サーバーから継承され、GlideRecord と同じ関数を実行し、ACL も適用します。
参照	ACL を AJAXGlideRecord (クライアント側の Glide レコード) に追加 このプロパティは、*glide.script.allow.ajaxevaluate* などのクライアントからのスクリプトの実行を保護および制限するプロパティと同じファミリーに属します。詳細については、「AJAXEvaluate を有効にする」を参照してください。

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

システムフィールドへの書き込みアクセスをアドミンユーザーに制限 (Security Center 7.0 の新機能)

*glide.rest.table_api.admin_only_sys_fields*システムプロパティを使用して、システムによって生成されたフィールドへの書き込みアクセスを制御します。

*glide.rest.table_api.admin_only_sys_fields*プロパティは、次のフィールドへの書き込みアクセスを制御します。

- sys_id
- sys_created_by
- sys_created_on
- sys_updated_by
- sys_updated_on

このプロパティを **true** に設定すると、アドミンのみがシステム生成値であるこれらのフィールドに書き込むことができます。プロパティが **false** に設定されている場合、またはプロパティがシステムプロパティ [sys_properties] テーブルに存在しない場合、テーブルの作成または書き込みアクセス権を持つユーザーは、 テーブル API を使用してこれらのシステム値に書き込むことができます。

アドミン以外のユーザーがレコードのシステムフィールドを更新できないようにするには、プロパティ *glide.rest.table_api.admin_only_sys_fields*を **true** に設定します。

詳細情報

属性	説明
構成名	*glide.rest.table_api.admin_only_sys_fields*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	false
フォールバック値	false
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：2.7 • CVSS スコア：低 • admin ロールを持たないユーザーが、sys_created_by や sys_updated_on などのフィールドを更新できました。このアクセス権は、作成または更新されたレコードメタデータの完全性に影響を与え、レコードが別のユーザーによって作成されたか、別の時間に更新されたかのように表示される可能性があります。
機能への影響	テーブル API を使用してレコードを作成または更新し、これらのフィールドを変更する、アドミンロールを持たない統合またはユーザーが影響を受けます。テーブル API に対するこれらの要求では、クエリパラメーター

属性	説明
	*sysparm_suppress_auto_sys_field*が **false** に設定され、要求本文でこれらのフィールドが設定されます。
依存関係と前提条件	なし

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

エージェントベースの **Office 365** グループ メンバーシップの変更に承認を要求する **(Security Center 7.0** の新機能)

システムプロパティを使用して、Microsoft 365 グループメンバーシップ AI エージェントを通じて Office 365 グループメンバーを追加または削除するための承認フローを有効にします。

sn_itsm_aia.office_365_group_member_approval.required システムプロパティを使用して、AI エージェントを介して Office 365 グループメンバーを追加または削除するための承認フローをオンかオフかを制御します。承認ワークフローが有効になっている場合は、承認レコードを *sn_itsm_aia.office_365_group_member_approval.group_id* システムプロパティで指定されたグループのメンバーによって承認されるように設定する必要があります。*sn_itsm_aia.office_365_group_member_approval.group_id* プロパティが構成されていない場合は、*Microsoft 365 group member approvers* グループが使用されます。

詳細情報

属性	説明
構成名	*sn_itsm_aia.office_365_group_member_approval.required*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
フォールバック値	true
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：4.9 • CVSS スコア：中 • Microsoft 365 グループ メンバーシップ AI エージェントにアクセスできるユーザーは、指定された承認者グループからの明示的な承認が必要ない場合、Azure AD グループに対して Office 365 グループ メンバーを追加および削除できます。適切な承認なしにメンバーを追加/削除することで特権が昇格するリスクを軽減するために、これらの変更が承認されていることを確認します。
機能への影響	*sn_itsm_aia.office_365_group_member_approval.required*が **true** に設定されている場合、*sn_itsm_aia.office_365_group_member_approval.<group_i*で指定されたグループのメンバーは、Office

属性	説明
	365 グループメンバーの追加または削除を要求するインシデントを承認する必要があります。*sn_itsm_aia.office_365_group_member_approval.required* **false** に設定されている場合、承認は不要で、AI エージェントは Office 365 グループに対してメンバーを追加または削除するプロセスを自律的に処理できます。
依存関係と前提条件	なし

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

代理操作ユーザーがアプリケーションデータを表示できないようにします

システムプロパティを使用して、代理操作しているユーザーがアプリケーションデータを表示できないようにします。

アカウントの代理操作を行うときに、アドミンレベルがそのユーザーに属するアプリケーション固有のデータにアクセスできないようにします。この権限は、アプリケーションに固有のシステムプロパティを作成することで、アプリケーションレベルで設定できます。

これらのシステムプロパティは、 *<scope>.impersonateCheck* 命名形式 (*sn_hr_core.impersonateCheck* など) を使用します。値が **true** のシステムプロパティを作成して、ユーザーがアカウントの代理操作を行うときに、他のユーザーに属するアプリケーション固有のデータにアクセスできないようにします。

注: すべてのアプリケーションがこの構成で動作するように設計されているわけではなく、この目的のためにシステムプロパティ [sys_properties] レコードがあるわけでもありません。このプロパティを使用するように、次のスコープが設定されています。

- sn_opp_market
- sn_jny
- sn_imt_vaccine
- sn_imt_health_test
- sn_hr_core
- sn_egd_goals
- sn_egd_core
- sn_egd_act
- sn_em
- sn_talent_aia

システムプロパティ [sys_properties] テーブルに *<scope>.impersonateCheck* プロパティがあるアプリケーションごとに、プロパティ値が **true** に設定されていることを確認します。

注: これらのプロパティは、特定のアプリケーションのスコープ指定されたアドミニストレーターのみが変更できます。

このスクリプトを使用して、インスタンスで更新または作成する必要があるプロパティを検索します。

```
var properties = [
   'sn_opp_market.impersonateCheck',
   'sn_jny.impersonateCheck',
   'sn_imt_vaccine.impersonateCheck',
   'sn_imt_health_test.impersonateCheck',
   'sn_hr_core.impersonateCheck',
   'sn_egd_goals.impersonateCheck',
   'sn_egd_core.impersonateCheck',
   'sn_egd_act.impersonateCheck',
'sn_em.impersonateCheck',
'sn_talent_aia.impersonateCheck'
];

var pm = new GlidePluginManager();

for (var i = 0; i < properties.length; i++) {
   var property = properties[i];
   var application = property.split('.')[0];
   var propertyValue = gs.getProperty(property, 'false');

   if (pm.isActive(application) && propertyValue.toLowerCase() != 'true') {
      gs.print(property);
   }
}
```

詳細情報

属性	説明
構成名	*<scope>.impersonateCheck*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	false
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア:3.8 • CVSS スコア：低 • これらのプロパティの値が **false** の場合、アドミンレベルのユーザーは別のユーザーの代理操作を行い、代理操作されたユーザーのアクセス権でアプリケーションデータにアクセスできます。これは望ましくない場合や、特定のアプリケーションコンテキストで不正なデータアクセスを許す可能性があります。
依存関係と前提条件	なし

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

OAuth ステータスパラメーター検証の適用

クロスサイトリクエストフォージェリ (CSRF) 攻撃からインスタンスを保護するように *glide.oauth.state.parameter.required* プロパティを設定します。

glide.oauth.state.parameter.required プロパティを使用すると、認証コードフローの OAuth 要求でステータスパラメーターを要求できます。ステータスパラメーターは文字列値であり、特殊文字を含めたり、空白にしたりすることはできません。このプロパティを **true** に設定すると、攻撃者は認証中にクロスサイトリクエストフォージェリ (CSRF) 攻撃を実行できなくなります。これにより、代理操作されたユーザーからの攻撃からインスタンスが保護されます。

詳細情報

属性	説明
構成名	*glide.oauth.state.parameter.required*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：4.2 • CVSS スコア：中 • セキュリティリスクの詳細：CSRF 攻撃を確実に防止するには、このプロパティを **true** に設定します。
依存関係と前提条件	なし

厳格なユーザー画像アップロードを強制する

glide.security.strict.user_image_upload プロパティを使用して、ユーザーレコードでプロファイル画像のアップロード/更新を実行する際のアクセス制御を有効にします。

この設定では、無許可のユーザーが別のユーザーのプロファイルに画像をアップロードする可能性が生じます。

- このプロパティを **true** に設定すると、写真をアップロードするときにテーブル ACL が適用され、許可されたユーザーのみが画像をアップロードできるようになります。
- **false** に設定すると、[写真] フィールドへの画像アップロードに ACL は適用されません。

詳細情報

属性	説明
プロパティ名	*glide.security.strict.user_image_upload*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	アクセス制御

属性	説明
目的	ユーザー画像のアップロードを許可されたユーザーのみに制限すること
推奨値	true
セキュリティリスク評価	3.7
機能への影響	承認されたユーザーは引き続きユーザープロファイルに画像をアップロードできるため、機能への影響はありません。
セキュリティリスク	(例) このプロパティを **false** に設定すると、認証されたユーザーは別のユーザーのアカウントに承認なしで画像をアップロードできます。

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

外部ユーザー登録用の電子メールドメインを制限する (セキュリティセンター 1.3、1.5、および **2.0** で更新)

sn_ext_usr_reg.allowed_email_domains プロパティを使用して、許容可能な外部メールドメインを一覧表示します。

sn_ext_usr_reg.allowed_email_domains システムプロパティは、ServiceNowインスタンスへの自己登録が許可されるメールアドレスを定義します。形式は、example@domain2.com などのメールが受け入れられる domain1.com,domain2.com などの受け入れ可能なメールドメインのカンマ区切りリストにする必要があります。受け入れ可能なドメインのリストで **sn_ext_usr_reg.allowed_email_domains** が設定されていない場合、メールアドレスを持つユーザーはインスタンスにアカウントを登録できます。定義されていない場合、悪意のある攻撃者が、インスタンスへの認証されたアクセスを取得するために、望ましくないドメインのメールアドレスを使用して登録を実行する可能性があります。

詳細情報

属性	説明
プロパティ名	*sn_ext_usr_reg.allowed_email_domains*
構成タイプ	システムのプロパティ (/sys_properties_list.do)、Communities プロパティ
カテゴリ	アクセス制御
目的	ユーザーのメールの登録を許可するためにメールドメインを一覧表示します。
推奨値	空ではない値として設定します
構成タイプ	文字列
セキュリティリスク	(高) 悪意のあるアクターが、不要なドメインからのメールアドレスを使用して登録を実行する可能性があります。**sn_ext_usr_reg.allowed_email_domains** が空の値に設定されていないことを確認します。
セキュリティリスク評価	7.5
参照	Communities

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

高セキュリティプラグインを有効にする (Security Center 1.3 で更新)

High Security プラグインをアクティブ化すると、インスタンスのセキュリティレベルを制御する数百もの異なる構成が作成または更新されます。これらの構成は、厳格なアクセス制御、入力検証、および出力エンコーディングを有効にすることで、上位の OWASP 攻撃の多くを軽減します。

これらの設定には次のような内容が含まれます。

- アクセス制御
- ビジネスルール
- システムプロパティ
- UI ポリシーアクション
- スクリプトアクション
- スクリプトインクルード

例

次のプロパティの例を参照してください。

プロパティ	トピック
glide.ui.escape_all_script	Jelly スクリプトをエスケープ (セキュリティセンター 1.3 および 1.5 で更新)
glide.security.strict.actions	実行前の UI アクションの条件のチェック
glide.security.csrf_previous.allow	Anti-CSRF トークンを有効にする (Security Center 1.3 の新機能、1.5 で更新、2.0 で削除)
glide.security.csrf.strict.validation.mode	CSRF 検証をバイパスする警告をユーザーが受け入れるのを防ぐ (Security Center 1.3 および 1.5 で更新)

詳細情報

属性	説明
プラグイン名	com.glide.high_security
構成タイプ	[システム定義] > [プラグイン - 開発]
カテゴリ	アクセス制御
目的	このプラグインをアクティブ化する必要があります。これにより、インスタンスのセキュリティレベルが向上し、CSRF、XSS、セッション Cookie のセキュリティ保護、ファイルのアップロードなどの owasp 上位 10 件の攻撃が軽減されることで、攻撃対象領域が縮小します。
推奨値	有効
セキュリティリスク評価	9.8
機能への影響	このプラグインは、UI と機能に影響を与える可能性があるシステムセキュリティ構成をいくつか有効にします。

属性	説明
セキュリティリスク	(高) 多くのセキュリティ設定が意図せずオープンのままになるため、一部の重大な脆弱性が無防備になる可能性があります。
参照	高セキュリティ設定の有効化 高セキュリティ設定

プラグインのアクティブ化の詳細については、「プラグインを有効にする」を参照してください。

アドミン優先 ACL

glide.security.admin.override.accessterm プロパティは、上書きが有効になっている場合でも、アドミニストレーターが ACL 評価を上書きできないように制御します。

詳細情報

属性	説明
プロパティ名	*glide.security.admin.override.accessterm*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	アクセス制御
目的	アドミニストレーターが ACL 評価を上書きできないように制御します。
データタイプ	ブーリアン
推奨値	true
デフォルト値	True
セキュリティリスク	(低) ACL は累積的に評価されます。指定されたフィールドに多数の ACL があり、そのうちの 1 つでアドミン優先オプションが false (未選択) の場合、すべての ACL の有効なアドミン優先は falseと見なされます。
セキュリティリスク評価	3.8
参照	アクセス制御リストのルール

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

非アクティブなユーザーのログインを防ぐ (Security Center 1.5 の新機能)

このプロパティを構成して、非アクティブなユーザーがインスタンスで認証できるかどうかを制御します。

glide.authenticate.only.allow.active.user.login プロパティの値を true (推奨) に設定すると、非アクティブとマークされた sys_user テーブルのユーザーはインスタンスにログインできなくなり、ロックアウトされます。

この設定が false に構成されている場合、ユーザーは以前にアクセスしていたインスタンスとデータに引き続きアクセスできる可能性があります。

詳細情報

属性	説明
構成名	*glide.authenticate.only.allow.active.user.login*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：7.5 • CVSS スコア：高 • セキュリティリスクの詳細：このプロパティを推奨値である true に設定しないと、解雇された従業員などの非アクティブなユーザーが引き続きインスタンスやデータにアクセスできる可能性があります。
依存関係と前提条件	なし

仮想エージェント埋め込み **Web** クライアントへの非認証アクセスを防止します

認証されていないユーザーが埋め込み Web クライアントにアクセスできないように *sn_va_web_client_app_embed* テーブルを構成する方法について説明します。

仮想エージェントの埋め込み Web クライアントである UI ページ *sn_va_web_client_app_embed* には、sys_publicテーブルで true とマークされたアクセス制御リスト (ACL) がデフォルトで含まれています。パブリックアクセシビリティが必要なユースケースがあることが確認されていますが、これはデフォルトでパブリックアクセス可能に設定するための標準ではありません。

非認証ユーザーに埋め込み Web クライアントが必要ない場合は、公開ページ [sys_public] テーブルで *sn_va_web_client_app_embed* レコード (sys_id 04b1905473222300e985658b4cf6a7ef) を開き、[アクティブ] フィールドの選択を解除してページを非アクティブ化します。

詳細情報

属性	説明
構成名	**sn_va_web_client_app_embed**
構成タイプ	UI ページ (sys_ui_page_list.do)
データタイプ	table
推奨値	**sn_va_web_client_app_embed**公開ページ [sys_public] (sys_id 04b1905473222300e985658b4cf6a7ef) が存在しないか、アクティブではありません。
デフォルト値	利用不可 (これはテーブル値です)
カテゴリ	アクセス制御

属性	説明
セキュリティリスク	• 重大度スコア：7.5 • CVSS スコア (CVSS score)：高 • セキュリティリスクの詳細:非認証ユーザーに埋め込み Web クライアントが必要ない場合は、*sn_va_web_client_app_embed* UI ページを非アクティブ化することをお勧めします。
依存関係と前提条件	なし

JSONP 要求を信頼できる URL に制限する (Security Center 1.3 で更新)

AngularJS $http サービスの信頼できる URL を指定して、JSONP 要求を許可または拒否します。

AngularJS $httpサービスの信頼できる URL のみが JSONP 要求を許可/拒否できるようにすることで、インスタンスのセキュリティを強化します。これらのプロパティが構成されて有効になっていない場合、任意の URL に対して JSONP 要求が許可されます。

angular.jsonp.inclusion_list.urls システムプロパティの値を使用して、信頼され、この目的が許可されている URL のリストを定義します。許可された JSONP を **angular.jsonp.inclusion_list.urls** にリストされている URL のみに制限するには、**angular.jsonp.inclusion_list.enabled** システムプロパティの値を **true** に設定します。

詳細情報

属性	説明
構成名	*angular.jsonp.inclusion_list.enabled*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：中 • CVSS スコア：5.4 • セキュリティリスクの詳細：このプロパティを **false** に設定すると、任意の URL への JSONP 要求が可能になります。
依存関係と前提条件	なし

CSRF 検証をバイパスする警告をユーザーが受け入れるのを防ぐ

CSRF 検証をバイパスする警告をユーザーが受け入れるのを防ぐことで、クロスサイトリクエストフォージェリ (CSRF) のリスクを軽減します。

CSRF トークンの厳密な検証を有効にして、クロスサイトリクエストフォージェリ (CSRF) トークンが再利用されて CSRF 攻撃が許可されるのを防ぎます。

glide.security.csrf.strict.validation.mode システムプロパティ値を **true** に設定して、CSRF トークンの厳格な検証を有効にします。このプロパティがシステムプロパティ [sys_properties] テーブルに存在しない場合、Xanadu 以降、デフォルト値は **true** です。

詳細情報

属性	説明
技術的な構成名	glide.security.csrf.strict.validation.mode
プラグインの適用性	なし
セキュリティリスク	クロスサイトリクエストフォージェリは、インスタンスデータの完全性を侵害する重大なセキュリティリスクです。攻撃者は、インスタンスユーザーの信頼を悪用して、どのインスタンスユーザーにでも CSRF 攻撃を開始することができます。ソーシャルエンジニアリング攻撃を利用して、ユーザーが攻撃者の代わりにインスタンスに対して誤った要求を送信する可能性があります。
共通脆弱性スコアリングシステム (CVSS) スコア	3.7
共通脆弱性スコアリングシステム (CVSS) 評価	低
機能への影響	この修正により、ユーザーがインスタンスへの書き込み要求を送信する前に、追加の検証手順が有効になります。現在の CSRF トークンが以前に使用されたかどうかをチェックします。含まれている場合は、以降の書き込み要求の送信を防止します。
依存関係と前提条件	なし
データタイプ	ブール
ベースシステム値	true
フォールバック値	true
推奨値	true

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

生のデータベースクエリの実行を無効にする (Security Center 1.3 で更新、2.0 で削除)

ユーザーがデータベースに対して生の SQL クエリを実行できるかどうかを制御します。

glide.db.allow_unsafe_dbi_execute_sql プロパティを使用すると、ユーザーは生の SQL クエリをデータベースで実行できます。これにより、GlideRecord の制限外のテーブルとデータにアクセスできます。このプロパティが推奨値の *false* に設定されていない場合、悪意のある SQL ステートメントの実行につながる可能性がある Glide Scriptable から *dbi.executeStatement()* を呼び出すことができるようになります。

⚠ 警告: このプロパティは安全であり、DB オーバーライドはありません。

詳細情報

属性	説明
構成名	*glide.db.allow_unsafe_dbi_execute_sql*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	false
デフォルト値	false
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：7.2 • CVSS スコア：高 • セキュリティリスクの詳細：このプロパティを **false** に設定しないと、Glide Scriptable からの *dbi.executeStatement()* 呼び出しが可能になります。
依存関係と前提条件	なし
参照	アクセス制御リストのルール

記事に対するユーザーコメントを非表示にする **(Security Center 1.3** の新機能)

glide.knowman.show_user_feedback プロパティを使用して、フィードバックコメントを表示するかどうかを制御します。

glide.knowman.show_user_feedback が never に設定されていない場合、フィードバックコメントは、Glide プロパティ *glide.knowman.show_user_feedback.roles* で定義されたロールを持つユーザーに対して、ナレッジベース (KB) 記事に表示されます。フィードバックコメントには機密情報が含まれている可能性があるため、フィードバックを表示したくない場合があります。このプロパティが never に設定されていない場合、機密情報がフィードバックで開示された場合に機密性に影響を与える可能性があります。

詳細情報

属性	説明
構成名	*glide.knowman.show_user_feedback*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	選択リスト
推奨値	まったくない
デフォルト値	onLoad
カテゴリ	アクセス制御

属性	説明
セキュリティリスク	• 重大度スコア：3.5 • CVSS スコア：低 • セキュリティリスクの詳細：このプロパティを never に設定しないと、機密情報がフィードバックコメントで公開される可能性があります。
依存関係と前提条件	なし
機能への影響度	構成で言及されている選択肢に基づいて、KB 記事にユーザーコメントを表示します。

クライアント呼び出し可能スクリプトインクルードにはデフォルトで認証を必須とする (Security Center 1.3 で更新)

デフォルトでは、明示的に可視化を設定していないクライアント呼び出し可能スクリプトインクルードは、公開状態になります。必要に応じて *glide.script.ccsi.ispublic* プロパティを追加すると、公開ページからアクセスされるクライアント呼び出し可能スクリプトインクルード全体に対してプライバシーコントロールができるようになります。

このプロパティを追加する場合は、値を **false** に設定する必要があります。これは、クライアント呼び出し可能スクリプトインクルードがすべて非公開であり、公開ページで可視化を変更するという指定になります。

注: 値が **true** のプロパティを追加したり、値を **false** から **true** に変更したりすることはできません。これを実行しようとすると、エラーメッセージが表示されます。

必要に応じて、isPublic() 関数を追加することにより、クライアント呼び出し可能スクリプトインクルードのプライバシー設定を個別に変更できます。

- isPublic() 設定は *glide.script.ccsi.ispublic* プロパティよりも優先されます。
- たとえば、個別のスクリプトで isPublic() を **true** に設定すると、このスクリプトが公開状態になります。これは、他のクライアント呼び出し可能スクリプトインクルードをすべて非公開にする *glide.script.ccsi.ispublic* プロパティを上書きします

警告: これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

詳細情報

属性	説明
プロパティ名	*glide.script.ccsi.ispublic*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	アクセス制御
目的	クライアント呼び出し可能スクリプトインクルードを非公開にすると、公開ページにアクセスするゲストはクライアント呼び出し可能スクリプトインクルードにアクセスできないということになります。ログインしていないユーザーは、非公開スクリプトを実行できません。
推奨値	false

自動翻訳

属性	説明
デフォルト値	false
セキュリティリスク評価	7.5
機能への影響	クライアント呼び出し可能スクリプトインクルードが公開として指定されている場合 (つまり、このプロパティがない場合)、非認証ユーザーはクライアントスクリプトを実行できます。ログインしていないユーザーによるスクリプトの実行を制限するプロパティを追加します。
セキュリティリスク	(高) このプロパティを追加しない場合、クライアント側スクリプトインクルードが ACL を回避し、その結果意図せずに機能が公開される可能性があります。クライアントスクリプトに機密情報がある場合は、有害なセキュリティリスクが発生するおそれがあります。
ワークアラウンド	*glide.script.ccsi.ispublic* プロパティを **false** に設定すると、すべてのクライアント呼び出し可能スクリプトインクルードが非公開になります。 isPublic() 関数を追加することにより、クライアント呼び出し可能スクリプトインクルードのプライバシー設定を個別に変更できます。isPublic() 関数は *glide.script.ccsi.ispublic* プロパティよりも優先されます。スクリプトインクルードに次の構文を追加します。 isPublic:function(){return[true/false];},

本番インスタンスの動作を強制する (セキュリティセンター **1.3** および **1.5** で更新)

インスタンスを本番インスタンスのように処理するか非本番インスタンスのように処理するかを構成します。

glide.installation.production プロパティが推奨値の **true** に設定されていない場合、インスタンスは本番インスタンスとして扱われないため、zboot やその他の危険性のあるスクリプトの実行が許可されます。本番インスタンスを非本番インスタンスとして評価できるようにすると、情報漏えいやサービス拒否 (DoS) 攻撃につながる可能性があります。

詳細情報

属性	説明
構成名	*glide.installation.production*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：6.3 • CVSS スコア：中

属性	説明
	• セキュリティリスクの詳細：このプロパティ値を **false** に設定すると、インスタンスが非本番インスタンスとして扱われ、zboot やその他の危険性のあるスクリプトを実行できるようになります。
依存関係と前提条件	なし

バックグラウンドスクリプトへのアクセスを制限する (Security Center 1.3 および 2.0 で更新)

glide.script_processor.admin プロパティを構成して、Script Background モジュールへのアクセスに必要なロールを設定します。

このプロパティは、[スクリプトバックグラウンド] モジュールにアクセスするために必要なロールを保持します。*glide.script_processor.admin* が推奨値とデフォルト値の admin に設定されていない場合、低い特権ロールを持つユーザーはインスタンスでバックグラウンドスクリプトを実行できます。これにより、ACL システムが完全にバイパスされ、テーブルへのフルアクセスが可能になります。

プロパティ *glide.script_processor.admin* がアドミンに設定されていることを確認します。これはインスタンスのデフォルト値です。

⚠ **警告:** これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

詳細情報

属性	説明
構成名	*glide.script_processor.admin*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	文字列
推奨値	admin
デフォルト値	admin
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：8.8 • CVSS スコア：高 • セキュリティリスクの詳細：このプロパティを推奨値の **admin** に設定しないと、すべてのユーザーがインスタンスでバックグラウンドスクリプトを実行できます。
依存関係と前提条件	なし

ターゲットテーブルが空のメールへのアクセスを制限する

glide.email.email_with_no_target_visible_to_all プロパティをアクティブ化して、ユーザーからメールへのアクセスを制限します (メール送信元のユーザーであるか、admin ロールを持つ場合を除く)。

許可されていないユーザーは、ターゲットレコードのない [sys_email_list] テーブル内のメールにアクセスできます。このプロパティは、メールエントリに ACL を適用するのではなく、メール送信者と admin ロールを持つユーザーのみにアクセスを制限します。

注: インスタンスによって送受信されたメールは、[sys_email_list] テーブルに表示されます。ただし、[エラー] および [無視] ステータスがマークされた受信メールにかぎり、空のターゲットテーブルを設定する必要があります。

詳細情報

属性	説明
プロパティ名	*glide.email.email_with_no_target_visible_to_all*
構成タイプ	システムプロパティ (/sys_properties_list.do)
構成	アクセス制御
目的	ユーザーがアクセスを許可しない場合に、メールクライアントでメールが表示されないようにすること
推奨値	false
デフォルト値	true
セキュリティリスク評価	6.5
機能への影響	ユーザーは、アドミンまたはメールの送信元でないかぎり、ターゲットテーブルが空のメールを表示できなくなります。
セキュリティリスク	(中) プロパティが有効でない場合、許可されていないユーザーは [target_table] フィールドが空であるメールにアクセスできます。
参照	詳細なメールプロパティ https://support.servicenow.com/kb_view.do?sysparm_article=KB0690043

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

特定の IP 範囲プラグインへのアクセスを制限する (Security Center 1.3 で更新)

com.snc.ipauthenticator プラグインを使用して、特定の IP 範囲へのアクセスを制限します。インスタンスに対するパブリック アクセスが意図されている場合を除き、アドミニストレーターは割り当てられた IP ネットブロックにアクセスを制限する必要があります。

必須条件

このプラグインを true に設定すると、特定の IP 範囲へのアクセスが制限されます。インスタンスに対するパブリック アクセスが意図されている場合を除き、アドミニストレーターは割り当てられた IP ネットブロックにアクセスを制限する必要があります。IP アドレスの除外リスト (拒否) または包含リスト (許可) は、IP アドレスアクセス制御 (ip_access_list.do) を使用して作成できます。

このプロパティを設定する前に、IP 範囲ベースの認証 (com.snc.ipauthenticator)*com.snc.ipauthenticator* プラグインを有効にする必要がありま

す。詳細については、「IP 範囲ベースの認証」および「設定手順」の項 (下記) を参照してください。

プラグイン *com.snc.ipauthenticator* が有効になっており、テーブルip_accessに少なくとも 1 つのアクティブな IP アクセスポリシーがあることを確認します。

詳細情報

属性	説明
プラグイン名	• com.snc.ipauthenticator • ip_access
構成タイプ	[System Security] > [IP アドレスアクセス制御]
カテゴリ	アクセス制御
目的	インスタンスにアクセスできるまたはアクセスできない IP アドレスの範囲を、信頼できるドメインと信頼できないドメインのリストに追加すること
推奨値	アクティブ
デフォルト値	なし。これはプラグインであり、Glide プロパティではありません。したがって、デフォルト値はありません。
セキュリティリスク評価	5.3
機能への影響	顧客が拒否した IP 範囲がこの修正アイテムに使用されます。顧客がターゲットリストを定義するため、影響はありません。
セキュリティリスク	(低) IP アクセス制御機能を使用して、インターネット上のターゲットインスタンスへの不必要な開示を制限する必要があります。
参照	IP 範囲ベースの認証

設定手順

1. com.snc.ipauthenticator プラグインがアクティブであることを確認します。
2. 移動先 システムセキュリティ > **IP** アドレスアクセス制御.
3. [新規] をクリックして、IP アドレスの除外リスト (拒否) または包含リスト (許可) を作成します。
4. [送信] をクリックします。

ナレッジベースへのアクセスを制限する **(Security Center 1.3** の新機能)

glide.knowman.block_access_with_no_user_criteria プロパティは、ナレッジベース記事へのユーザーの読み取り/書き込みアクセスを制御するために使用されます。

詳細情報

属性	説明
構成名	*glide.knowman.block_access_with_no_user_criteria*

属性	説明
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	false
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：9.1 • CVSS スコア：重大 • セキュリティリスクの詳細：このプロパティが推奨値の true に設定されていない場合、すべてのユーザーがナレッジベースで読み取りおよび寄稿可能になります。
依存関係と前提条件	なし
機能への影響	[読込可能] および [寄稿可能] のいずれも指定されていない場合に、ナレッジベースへのアクセスを拒否します。

CMDB モデルの権限を制限する (セキュリティセンター 1.3 および 1.5 で更新)

csm_cmdb_model.customer_visible_flag システムプロパティを使用し、CMDB モデルへの追加のアクセス制御として、製品モデルテーブルのデータへの顧客のアクセスを制限します。

csm_cmdb_model.customer_visible_flag プロパティを **true** に設定して、以下にリストされているテーブルの [顧客に表示] フィールドを有効にします。

- 製品モデルテーブル [cmdb_model]
- ソフトウェアモデルテーブル [cmdb_software_product_model]
- アプリケーションモデルテーブル [cmdb_application_product_model]
- 消耗品モデルテーブル [cmdb_consumable_product_model]
- 施設モデルテーブル [cmdb_facility_product_model]
- ハードウェアモデルテーブル [cmdb_hardware_product_model]

このプロパティを **true** に設定すると、すべての cmdb_model 値がデフォルトで非表示になります。

cmdb_model テーブルの customer_visible 列/属性を考慮せずに、sn_esm_user がアクセスできるベース cmdb_model ACL に依存するには、このプロパティを **false** に設定します。

詳細情報

属性	説明
プロパティ名	*csm_cmdb_model.customer_visible_flag*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	アクセス制御

属性	説明
目的	**true** に設定すると、[顧客に表示] フィールドの設定を使用して、カスタマーサービスポータルの製品モデルデータへのアクセスが決定されます。
推奨値	true
デフォルト値	false
構成タイプ	ブーリアン
セキュリティリスク	(中) sn_esm_user ロールと初期設定の ACL を持つすべてのユーザーが、CMDB モデルに対する権限を持つ可能性があります。 **注:** このロールは、外部ユーザーに付与される傾向があります。外部ユーザーに望まない CMDB モデルへの権限が付与される可能性があります。
参照	カスタマーサービスポータルで製品モデルデータへのアクセスを制限します

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

添付ファイルへの非認証アクセスを制限する

glide.image_provider.security_enabled プロパティを使用して画像のセキュリティ設定を制御します。**true** に設定すると、認証および承認されたユーザーにのみ画像が表示されます。**false** に設定すると、添付ファイルへの URL を持つすべてのユーザーに画像が表示されます。

機密情報の漏洩を防ぐために、インスタンス上の画像を保護します。インスタンス上の画像には、末尾 が .iix の URL を使用してアクセスできます。

これらの URL を介して画像にアクセスできないようにするには、**glide.image_provider.security_enabled** システムプロパティを **true** に設定します。

注:

元のテーブルが次のいずれかである場合、このプロパティは添付ファイルテーブルの画像では優先されません。

- 文房具 [sysevent_email_style]
- ようこそページセクション [sys_home]
- システムのプロパティ [sys_properties]

一部の添付ファイルに機密情報が含まれている可能性があるため、非認証ユーザーには制限を適用する必要があります。

詳細情報

属性	説明
プロパティ名	*glide.image_provider.security_enabled*
構成タイプ	システムプロパティ (/sys_properties_list.do)

属性	説明
カテゴリ	アクセス制御
目的	*.iix* 形式を使用してレンダリングされた添付ファイルへの非認証アクセスを防止します。
推奨値	true
デフォルト値	false
機能への影響	機能に大きな影響はありません。以前に .iix に直接アクセスしていたユーザーは認証を受ける必要があるため、ユーザーエクスペリエンスに多少影響する可能性があります。
セキュリティリスク	• 重大度スコア:6.5 • セキュリティリスクの詳細:一部の添付ファイルに機密情報が含まれている可能性があるため、非認証ユーザーには制限を適用する必要があります。
参照	添付ファイルの管理 利用可能なシステムプロパティ

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

カスタムジャーナルエントリへのアクセスを制限する (セキュリティセンター **1.3** で更新、2.0 で削除)

glide.live_feed.custom_journal.acl_check_enabled プロパティを使用して、カスタムジャーナルフィールドの ACL を優先します。

glide.live_feed.custom_journal.acl_check_enabled が推奨値の true に設定されていない場合、すべてのユーザーはライフフィード機能内のすべてのジャーナルエントリを表示できます。プロパティを true に設定すると、推奨機能であるカスタムジャーナルフィールドの ACL が優先されます。

詳細情報

属性	説明
プロパティ名	*glide.live_feed.custom_journal.acl_check_enabled*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	アクセス制御
目的	ACL に基づいてどのユーザーにどのジャーナルエントリーを表示するかを制御します。
推奨値	true
デフォルト値	true
構成タイプ	ブーリアン。
セキュリティリスク	(中) true に設定すると、ACL に合格するカスタムジャーナルエントリーのみがライブフィードに表示されます。そ

属性	説明
	れ以外の場合、すべてのユーザーがすべてのジャーナルエントリーを表示できます。

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

フローコンテキストの読み取りアクセスを制限する (Security Center 1.5 の新機能)

com.snc.process_flow.reporting.require_flow_access プロパティを使用して、ユーザーがフローチェックを読み取るときに追加のアクセスチェックを必要とするかどうかを設定します。

com.snc.process_flow.reporting.require_flow_access プロパティが推奨値である true に設定されている場合、フローコンテキストを読み取ろうとするユーザーに対して追加のアクセスチェックが行われます。このプロパティが false に設定されている場合、ある程度の情報の開示が発生する可能性があります。

詳細情報

属性	説明
構成名	*com.snc.process_flow.reporting.require_flow_access*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：2.7 • CVSS スコア：低 • セキュリティリスクの詳細：このプロパティを false に設定すると、既存の動作が保持されます。このプロパティを true に設定すると、読み取りアクセスのセキュリティレイヤーが追加されます。
依存関係と前提条件	なし
機能への影響度	このプロパティを有効にすると、フローコンテキストレコードの読み取りに対するセキュリティが強化されます。このインスタンスは、フローコンテキストを読み取ろうとするユーザーが親フローへの読み取りアクセスも持つことを強制します。

代理操作をアドミンに制限する (Security Center 2.0 の新機能)

glide.sys.permissive.impersonate プロパティを使用すると、admin 以外のロールが他のユーザーの代理操作を行うのを防ぐことができます。

glide.sys.permissive.impersonate プロパティが false に設定されている場合、admin ロールを持つユーザーのみが他のユーザーの代理操作を行うことができます。このプロパティを true に設定すると、ユーザーは、代理操作 API を公開するアプリケーションコンポーネントを利用して、

より高い権限を持つユーザーの代理操作を行うことができます。アドミン以外のユーザーが代理操作機能にアクセスできるためにこれらのアプリケーションコンポーネントが正しく構成されていない場合、不正アクセスが発生する可能性があります。

アドミン以外のユーザーに他のユーザーの代理操作機能を付与する必要がある場合は、プロパティをデフォルト値以外の値に設定することをお勧めします。

⚠ **警告:** これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

詳細情報

属性	説明
構成名	*glide.sys.permissive.impersonate*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	false
デフォルト値	false
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：6.7 • CVSS スコア：中 • セキュリティリスクの詳細:このプロパティを推奨値の false に設定しないと、アドミン以外のユーザーが API を公開するアプリケーションコンポーネントを利用し、より高い権限を持つユーザーの代理操作を行う可能性があります。
依存関係と前提条件	なし
機能への影響	アドミン以外のユーザーは、他のスクリプトや UI ページをカスタマイズして、代理操作機能にアクセスできます。ただし、正しいユーザーのみにこれらの機能へのアクセスが許可されるようにすることが重要です。 ℹ **注:** *glide.sys.permissive.impersonate* が true に設定されている場合でも、*impersonate* ロールを持つアドミン以外のユーザーは引き続き代理操作を行うことができます。

Security Jump Start プラグイン **(ACL** ルール) を有効化する **(Security Center 1.3** で更新)

Security Jump Start (ACL Rules) (com.snc.system_security*com.snc.system_security*) プラグインをアクティブ化して、ServiceNow AI Platform 内の主要なシステムテーブルの一部でアクセス制御を検証する重要な ACL を複数作成します。

これらのルールは、多くのシステムテーブルを保護するためのジャンプスタートを提供し、組織がより簡単にインスタンスを本番環境に移行できるようにします。Security Jump Start (ACL ルール) プラグインは、すべての新しいインスタンスに自動的にインストールされます。

詳細情報

属性	説明
プラグイン ID	*com.snc.system_security*
構成タイプ	[システム定義] > [プラグイン]
カテゴリ	アクセス制御
目的	Security Jump Start (ACL Rules) プラグインをアクティブ化して、適切なセキュリティコンプライアンスを実現します。 これは、最初にシステムテーブルを保護する基本的な ACL をいくつか提供します。インスタンスのデフォルトのプロビジョニングに付属する各システムテーブルに対して手動で作成するのではありません。これらの ACL は、新しく作成されたインスタンスを迅速に本番環境に移行する必要がある場合に役立ちます。
推奨値	アクティブ
デフォルト値	なし。これはプラグインであり、Glide プロパティではないため、デフォルト値はありません。このプラグインは、zBoot (リセット) によってデフォルトでインストールされます。
セキュリティリスク評価	8.1
機能への影響	インスタンスの既存の ACL を監査せずにこのプラグインをインストールすると、機能に大きな影響があります。修正を行う前に、顧客への働きかけと定義が必要です。
セキュリティリスク	(高) アクセス制御を適用して、インスタンスへの意図しないアクセスをロックする必要があります。ACL ジャンプスタートルールは、多くのシステムテーブルを保護するための開始点を提供し、組織が迅速に本番環境に移行できるようにするために作成されました。
参照	Security Jump Start - ACL ルール

設定手順

インスタンスでこのプラグインがアクティブ化されていない場合は、ServiceNow サポートにお問い合わせください。この時点でプラグインをアクティブ化すると、本番環境で既に使用されているテーブルへのセキュリティアクセスが変更される可能性があります。プラグインから提供される新しい ACL ルールにアドミニストレーターが注目している場合は、必要に応じて既存のインスタンスに 1 つ以上の ACL ルールを手動で作成できます。この ACL のリストは、その場合の指針として使用できます。

インポートセット **API** 内での安全な複数挿入操作の使用 (セキュリティセンター **1.3** の新機能)

com.glide.import_set_api.insert_multiple_optimize プロパティを使用して、インポートセット API 内の複数挿入操作に *GlideRecordSecure* または *GlideRecord* のどちらを使用するかを制御します。

com.glide.import_set_api.insert_multiple_optimize が推奨値の false に設定されている場合、*GlideRecordSecure* を使用してレコードが挿入され、テーブルレベルのアクセス制御リ

スト (ACL) が評価されます。このプロパティが true に設定されている場合、*GlideRecord* がレコードの挿入に使用され、テーブルレベルの ACL は評価されません。また、インポートセット API の複数挿入 REST エンドポイント ACL (sys_id：3101b770ff2211105cf343d0653bf182) が有効であること、およびユーザーが import_transformer ロールを持っていることを確認する必要があります。

詳細情報

属性	説明
構成名	*com.glide.import_set_api.insert_multiple_optimize*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	false
デフォルト値	false
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：6.5 • CVSS スコア：中 • セキュリティリスクの詳細：このプロパティが false に設定されていない場合、特権の低いユーザーが自分の特権ロールのスコープ外のテーブルにデータを挿入できる可能性があります。
元に戻せる動作	安全なデータベース上書きとデータベース上書きなしの両方。
依存関係と前提条件	なし
機能への影響度	このプロパティは、GlideRecord を使用してデータを保存することで、インポートセット API のパフォーマンスを最適化します。パラメーターが設定されている場合、API にアクセスするには、統合ユーザーが import_transformer ロールを持っている必要があります。
参照	https://developer.servicenow.com/blog.do?p=/post/gliderecord-vs-gliderecordsecure

SOAP 要求の厳格なセキュリティを強制する (Security Center 1.3 で更新)

glide.soap.strict_security プロパティを使用して、Web サービスのセキュリティを適用します。

このプロパティでは、次の組み合わせを使用します。

- HTTP プロトコルを介したベーシック認証のチャレンジ/応答、および
- Security Jump Start プラグイン (ACL ルール) を有効化する (Security Center 1.3 で更新) のシステムレベルのアクセス制御。

このプロパティを **true** に設定すると、次のアクションが実行されます。

- ユーザーが操作を実行するための適切なロールを持っている場合は、受信 SOAP 要求に検証するロール承認がないかチェックします。これは、作成、読み取り、更新、または削除操作を実行するときに、ServiceNow AI Platform テーブルに対して行われる SOAP Web サービスの呼び出し/要求中に発生します。
- テーブルのデータを SOAP データの形式で取得するときに、システムレベルの ACL をチェックします。
- フィールドレベルの ACL で、テーブルのフィールドに対して実行された CRUD 操作がないかチェックします。

ACL チェックは標準テーブル API 呼び出しに対してのみ完了し、Web サービスに対しては適用されません。

詳細情報

属性	説明
プロパティ名	*glide.soap.strict_security*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	アクセス制御
デフォルト値	true
推奨値	true
機能への影響	この修正では、インスタンス上で SOAP データの形式でテーブル/ページからデータを取得する際に、システムレベルのアクセス制御が適用されます。現在このデータにアクセスしているユーザーがいる場合、ACL ルールに基づいてデータへのアクセスが制限/許可されます。 SOAP データにアクセスできるデフォルトのロールについては、「SOAP Web サービス」を参照してください。
セキュリティリスク	(中) 受信 SOAP 要求で適切な承認が構成されていない場合、無許可のユーザーがターゲットインスタンスの機密コンテンツ/データにアクセスする可能性があります。
参照	受信 SOAP の厳格なセキュリティの適用 SOAP Web サービス

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

必須の **JMS** 接続ファクトリ **(Security Center 1.3** の新機能、1.5 および **2.0** で更新)

mid.property.jms.command.allowed_factory_names プロパティは 、MID サーバーが使用できる Java Messaging Service (JMS) 接続ファクトリを制御します。

これは、JMS アクティビティまたはアクション用のプラグインに必要な一部のファクトリを対象としています。追加のファクトリを含めることは、許可されたファクトリで攻撃者が利用できる機能に依存する JDNI 挿入など、脆弱性に対する一連の攻撃のステップになる可能性があります。脆弱性が利用される可能性を防ぐために、必要なデフォルト以外のファクトリを含めないでください。

このセキュリティリスクを修正するには、mid プロパティ *mid.property.jms.command.allowed_factory_names* に指定された名前のリストを確認します。デフォルトの connectionFactory、queueConnectionFactory、および topicConnectionFactory 以外の追加の Java ファクトリ名が必要であるかを確認してください。

詳細情報

属性	説明
構成名	*mid.property.jms.command.allowed_factory_names*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	文字列
デフォルト値	*connectionFactory*、*queueConnectionFactory*、*topicConnectio*
推奨値	*connectionFactory*、*queueConnectionFactory*、*topicConnectio*
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：4.1 • CVSS スコア：中 • セキュリティリスクの詳細：MID サーバー (*com.glideapp.agent*) プラグインがアクティブな場合は、mid プロパティ *mid.property.jms.command.allowed_factory_names* に提供されている名前のリストを確認してください。デフォルトの connectionFactory、queueConnectionFactory、および topicConnectionFactory 以外の追加のファクトリ名が必要であるかを確認してください。
依存関係と前提条件	なし

ロールによるグローバルアプリ開発の制限 (Security Center 2.0 の新機能)

sn_g_app_creator.allow_global プロパティを使用して、ガイド付きアプリケーションクリエーターを使用してグローバルスコープでアプリケーションを作成できるユーザーを制御します。

*sn_g_app_creator.allow_global*が推奨値の false に設定されている場合、グローバルスコープでアプリケーションを作成するには、*sn_g_app_creator.global* ロールが必要です。逆に、安全でない値 true に設定すると、基本の sn_g_app_creator.app_creator ロールを持つすべてのユーザーがグローバルアプリケーションを作成できます。グローバルアプリケーションにはスコープ保護がないため、開発者は特定のスコープを超えて広範な機能にアクセスできます。グローバルアプリケーション開発を追加ロールを持つユーザーに制限することは、最小特権の原則に従います。

注: このプロパティは、インスタンスでは事前設定されていません。組織のニーズに応じて、このプロパティを手動で作成して構成する必要があります。

詳細情報

属性	説明
構成名	*sn_g_app_creator.allow_global*

属性	説明
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	false
デフォルト値	false
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア:3.3 • CVSS スコア：低 • セキュリティリスクの詳細:このプロパティを推奨値に設定しないと、*sn_g_app_creator.app_creator* ロールを持つすべてのユーザーが、最小特権の原則に準拠しないグローバルアプリケーションを作成する可能性があります。
依存関係と前提条件	なし
機能への影響	API (/api/now/templates) を拡張して、グローバルアプリケーションの作成の ACL とプロパティを検証しました。

外来の明示的なロールアクセスコントロール条件のレビュー (セキュリティセンター **1.5** で削除)

すべてのユーザーが内部リソースにアクセスするための snc_internal ロール、または外部リソースにアクセスするための snc_external ロールのどちらかを持つことを義務付けるために、Explicit Roles プラグインが推奨されます。

このプラグインをインストールすると、すべての既存のユーザーに snc_internal ロールがアサインされ、既存のアクセス制御リスト (ACL) にロールの条件が設定されます。自動化ロジックまたはインスタンスアドミンによる介入により、より厳格なロール要件がすでに含まれている ACL に、snc_internal ロールまたは snc_external ロールが誤って追加される場合があります。ACL ロールの評価は、ACL にマップされたロールを含むすべてのユーザーで合格するため、snc_internal または snc_external を追加すると、ACL の意図した目的に対して広範になりすぎる可能性があります。これにより、特権の低いユーザーに ACL を介してアクセスが許可された場合、データの漏洩が発生する可能性があります。

たとえば、snc_internal ロールと admin ロールの両方をテーブル内の同じ ACL にマップする必要はありません。ACL がアドミンにアクセスを許可することを目的としている場合、snc_internal ロールは間違いです。または、ACL はすべての snc_internal ユーザーにアクセスを許可することを目的としている場合、admin ロールは不要になります。Explicit Roles プラグインがインストールされている場合は、snc_internal または snc_external のロール条件が含まれている ACL に、別のロールの条件も含まれていることを確認してください。ロールが特定のユースケースに対して機能できる場合は、その結果を定期的にレビューする必要があります。

重要: このハードニング設定は、次の Security Center v1.5 ストアパッチリリースおよび将来のバージョンで削除される予定です。Washington リリースでは、「明示的ロール ACL 構成チェックスイート」と呼ばれるインスタンススキャンスイートを利用できます。この新しいインスタンススキャンの結果を確認することをお勧めします。

SOAP 要求のゲストユーザーを設定する **(Security Center 1.3** および **2.0** で更新)

このプロパティを構成して、非認証 SOAP 要求のアクセスレベルを制御します。

このプロパティは、非認証 SOAP 要求のアクセスレベルを制御します。このプロパティが推奨値 *soap.guest* に設定されていない場合、または権限が制限されたユーザーに設定されている場合は、SOAP 要求がそのユーザーの代わりに実行されます。このプロパティが空白の場合、アドミニストレーターレベルまたはメンテナンスレベルの操作への認証されていないアクセスが可能になり、インスタンス内のすべてのセキュリティ制御が無効になります。

詳細情報

属性	説明
構成名	*com.glide.soap.guest_user*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	文字列
推奨値	soap.guest
デフォルト値	soap.guest
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：8.1 • CVSS スコア：高 • セキュリティリスクの詳細：このプロパティを空白に設定すると、アドミニストレーターまたはメンテナンスレベルの操作への認証されていないアクセスが可能になります。
依存関係と前提条件	なし

お気に入りへのパブリックアクセスを無効にする (セキュリティセンター **1.3** および **2.0** で更新)

glide.ui.magellan.favorites.allow_public を使用して、認証されていないユーザーがナビゲーターで [お気に入り] を表示できるようにするかどうかを指定します。

お気に入りへのパブリックアクセスは、 *glide.ui.magellan.favorites.allow_public* が **false** に設定されている場合に準拠しています。

詳細情報

属性	説明
プロパティ名	*glide.ui.magellan.favorites.allow_public*
構成タイプ	システムプロパティ (/sys_properties_list.do)
インスタンスセキュリティセンターでの構成	あり
目的	認証されていないユーザーがナビゲーターで [お気に入り] を表示できるようにするかどうかを制御します。
タイプ	True/False
推奨値	*false*

属性	説明
セキュリティ依存関係	*glide.ui.magellan.favorites.allow_public* を **false** に設定します。
機能への影響度	(中) このプロパティを有効にすると、権限のないユーザーから保護するレイヤーとして機能します。
セキュリティリスク	(中)このプロパティが有効になっていない場合、機密データに不正にアクセスされるリスクがあります。
参照	お気に入りオプションの構成

SNC アクセスコントロールプラグインを有効化する (セキュリティセンター 1.3 で更新)

SNC Access Control (com.snc.snc_access_control) プラグインをアクティブ化して、 カスタマーサービス & サポート 担当者によるインスタンスへのアクセスを制御します。

SNC アクセス制御 (com.snc.snc_access_control) プラグインを有効にして ServiceNow カスタマーサービスとサポート担当者が明示的な許可なくインスタンスにアクセスできないようにします。インスタンスへのアクセスはすべて監査されますが、このアクセスを制御することもできます。このアクセス方法は監査可能および追跡済み。

> **注:** 製品をサポートおよび管理する能力を持つその他の認定 ServiceNow 運用担当者は、基盤となるインフラストラクチャで管理アクションを実行する必要があります。このプラグインを有効にすると、サポートサービスレベルと可用性 SLA に影響する場合があります。可用性 SLA は、サポートスタッフ担当者がインスタンスへのアクセスを許可された時間から測定されます。

SNC アクセス制御 (com.snc.snc_access_control) プラグインを有効にして、明示的な権限なしでインスタンスへのアクセスを制限します。この機能の詳細については、「ServiceNow アクセス制御」を参照してください。アクティブ化情報については、次を参照してください。 ServiceNowアクセス制御を有効にする

詳細情報

属性	説明
プラグイン名	*com.snc.snc_access_control*
構成タイプ	[システム定義] > [プラグイン]
カテゴリ	アクセス制御
目的	カスタマーサービス & サポートの従業員によるインスタンスへのアクセスの禁止
推奨値	アクティブ
デフォルト値	なし。これはプラグインであり、Glide プロパティではないため、デフォルト値はありません。デフォルトでは、このプラグインはインストールされません。
必要なロール	顧客アドミニストレーターはプラグインをアクティブ化できません。プラグインをアクティブ化するには昇格された権限が必要なため、明示的に要求する必要があります。
セキュリティリスク評価	3.3

属性	説明
機能への影響	このプラグインが非アクティブの場合、すべての従業員が顧客のインスタンスにアクセスできます カスタマーサービス & サポート 。プラグインを有効にすると、顧客は承認されたカスタマーサービス & サポートの従業員のみにアクセスを制限できます。
セキュリティリスク	(高) より幅広いユーザーグループに対してインスタンスアクセスが不必要に公開されます。
参照	ServiceNow アクセス制御

設定手順

1. プラグインを要求するには、「ServiceNowアクセス制御を有効にする」の手順に従います。顧客は HI から SNC Access Control プラグイン (com.snc.snc_access_control) を要求する必要があります。
2. SNC アクセス制御を有効にするには、「ServiceNowアクセス制御の構成」の手順に従います。アクセス制御レコードを設定して、インスタンスにログインする権限を持つカスタマーサービス & サポートの従業員を 1 人以上指定します。

ドキュメント分類を使用して、パブリックにアクセス可能なドキュメントを制限する (Security Center 7.0 の新機能)

システムプロパティを使用して、固定リンクドキュメントへのパブリックアクセスを制御します。

> **注:** このハードニング設定は、ハードニングベースラインの一部ではありません。セキュリティセンターの強化ページには表示されず、ハードニングスコアに影響します。

デフォルトでは、ドキュメント管理プラグインには次の分類が含まれています。

- 公開
- 制限付き
- 機密
- なし (分類なし)

*com.snc.documents.permalink.allowed_classifications*を使用して、ドキュメント分類のリストを作成します。これらの分類のドキュメント (および分類されていないドキュメント) は、適切なリンクを使用して、認証されていないすべてのユーザーに公開されます。

このプロパティを使用して、ドキュメントの固定リンクへのパブリックアクセスを制御します。以前は、これらのリンクは、リンクを知っている人なら誰でも公開できました。このプロパティの値は、特定のニーズによって異なります。パブリックアクセスを有効にするために、このプロパティに追加のカスタムドキュメントカテゴリを追加する必要がある場合があります。

*com.snc.documents.permalink.allowed_classifications*プロパティの値をドキュメント分類のカンマ区切りリストに設定します。これらの分類を持つドキュメントは、非認証ユーザーによって公開されます。

このプロパティがシステムプロパティ [sys_properties] テーブルに存在しない場合、デフォルトで空のリストになります。この場合、分類されていないドキュメントのみが、非認証ユーザーによって公開されます。このプロパティのデフォルト値は **public** です。これは、 公開 として分類されたドキュメントまたは分類されていないドキュメントに、非認証ユーザーがアクセスできることを意味します。

詳細情報

属性	説明
構成名	*com.snc.documents.permalink.allowed_classifications*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	文字列リスト
推奨値	公開
デフォルト値	公開
フォールバック値	<空>
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：5.3 • CVSS スコア：中 • 公開アクセスすべきではないドキュメントを含むドキュメントカテゴリをこのプロパティに追加すると、機密情報が非認証ユーザーに公開される可能性があります。
機能への影響	パブリックにアクセス可能にする必要があるドキュメントを含むドキュメントカテゴリがこのプロパティに追加されていない場合、それらのドキュメントにアクセスできません。
依存関係と前提条件	なし

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

Glide DB 関数のクエリー ACL を検証する (Security Center 7.0 の新機能)

システムプロパティを使用して、クエリー ACL を Glide DB 機能に適用するかどうかを制御します。

glide.db.encoded_query.check_function_field_query_acls システムプロパティが **true** に設定されている場合、クエリ ACL (query_range および query_match) は、次の Glide DB 関数でデフォルトで検証されます。

- glidefunction:位置
- glidefunction:サブ文字列
- glidefunction:concat
- glidefunction:coalesce
- glidefunction:長さ

追加関数でこれらのクエリー ACL を検証するには、それらの関数を *glide.db.encoded_query.force_query_range_on_functions* システムプロパティに追加します。

*glide.db.encoded_query.check_function_field_query_acls*システムプロパティを **true** に設定するか、プロパティがシステムプロパティ [sys_properties] リストにないことを確認します。

詳細情報

属性	説明
構成名	• *glide.db.encoded_query.check_function_field_query_acls* • *glide.db.encoded_query.force_query_range_on_functions*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	• ブール • 文字列リスト
推奨値	• true • glidefunction:position、glidefunction:substring、glidefunction:conca
デフォルト値	• true • glidefunction:position、glidefunction:substring、glidefunction:conca
フォールバック値	• true • glidefunction:position、glidefunction:substring、glidefunction:conca
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア:5.3 • CVSS スコア：中 • *glide.db.encoded_query.check_function_field_query_acls* システムプロパティが存在し、値が **true** に設定されていない場合、ログインしているユーザーがデータを盲目的に推測でき、機密情報が開示される可能性があります。
機能への影響	ユーザーは、機能フィールドの値が表示されることを期待しても、機能フィールドに ACL が適用されているためにブロックされる場合があります。
依存関係と前提条件	なし

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

API と Web サービス

API および Web サービスカテゴリでは、アプリケーションに適切な認証、承認、およびセッション管理があることを確認し、信頼境界を横切るすべての入力を検証します。またそのカテゴリにはすべての API タイプのセキュリティコントロールが含まれています。

このカテゴリの特定のコントロールは、SOAP Web サービスの XDS スキーマ検証や graphQL API のサービス拒否保護など、サービスタイプ別の入力検証に対応します。

SOAP コンテンツタイプの検証 (セキュリティセンター 1.3 で更新)

glide.soap.require_content_type_xml プロパティを使用して、テキスト/xml のコンテンツタイプの検証を有効にし、無効な SOAP 要求から保護します。

- **true** に設定すると、ServiceNow AI Platform はテキスト/xml のコンテンツタイプを検証して無効な SOAP 要求から保護します。
- **false** に設定すると、すべてのコンテンツタイプの値が許可されます。

⚠ 警告: これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

詳細情報

属性	説明
プロパティ名	*glide.soap.require_content_type_xml*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	API と Web サービス
目的	無効な SOAP 要求から保護すること
推奨値	true
デフォルト値	true
セキュリティリスク評価	8.8
機能への影響	この修正により、すべての受信 SOAP 要求に対して SOAP コンテンツタイプの検証が有効になります。 • 受信要求に text/xml 以外のコンテンツタイプを使用している場合は、SOAP トランザクションが失敗する可能性があります。 • 正しい MIME タイプを使用していない場合、サードパーティの統合が中断される可能性があります。
セキュリティリスク	(中) 受信 SOAP 要求を受け入れる場合、適切な検証が実行され、関連するコンテンツタイプが要求の一部として定義されていることが確認されます。これにより、セキュリティリスクと見なされる可能性のある無効な SOAP 応答が制限されます。
参照	コンテンツタイプ

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

PDF 要求に認証を必須とする (セキュリティセンター 1.3 で更新)

glide.basicauth.required.pdf プロパティを使用して、受信 PDF 要求でベーシック認証を要求するかどうかを指定します。

プロパティ *glide.basicauth.required.pdf* がsys_propertiesテーブルに存在し、true に設定されていることを確認します。

詳細情報

⚠ **警告:** これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

属性	説明
プロパティ名	*glide.basicauth.required.pdf*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	API と Web サービス
目的	PDF 要求に対してベーシック認証を適用すること
推奨値	true
デフォルト値	true
セキュリティリスク評価	7.5
機能への影響	この修正では、ベーシック認証とシステムレベルのアクセス制御の形で、認証方法の組み合わせが適用されます。 • インスタンス上で PDF データの形式でテーブル/ページからデータを取得する際に、この認証を実行します。 • 現在このデータにアクセスしているゲストユーザーを制限します。必要であれば、このコンテンツへのアクセスを必要とするユーザーのために、必要なアクセス制御権限を持つ新しいアカウントを作成する場合があります。 詳細については、「Web サービスインポートセット」を参照してください。
セキュリティリスク	(高) 受信 PDF 要求で適切な承認が構成されていない場合、無許可のユーザーがターゲットインスタンスの機密コンテンツ/データにアクセスする可能性があります。

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

イベント管理 **HTTP** プロセッサで認証を必須とする **(Security Center 1.3** の新機能、1.5 で更新、2.0 で削除)

イベント管理プラグイン (*com.glideapp.itom.snac*) が有効になっている場合に、Amazon Simple Notification Service (SNS) の受信要求に対して安全なベーシック認証を確立する方法について説明します。

glide.basicauth.required.evtmgmthttpprocessor プロパティが推奨値の **true** に設定されておらず、イベント管理プラグイン (*com.glideapp.itom.snac*) がアクティブな場合、すべてのインバウンド Amazon Simple Notification Service (SNS) 要求に対してベーシック認証は必要ありません。これにより、インスタンスデータへの非認証アクセスが発生する可能性があります。

このセキュリティリスクを修復するには、*glide.basicauth.required.evtmgmthttpprocessor* が **true** に設定され、*com.glideapp.itom.snac* がアクティブになっていることを確認してください。

詳細情報

属性	説明
構成名	*glide.basicauth.required.evtmgmthttpprocessor*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	API と Web サービス
セキュリティリスク	• 重大度スコア：7 • CVSS スコア：高 • セキュリティリスクの詳細：*glide.basicauth.required.evtmgmthttpprocessor* が推奨値の true に設定されておらず、*com.glideapp.itom.snac* プラグインをアクティブにしていない場合、インバウンド SNS 要求に対してベーシック認証が有効になりません。これにより、インスタンスデータへの非認証アクセスが発生する可能性があります。
依存関係と前提条件	なし
参照	• https://docs.aws.amazon.com/sns/latest/dg/welcome.html • アクセス制御
機能への影響度	*glide.basicauth.required.evtmgmthttpprocessor* が推奨値の true に設定されておらず、イベント管理プラグイン (com.glideapp.itom.snac) がアクティブな場合、すべてのインバウンド Amazon Simple Notification Service (SNS) 要求に対してベーシック認証は必要ありません。これにより、インスタンスデータへの非認証アクセスが発生する可能性があります。

SOAP 要求に認証を必須とする **(Security Center** 1.3、1.5、および **2.0** で更新)

glide.basicauth.required.soap プロパティを使用して、受信 SOAP 要求で基本承認を要求するかどうかを指定します。

詳細情報

⚠ **警告:** これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

属性	説明
プロパティ名	*glide.basicauth.required.soapglide.soap.require_ws_securi*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	API と Web サービス
目的	SOAP 要求承認の適用
推奨値	true
セキュリティリスク評価	8.1
機能への影響	この修正では、ベーシック認証とシステムレベルのアクセス制御の形で、認証方法の組み合わせが適用されます。 • インスタンス上で SOAP データの形式でテーブル/ページからデータを取得する際に、この認証を実行します。 • 現在このデータにアクセスしているゲストユーザーを制限します。 • このコンテンツへのアクセスを必要とするユーザーのために、必要なアクセス制御権限を持つアカウントを作成します。 詳細については、「SOAP Web サービス」および「MID サーバー認証情報と SOAP 要求」を参照してください。
セキュリティリスク	(中) データソースの SOAP 要求で適切な承認が構成されていない場合、無許可のユーザーがターゲットインスタンスの機密コンテンツ/データにアクセスする可能性があります。
参照	認証

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

アンロード要求に認証を必須とする (Security Center 1.3 で更新)

glide.basicauth.required.unl (useUnloadFormat) プロパティを使用して、受信アンロード要求でベーシック認証を要求するかどうかを指定します。

詳細情報

属性	説明
プロパティ名	*glide.basicauth.required.unl*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	API と Web サービス
目的	アンロード要求に対してベーシック認証を適用すること
推奨値	true
セキュリティリスク評価	7.5

属性	説明
機能への影響	この修正では、ベーシック認証とシステムレベルのアクセス制御の形で、認証方法の組み合わせが適用されます。インスタンス上でアンロードデータの形式でテーブル/ページからデータを取得する際に、この認証を実行します。
セキュリティリスク	(高) データソースのアンロード要求で適切な承認が構成されていない場合、無許可のユーザーがターゲットインスタンスの機密コンテンツ/データにアクセスする可能性があります。*glide.basicauth.required.unl* が sys_properties_table にあり、true に設定されていることを確認します。
参照	認証

csv 要求に認証を必須とする (Security Center 1.3 で更新)

glide.basicauth.required.csv プロパティを使用して、受信 CSV (カンマ区切り値) 要求でベーシック認証を要求するかどうかを指定します。

詳細情報

⚠ **警告:** これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

属性	説明
プロパティ名	*glide.basicauth.required.csv*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	API と Web サービス
目的	CSV 要求に対してベーシック認証を適用すること
推奨値	true
セキュリティリスク評価	7.5
機能への影響	この修正では、ベーシック認証とシステムレベルのアクセス制御の形で、認証方法の組み合わせが適用されます。 • インスタンス上の CSV データの形式でテーブル/ページからデータを取得する際に、この認証を実行します。 • 現在このデータにアクセスしているゲストユーザーを制限します。必要であれば、このコンテンツへのアクセスを必要とするユーザーのために、必要なアクセス制御権限を持つ新しいアカウントを作成する場合があります。 詳細については、「CSV 形式のファイルからのデータの取得」を参照してください。
セキュリティリスク	(高) 受信 CSV 要求で適切な承認が構成されていない場合、無許可のユーザーがターゲットインスタンスの機密コンテンツとデータにアクセスする可能性があります。*glide.basicauth.required.csv* が sys_properties_table にあり、true に設定されていることを確認します。

属性	説明
参照	Web サービスセキュリティ

Excel 要求に認証を必須とする (Security Center 1.3 で更新)

glide.basicauth.required.excel プロパティを使用して、受信 Excel 要求でベーシック認証を要求するかどうかを指定します。

詳細情報

警告: これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

属性	説明
プロパティ名	*glide.basicauth.required.excel*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	API と Web サービス
目的	Excel 要求に対してベーシック認証を適用すること
推奨値	true
セキュリティリスク評価	7.5
機能への影響	この修正では、ベーシック認証とシステムレベルのアクセス制御の形で、認証方法の組み合わせが適用されます。 • インスタンス上で Excel データの形式でテーブル/ページからデータを取得する際に、この認証を実行します。 • 現在このデータにアクセスしているゲストユーザーを制限します。必要であれば、このコンテンツへのアクセスを必要とするユーザーのために、必要なアクセス制御権限を持つ新しいアカウントを作成する場合があります。
セキュリティリスク	(高) 受信 Excel 要求で適切な承認が構成されていない場合、無許可のユーザーがターゲットインスタンスの機密コンテンツ/データにアクセスする可能性があります。

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

インポート要求に認証を必須とする (Security Center 1.3 で更新)

glide.basicauth.required.importprocessor プロパティを使用して、受信インポート要求でベーシック認証を要求するかどうかを指定します。

詳細情報

警告: これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

属性	説明
プロパティ名	*glide.basicauth.required.importprocessor*

属性	説明
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	API と Web サービス
目的	インポート要求にベーシック認証を適用すること
推奨値	true
セキュリティリスク評価	5.3
機能への影響	この修正では、ベーシック認証とシステムレベルのアクセス制御の形で、認証方法の組み合わせが適用されます。 • データソースをインスタンステーブル/ページにインポートするときに、この認証を実行します。 • 現在このデータにアクセスしているゲストユーザーを制限します。必要であれば、このコンテンツへのアクセスを必要とするユーザーのために、必要なアクセス制御権限を持つ新しいアカウントを作成する場合があります。 詳細については、「CSV 形式のファイルからのデータの取得」を参照してください。
セキュリティリスク	(中) データソースインポート要求で適切な承認が構成されていない場合、無許可のユーザーがターゲットインスタンスの機密コンテンツ/データにアクセスする可能性があります。
参照	SOAP Web Services のセキュリティ SOAP Web サービス

JSONv2 要求に認証を必須とする (Security Center 1.3 で更新)

glide.basicauth.required.jsonv2 プロパティを使用して、受信 JSONv2 要求で基本承認を要求するかどうかを指定します。

詳細情報

⚠ **警告:** これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

属性	説明
プロパティ名	*glide.basicauth.required.jsonv2*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	API と Web サービス
目的	JSONv2 要求承認の適用
セキュリティリスク評価	7.5
推奨値	true
機能への影響	この修正では、ベーシック認証とシステムレベルのアクセス制御の形で、認証方法の組み合わせが適用されます。

属性	説明
	• インスタンス上で JSON データの形式でテーブル/ページからデータを取得する際に、この認証を実行します。 • 現在このデータにアクセスしているゲストユーザーを制限します。 • このコンテンツへのアクセスを必要とするユーザーのために、必要なアクセス制御権限を持つアカウントを作成します。 詳細については、「JSONv2 Web サービス JSONv2 Web サービス 」を参照してください。
セキュリティリスク	(高) データソースの JSON 要求で適切な承認が構成されていない場合、無許可のユーザーがターゲットインスタンスの機密コンテンツ/データにアクセスする可能性があります。
参照	認証 着信 JSONv2 要求に対するベーシック認証の要求

WSDL 要求に認証を必須とする (セキュリティセンター 1.3 および 1.5 で更新)

glide.basicauth.required.wsdl プロパティを使用して、受信 WSDL (Web Services Description Language) 要求でベーシック認証を要求するかどうかを指定します。

glide.basicauth.required.wsdl が推奨値の true に設定されていない場合、WSDL 要求のベーシック認証が無効になります。WSDL は、インスタンステーブルスキーマなどの Web サービスを記述するために使用されるプロトコルであり、テーブル内のデータを共有するためのメカニズムではありません。このプロパティを true に設定すると、非認証ユーザーにテーブルスキーマを公開できます。

注: 受信 WSDL 要求にベーシック認証を要求しないという選択をする場合は、アクセス制御 (ACL) ルールを変更してゲストユーザーが WSDL コンテンツにアクセスできるようにする必要があります。

詳細情報

警告: これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

属性	説明
プロパティ名	*glide.basicauth.required.wsdl*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	API と Web サービス
目的	WSDL 要求に対してベーシック認証を適用すること
推奨値	true
セキュリティリスク評価	5.3
機能への影響	この修正では、ベーシック認証とシステムレベルのアクセス制御の形で、認証方法の組み合わせが適用されます。

属性	説明
	• インスタンス上で WSDL データの形式でテーブル/ページからデータを取得する際に、この認証を実行します。 • 現在このデータにアクセスしているゲストユーザーを制限します。必要であれば、このコンテンツへのアクセスを必要とするユーザーのために、必要なアクセス制御権限を持つ新しいアカウントを作成する場合があります。
セキュリティリスク	(中) WSDL Web サービスで適切な承認が構成されていない場合、無許可のユーザーがターゲットインスタンスの機密 WSDL コンテンツ/データにアクセスする可能性があります。
参照	Web サービスセキュリティ

XML 要求に対する認証を必須とする (セキュリティセンター 1.3 で更新)

glide.basicauth.required.xml プロパティを使用して、受信 XML 要求でベーシック認証を要求するかどうかを指定します。

詳細情報

⚠ **警告:** これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

属性	説明
プロパティ名	*glide.basicauth.required.xml*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	API と Web サービス
目的	XML 要求に対してベーシック認証を適用すること
セキュリティリスク評価	7.5
推奨値	true
機能への影響	この修正では、ベーシック認証とシステムレベルのアクセス制御の形で、認証方法の組み合わせが適用されます。 • インスタンス上で XML データの形式でテーブル/ページからデータを取得する際に、この認証を実行します。 • 現在このデータにアクセスしているゲストユーザーを制限します。必要であれば、このコンテンツへのアクセスを必要とするユーザーのために、必要なアクセス制御権限を持つ新しいアカウントを作成する場合があります。 詳細については、「XML パーサーステップ」を参照してください。
セキュリティリスク	(高) 受信 XML 要求で適切な承認が構成されていない場合、無許可のユーザーがターゲットインスタンスの機密コンテンツ/データにアクセスする可能性があります。
参照	認証

XML 出力要求に対する認証を必須とする (Security Center 1.3 で更新)

すべての受信 XMLOutputProcessor 要求に対して基本認証が必要となるように、このプロパティを構成します。

glide.basicauth.required.xmloutputprocessor プロパティが推奨値の **true** に設定されていない場合、受信 XMLOutputProcessor 要求に基本認証は必要ありません。これにより、インスタンスから情報が非認証で開示される可能性があります。

⚠ 警告: これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

詳細情報

属性	説明
構成名	*glide.basicauth.required.xmloutputprocessor*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	API と Web サービス
セキュリティリスク	• 重大度スコア：7.5 • CVSS スコア：高 • セキュリティリスクの詳細：プロパティを推奨値の **true** に設定しないと、機密情報がインスタンスから漏洩する可能性があります。
依存関係と前提条件	なし

XSD 要求に認証を必須とする (Security Center 1.3 で更新)

glide.basicauth.required.xsd プロパティを使用して、受信 XSD (XML Schema Definition) 要求でベーシック認証を要求するかどうかを指定します。

プロパティ *glide.basicauth.required.xsd* がsys_propertiesテーブルに存在し、true に設定されていることを確認します。

詳細情報

⚠ 警告: これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

属性	説明
プロパティ名	glide.basicauth.required.xsd*glide.basicauth.required.xsd*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	API と Web サービス

属性	説明
目的	XSD 要求に対してベーシック認証を適用すること
推奨値	true
セキュリティリスク評価	5.3
機能への影響	この修正では、ベーシック認証とシステムレベルのアクセス制御の形で、認証方法の組み合わせが適用されます。 • インスタンス上で XSD データの形式でテーブル/ページからデータを取得する際に、この認証を実行します。 • 現在このデータにアクセスしているゲストユーザーを制限します。必要であれば、このコンテンツへのアクセスを必要とするユーザーのために、必要なアクセス制御権限を持つ新しいアカウントを作成する場合があります。 詳細については、「非インタラクティブセッション」を参照してください。
セキュリティリスク	(中) 受信 XSD 要求で適切な承認が構成されていない場合、無許可のユーザーがターゲットインスタンスの機密コンテンツ/データにアクセスする可能性があります。
参照	認証

スクリプト要求に認証を必須とする (Security Center 1.3 で更新)

glide.basicauth.required.scriptedprocessor プロパティを使用して、受信スクリプト要求でベーシック認証を要求するかどうかを指定します。

詳細情報

⚠ **警告:** これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

属性	説明
プロパティ名	*glide.basicauth.required.scriptedprocessor*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	API と Web サービス
目的	スクリプト要求に対してベーシック認証を適用すること
推奨値	true
セキュリティリスク評価	7.2
機能への影響	この修正では、ベーシック承認の形式で認証が適用されます。

属性	説明
	• インスタンスでスクリプト要求を処理する際に、この認証を実行します。 • 現在このデータにアクセスしているゲストユーザーを制限します。必要であれば、このコンテンツへのアクセスを必要とするユーザーのために、必要なアクセス制御権限を持つ新しいアカウントを作成する場合があります。
セキュリティリスク	(高) 受信スクリプト要求で適切な承認が構成されていない場合、無許可のユーザーがターゲットインスタンスの機密コンテンツ/データにアクセスします。
参照	認証

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

スキーマ要求に認証を必須とする (Security Center 1.3 で更新)

glide.basicauth.required.schema プロパティを使用して、すべての受信テーブルスキーマプロセッサ要求に対して基本認証を要求します。

受信 Table Schema Processor は、プラットフォームに対する受信スキーマ要求を処理します。

すべての受信 Table Schema Processor 要求に対して基本認証を要求するには、*glide.basicauth.required.schema* を推奨値の **true** に設定します。すべての受信 Table Schema Processor 要求に対して基本認証を要求しないようにするには、この値を **false** に設定します。

⚠ **警告:** これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

詳細情報

属性	説明
プロパティ名	*glide.basicauth.required.schema*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	API と Web サービス
目的	すべての受信 Table Schema Processor 要求に対して基本認証を要求します。
推奨値	True (デフォルト値)。
構成タイプ	ブーリアン
セキュリティリスク	(中) このプロセッサーからの認証を省略すると、インスタンスデータへの不正なアクセスが発生します。

RSS 要求に認証を必須とする (Security Center 1.3 で更新)

glide.basicauth.required.rss プロパティを使用して、受信 RSS 要求でベーシック認証を要求するかどうかを指定します。

詳細情報

⚠ 警告: これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

属性	説明
プロパティ名	*glide.basicauth.required.rss*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	API と Web サービス
目的	RSS 要求に対してベーシック認証を適用すること
推奨値	true
セキュリティリスク評価	7.5
機能への影響	この修正では、ベーシック認証とシステムレベルのアクセス制御の形で、認証方法の組み合わせが適用されます。 • インスタンスで RSS 要求を処理する際に、この認証を実行します。 • 現在このデータにアクセスしているゲストユーザーを制限します。必要であれば、このコンテンツへのアクセスを必要とするユーザーのために、必要なアクセス制御権限を持つ新しいアカウントを作成する場合があります。 詳細については、「RSS フィードジェネレーター」を参照してください。
セキュリティリスク	(高) 受信 RSS 要求で適切な承認が構成されていない場合、無許可のユーザーがターゲットインスタンスの機密コンテンツ/データにアクセスする可能性があります。
参照	RSS ベーシック認証

API 要求に認証を必須とする (Security Center 1.3 で更新)

glide.basicauth.required.api プロパティを使用して、受信 REST 要求の基本認証のセキュリティを強化します。

すべての REST 要求に対して認証を要求するには、*glide.basicauth.required.api* プロパティを **true** に設定します。すべての REST 要求に対して認証をバイパスするには、プロパティを **false** に設定します。

⚠ 警告: これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

詳細情報

属性	説明
プロパティ名	*glide.basicauth.required.api*
構成タイプ	システムプロパティ (/sys_properties_list.do)

属性	説明
カテゴリ	API と Web サービス
目的	着信 RSS 要求の基本認証
推奨値	True (デフォルト)
構成タイプ	文字列
セキュリティリスク	(高) 「glide.basicauth.required.api」が推奨値の「true」に設定されていない場合、API 要求でベーシック認証が無効になり、インスタンスデータへの非認証アクセスが発生します。
セキュリティリスク評価	8.6

アーキテクチャ、設計、および脅威のモデル化

この幅広いコントロールは、安全なアプリケーションを実装するための高レベルの設計上の考慮事項と重要な要素に対応しています。これには、可用性、機密性処理の整合性、否認防止、およびプライバシーが含まれます。さらに、安全なソフトウェア開発ライフサイクルの要素が含まれています。

証明書ベースの認証が強制されていません (セキュリティセンター 1.3 の新機能)

glide.authenticate.mutual.enabled プロパティは、証明書ベースの認証を有効にします。これは、ServiceNow AI Platform の REST および SOAP API への受信 REST 接続に対する相互認証の一種です。

相互認証では、SSL (Secure Socket Layer) 証明書を交換し、信頼できる認証局で証明書を検証することによって、サーバーとクライアント間の信頼を確立します。これにより、信頼できるソースが ServiceNow AI Platform に接続していることを検証できます。このインスタンスを推奨値の true に設定しないと、インスタンスが中間者攻撃 (MitM) に対して脆弱になる可能性があります。

セキュリティ上の脅威を緩和するために、受信 Web サービスの相互認証を有効にします。証明書ベースの認証プラグイン (*com.glide.auth.mutual*) を ServiceNow AI Platform に初めてインストールする場合は、「証明書ベースの認証の設定」の指示に従ってください。さらに、プラグインをアクティブ化するには、*glide.authenticate.mutual.enabled* プロパティが true に設定されていることを確認してください。

詳細情報

属性	説明
構成名	*glide.authenticate.mutual.enabled*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	アーキテクチャ、設計、および脅威のモデル化
セキュリティリスク	• 重大度スコア：5.3 • CVSS スコア：中

属性	説明
	• セキュリティリスクの詳細：このプロパティが推奨値の true に設定されていない場合、証明書ベースの認証では、信頼できる認証局で証明書が検証されません。これにより、攻撃者が MitM 攻撃を使用してインスタンスを攻撃する可能性が高まります。
依存関係と前提条件	なし
参照	• https://csrc.nist.gov/glossary/term/man_in_the_middle_attack • 証明書ベースの認証 • 相互認証の構成

HR アプリの ACL 評価で代理操作をチェックする (セキュリティセンター 1.3 の新機能、1.5 で更新)

sn_hr_core.impersonateCheck プロパティを使用して、ユーザーが別のユーザーの代理操作を行って自身の HR 情報にアクセスすることを防止します。

安全な設定にすると、アドミニストレーターは代理操作の使用中に他のユーザーの人事情報を表示できなくなります。このプロパティを安全でない設定にすると、アドミニストレーターがユーザーの代理操作を行い、代理操作されたユーザーのアクセス権でアンケート結果や監査レコードなどの HR データにアクセスできるようになります。メールなどのユーザー自身だけが利用できる情報など、このタイプのデータの性質上、これはお勧めしません。*sn_hr_core.impersonateCheck* を true に設定すると 、ユーザーが他のユーザーの代理操作を行っていない場合にのみ、HR 情報へのアクセスが許可されます。

詳細情報

属性	説明
構成名	*sn_hr_core.impersonateCheck*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	false
カテゴリ	アーキテクチャ、設計、および脅威のモデル化
セキュリティリスク	• 重大度スコア：2.7 • CVSS スコア：低 • セキュリティリスクの詳細：このプロパティを安全でない設定にすると、アドミニストレーターがユーザーの代理操作を行い、代理操作されたユーザーのアクセス権でアンケート結果や監査レコードなどの HR データにアクセスできるようになります。

属性	説明
依存関係と前提条件	なし
機能への影響度	このプロパティを true に設定すると、アドミニストレーターは代理操作の使用中に他のユーザーの HR 情報を表示できなくなります。false に設定すると、アドミニストレーターはユーザーの代理操作を行い、代理操作されたユーザーのアクセス権でアンケート結果や監査レコードなどの HR データにアクセスできるようになります。メールなどのユーザー自身だけが利用できる情報など、このタイプのデータの性質上、これはお勧めしません。sn_hr_core.impersonateCheck を true に設定すると 、ユーザーが他のユーザーの代理操作を行っていない場合にのみ、HR 情報へのアクセスが許可されます。

シングルサインオン **(SSO)** が有効になっているユーザーのローカルログインを無効にする

ユーザーレコードを更新して、シングルサインオン (SSO) が有効になっているユーザーのローカルログインを無効にします。

SSO 認証を使用するように構成されたユーザーは、ユーザー [sys_user] レコードの **user_password** フィールドに格納されているローカル認証情報を使用して、インスタンスまたはインスタンスの一部にアクセスできる場合があります。このアクセス権は、ロックアウトされていないユーザーのインタラクティブアクセスと非インタラクティブアクセスの両方に適用されます。SSO で構成されたユーザーがローカル認証情報を使用できないようにすることで、有効なローカルログイン認証情報が盗まれ、悪意のあるユーザーによって使用される可能性を低減できます。

SSO が有効になっているインスタンスでローカルログインがまだ有効になっているアカウントを特定して対処する手順については、Now Support ナレッジベース記事 KB1649420 を確認してください。

詳細情報

属性	説明
セキュリティリスク	ユーザーに対して SSO 認証が有効になっている場合は、そのユーザーがローカルにログインできないようにすることをお勧めします。これにより、有効なローカルログイン認証情報が盗まれ、悪意のあるユーザーによってログインに使用される可能性が低くなります。
共通脆弱性スコアリングシステム (CVSS) スコア	4.2
共通脆弱性スコアリングシステム (CVSS) 評価	中
機能への影響	SSO が設定されたユーザーは、ローカル認証情報を使用してログインできます。
依存関係と前提条件	シングルサインオンを有効にする必要があります (**glide.authenticate.multisso.enabled** システムプロパティを **true** に設定します)。
データタイプ	該当なし
ベースシステム値	該当なし

属性	説明
フォールバック値	該当なし
推奨値	該当なし

認証されていない公開レポートを無効にする (Security Center 2.0 で更新)

ユーザーがレポートを公開したりアクセスしたりできないようにするには、このプロパティを無効にします。このプロパティによって、レポートの公開レポート機能が無効になります。

glide.report.published_reports.enabled を **true** に設定してレポートの公開を有効にします。

Glide プロパティ *glide.report.published_reports.enabled* が存在し、値が false に設定されていることを確認します。プロパティがsys_propertiesテーブルに含まれていない場合は、新しいレコードを追加してください。

詳細情報

属性	説明
プロパティ名	*glide.report.published_reports.enabled*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	アーキテクチャ、設計、および脅威のモデル化
目的	レポートの公開レポート機能を無効化する。
タイプ	True | False
推奨値	**false**
セキュリティリスク評価	6.5
機能への影響	ユーザーはレポートを公開できません。
セキュリティリスク	(中) このプロパティが有効でないと、ユーザーがレポートにアクセスしたり公開したりして機密データが公開される可能性があります。レポートを公開して、ユーザー以外の人を含め、誰でもレポートにアクセスできる URL を作成します。誰かがその URL に移動すると、インスタンスから現在のデータでレポートが生成されます。
参照	レポートの公開

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

インバウンドクエリ要求でのフィールド ACL の強制

インスタンスでの受信クエリの検証方法を管理します。

glide.export.query.enforce_field_acl プロパティを使用して、インスタンスでの受信クエリの検証方法を確認します。このプロパティが推奨値の **true** に設定されている場合、フィールド ACL は受信クエリと比較して確認され、ユーザーに権限がない場合は却下されます。プロパティが **false** に設定されている場合、ACL は受信クエリと比較して確認されずに実行が継続されるため、権限のない当事者に情報が開示される可能性があります。

詳細情報

属性	説明
構成名	*glide.export.query.enforce_field_acl*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	アーキテクチャ、設計、および脅威のモデル化
セキュリティリスク	• 重大度スコア：4.4 • CVSS スコア：中 • セキュリティリスクの詳細：このプロパティが **false** に設定されている場合、ACL は受信クエリと比較して確認されないため、情報が開示される可能性があります。
依存関係と前提条件	なし

レポートビューでの読み取り **ACL** の適用

読み取り ACL をインスタンスに適用する方法を管理します。

テーブルまたはフィールドにレポートビュー ACL がない場合に、*glide.report.report_view.read_acl* プロパティを使用してレポート機能の読み取り ACL (テーブルレベル) を適用します。このプロパティが **enforce** に設定されていない場合、ACL がバイパスされ、機密情報が漏洩する可能性があります。

詳細情報

属性	説明
構成名	*glide.report.report_view.read_acl*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	文字列
推奨値	enforce
デフォルト値	enforce
カテゴリ	アーキテクチャ、設計、および脅威のモデル化
セキュリティリスク	• 重大度スコア：7.1 • CVSS スコア：高 • セキュリティリスクの詳細：このプロパティを **enforce** に設定しないと、ACL がバイパスされる可能性があります。
依存関係と前提条件	なし

ナレッジクイックリンクのクエリ **ACL** を強制する

システムプロパティを使用してナレッジクイックリンクのクエリ ACL を強制します。

com.glide.security.query_acl.enabled.knowledge_quick_links システムプロパティが **true** に設定されている場合、クエリー ACL がナレッジクイックリンクに適用されます。このプロパティが **false** に設定されている場合、 GlideRecord.addEncodedQuery のデフォルトの動作により、攻撃者はブラインドクエリを使用してデータを列挙し、盗み出すことができます。

プロパティがシステムプロパティ [sys_properties] テーブルに存在しない場合は、セキュアなデフォルトの **true** が使用されます。3 つ目のオプションの **external_and_guests** は、外部ユーザーとゲストに対してのみクエリー ACL を適用します。

com.glide.security.query_acl.enabled.knowledge_quick_links システムプロパティが **true** に設定されていることを確認します。

詳細情報

属性	説明
構成名	*com.glide.security.query_acl.enabled.knowledge_quick_link*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	文字列:三項演算子として使用される
推奨値	true
デフォルト値	true
フォールバック値	true
カテゴリ	アーキテクチャ、設計、および脅威のモデル化
セキュリティリスク	• 重大度スコア：5.3 • CVSS スコア：中 • セキュリティリスク:ACL がバイパスされ、フィールドデータを表示する権限のないユーザーにフィールドデータが開示される可能性があります。悪用されるテーブルによっては、この開示に機密データが含まれる可能性があります。
機能への影響	ユーザーは、フィールドレベルのアクセス権を持たないフィールドに対して特定のクエリを実行することはできません。
依存関係と前提条件	なし

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

サブリスト、リスト数、およびウィジェットデータテーブルに対してクエリ **ACL** を強制する

システムプロパティを使用して、サブリスト、リスト数、およびウィジェットデータテーブルのクエリにクエリー ACL を適用します。

グループ化されたリストや関連リストなどのサブリストクエリにクエリー ACL を適用するには、*com.glide.security.query_acl.enabled.sub_lists*を **true** に設定します。

リストカウントクエリにクエリ ACL を適用するには、*com.glide.security.query_acl.enabled.list_count*を **true** に設定します。

ウィジェットデータテーブルにクエリー ACL を適用するには、*glide.security.query_acl.enabled.data_table*を **true** に設定します。

これらのシステムプロパティのいずれかが **false** に設定されている場合、GlideRecord.addEncodedQuery のデフォルトの動作により、攻撃者はブラインドクエリを使用してデータを列挙し、盗み出すことができます。これらのプロパティがシステムプロパティ [sys_properties] テーブルに存在しない場合は、セキュアなデフォルトの true が使用されます。3 番目のオプションの *external_and_guests* は、外部ユーザーとゲストに対してのみ ACL を適用します。

これらのシステムプロパティがシステムプロパティ [sys_properties] テーブルに表示されていないか、 **true** に設定されていることを確認してください。

詳細情報

属性	説明
構成名	• *com.glide.security.query_acl.enabled.sub_lists* • *com.glide.security.query_acl.enabled.list_count* • *glide.security.query_acl.enabled.data_table*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	• true • true • true
デフォルト値	• true • true • true
フォールバック値	• true • true • true
カテゴリ	アーキテクチャ、設計、および脅威のモデル化
セキュリティリスク	• 重大度スコア：5.3 • CVSS スコア：中

属性	説明
	• セキュリティリスク:ACL はバイパスされ、フィールドデータを表示する権限のないユーザーにフィールドデータを開示する可能性があります。これには、悪用されるテーブルによっては機密データが含まれる可能性があります。
依存関係と前提条件	なし

有効なクエリ文字列の選択を強制する (Security Center 7.0 の新機能)

システムプロパティを使用して、URL クエリ文字列を介して渡された選択肢フィールド値が、レコードの作成時に有効でアクティブな選択肢になるようにします。

glide.ui.query_string.enforce_valid_choice_on_create システムプロパティが **true** に設定されている場合、プラットフォームは、レコードの作成時に URL クエリ文字列を介して (たとえば、リストフィルターから) 渡された選択フィールド値が有効なアクティブな選択肢であることを検証します。

無効な場合、値は無視され、フィールドはデフォルト値にフォールバックします。プロパティが **false** の場合、検証は非アクティブになり、システムは無効または非アクティブな値であっても、任意の値を受け入れます。この受け入れにより、レコードに誤ったデータや予期しないデータが保存される可能性があります。

glide.ui.query_string.enforce_valid_choice_on_create がシステムプロパティ [sys_properties] テーブルに存在し、**true** に設定されていることを確認します。プロパティがテーブルに存在しない場合、フォールバック値は **false** です。

詳細情報

属性	説明
構成名	*glide.ui.query_string.enforce_valid_choice_on_create*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	false
デフォルト値	false
フォールバック値	true
カテゴリ	アーキテクチャ、設計、および脅威のモデル化
セキュリティリスク	• 重大度スコア：2.2 • CVSS スコア：低 • セキュリティリスク:このプロパティを **true** に設定すると、新しいレコードの作成に有効な選択肢のみが使用されます。無効な選択をすると、ユーザーが意図しない値を選択できてしまうという、軽微で予期しないデータ整合性の問題が発生する可能性があります。

属性	説明
機能への影響	既存のインスタンスおよびワークフローは、無効または非アクティブな選択肢を含むフィルターに基づいて作成される新しいレコードに依存する場合があります。この動作を確認するには、次のプロセスを使用します。 **1.** アドミンユーザーとしてインスタンスにログインします。 **2.** 任意のテーブルに文字列フィールドを作成します。たとえば、 **Test1** と **Test2** の 2 つの選択肢があるインシデントを考えてみましょう。 **3.** 値が **Test2** に設定された文字列フィールドを選択して、インシデントテーブルにリストフィルターを作成します。 **4.** 辞書フィールドに移動し、 選択肢 **Test2** を非アクティブ化します。 **5.** 手順 3 で選択したフィルターに移動し、[新規] ボタンを選択します。 **6.** 新しく開いたレコードの [文字列タイプ] フィールドの値を確認します。プロパティが **true** に設定されている場合、文字列フィールドに値がないか、デフォルト値を表示する必要があります。 プロパティが **false** (デフォルト) に設定されている場合、文字列フィールドの値は **Test2** に設定されます。
依存関係と前提条件	なし

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

許可される **ServiceNow** 内部 **IP** アドレスを定義する (セキュリティセンター **1.3** および **1.5** で更新)

glide.ip.authenticate.strict プロパティを使用して、インスタンスで受信接続を確立できる IP 範囲を指定します。

glide.ip.authenticate.strict および **glide.ip.authenticate.allow.secured** システムプロパティを使用して、インスタンスアクセスがより広範なユーザーグループに不必要に公開されないようにします。

glide.ip.authenticate.strict システムプロパティが **true** に設定されている場合、内部ServiceNow担当者とシステムは、必須 IP 範囲からインスタンスへの受信接続のみを行うことができます。これにより、 ServiceNowの可視化がインスタンスの重要な内部インフラストラクチャに制限され、サポートスタッフや営業スタッフなどのより広範な ServiceNow 担当者が企業ネットワークを介してアクセスできなくなります。**glide.ip.authenticate.allow.secured** システムプロパティは、通常の認証済みアクセスと非認証診断ページを含む内部ServiceNow受信接続を許可します。

true に設定されていない場合は、**glide.ip.authenticate.allow** プロパティで定義されたより広い ServiceNow 内部 IP 範囲を使用して、これらの内部 ServiceNow 受信接続が許可されます。

自動翻訳

glide.ip.authenticate.allow.secured システムプロパティに信頼できる値のみが含まれていて、プロパティ **glide.ip.authenticate.strict** が **true** に設定されていることを確認してください。

⚠ 警告: このプロパティの値は、DB オーバーライドなしです。変更またはオーバーライドすることはできません。

詳細情報

属性	説明
プロパティ名	*glide.ip.authenticate.strict*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	アーキテクチャ、設計、および脅威のモデル化
目的	ServiceNow の従業員が、保護された IP 範囲を介してのみインスタンスにアクセスできるようにすること
推奨値	true
セキュリティリスク評価	4.3
機能への影響度	(低) このプロパティが有効になっていない場合、ServiceNow の従業員はすべての IP 範囲を介して顧客のインスタンスにアクセスできます。このプロパティを有効にすると、アクセスが保護された IP 範囲 (セキュア VPN、DC) セットに制限されます。 ℹ 注: このプロパティを true に設定すると、ServiceNow AI Platform は、インスタンスにアクセスできる IP 範囲セットに対して、パフォーマンス監視 IP 制限 (*glide.ip.authenticate.allow.secured*) プロパティの代わりに、より制限の強い *glide.ip.authenticate.allow.secured* プロパティを使用します。
セキュリティリスク	(低) より幅広いユーザーグループに対してインスタンスアクセスが不必要に公開されます。
参照	IP 範囲ベースの認証

ℹ 注: IP アクセス制御に追加されていない IP からのアクセスをすべて制限するには、「すべて拒否」ルールを IP アクセス制御に追加する必要があります。その後、必要な許可された IP をすべて IP アクセス制御に追加する必要があります。

レガシー **JQuery** 動作を無効にする (セキュリティセンター **1.3** で更新)

glide.jquery.legacy は、古いパッチが事前に適用されたバージョンの JQuery が使用され、ライブラリにパッチ未適用の脆弱性が生じるのを防ぐために使用されます。

古いパッチが事前に適用されたバージョンの JQuery が使用され、ライブラリにパッチ未適用の脆弱性が生じるのを防ぐには、*glide.jquery.legacy* を **false** に設定します。パッチが事前に適用されたバージョンの JQuery を許可するには、値を **true** に設定します。

自動翻訳

詳細情報

属性	説明
プロパティ名	*glide.jquery.legacy*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	アーキテクチャ、設計、および脅威のモデル化
目的	古いバージョンの AngularJS ライブラリで発見された脆弱性に対する攻撃から発生する潜在的なセキュリティリスクを防止します。
推奨値	False
構成タイプ	ブーリアン
セキュリティリスク	(高) ライブラリにパッチ未適用の脆弱性が生じる、古いパッチが事前に適用された JQuery バージョンの使用を防止します。システムプロパティは、パッチ適用されていないバージョンの angularJS に依存してカスタム実装を実行する場合はフェイルセーフです。
セキュリティリスク評価	7.1

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

GlideRecord スコープフェンシングの従来の動作を無効にする (セキュリティセンター **1.3** の新機能、1.5 および **2.0** で更新)

glide.record.legacy_cross_scope_access_policy_in_script プロパティは、スコープフェンシングを無効にし、スコープ対象のアプリがグローバルスクリプトインターフェイスにアクセスできるようにします。これは、GlideRecord のクロススコープアクセスに対するパッチとして作成されました。

GlideRecord は、そのレベルのアクセス権で構成されていないテーブルへのクロススコープ作成/更新アクセス権を提供しました。このスコープ指定のアクセス動作にパッチが適用されたときに、顧客のアプリケーションが壊れないようにするために、プロパティ *glide.record.legacy_cross_scope_access_policy_in_script* が作成されました。true の場合、クロススコープアクセスは従来の動作にフォールバックします (安全でない)。このプロパティは、スコープフェンシングを無効にし、スコープ対象アプリがグローバルスクリプトインターフェイスにアクセスできるようにします。

セキュリティのベストプラクティスは、スコープフェンシングの制限を設けることです。スコーピングにより、アプリケーションは最小特権の原則に従って、明示的なアクセス権を持つリソースまたはスコープ内のリソースにのみアクセスできます。この機能を無効にすると、機密性、可用性、および整合性に影響を与える可能性があります。

Glide プロパティ *glide.record.legacy_cross_scope_access_policy_in_script* を false に設定します。sys_propertiesテーブルに存在しない場合、デフォルト値は true です。

詳細情報

属性	説明
構成名	*glide.record.legacy_cross_scope_access_policy_in_script*

属性	説明
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	false
デフォルト値	true (sys_properties テーブルにプロパティが存在しない場合)
カテゴリ	アーキテクチャ、設計、および脅威のモデル化
セキュリティリスク	• 重大度スコア：5 • CVSS スコア：中 • セキュリティリスクの詳細：スコーピングにより、アプリケーションは、明示的なアクセス権を持つリソースにのみ、または最小特権の原則に従ってスコープ内でのみアクセスできます。この機能を無効にすると、機密性、可用性、および整合性に影響を与える可能性があります。
依存関係と前提条件	なし

レガシー **AngularJS** 動作を無効にする **(Security Center 1.3** で更新)

glide.angular.legacy プロパティを使用して、古いバージョンの AngularJS ライブラリで発見された脆弱性に対する攻撃から発生する潜在的なセキュリティリスクから保護します。

glide.angular.legacy を推奨値の **false** に設定すると、古いパッチが事前に適用されたバージョンの angularJS が使用されなくなります。古いパッチが事前に適用されたバージョンの angularJS を使用するには、 プロパティを **true** に設定します。

詳細情報

属性	説明
プロパティ名	*glide.angular.legacy*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	アーキテクチャ、設計、および脅威のモデル化
目的	システムプロパティは、パッチ適用されていないバージョンの angularJS に依存してカスタム実装を実行する場合はフェイルセーフです。
推奨値	False
構成タイプ	ブーリアン
セキュリティリスク	(高) 古いバージョンの angularJS を使用すると、古いバージョンの AngularJS ライブラリで発見された脆弱性に対する攻撃から発生する潜在的なセキュリティリスクが生じる可能性があります。
セキュリティリスク評価	7.1

データブローカー **REST API** に認証を必須とする **(Security Center 1.3** で更新)

glide.basicauth.required.databrokerrestapiprocessor プロパティを使用して、すべての受信データブローカー REST API 要求に対して基本認証を要求します。

このプロパティが **true** に設定されている場合、認証が適用されます。**false** に設定されている場合は、認証が使用されないため、機密情報がインスタンスから漏洩する可能性があります。

⚠ **警告:** これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

詳細情報

属性	説明
構成名	*glide.basicauth.required.databrokerrestapiprocessor*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	アーキテクチャ、設計、および脅威のモデル化
セキュリティリスク	• 重大度スコア：8.6 • CVSS スコア：高 • セキュリティリスクの詳細：プロパティが **false** に設定されている場合、API 認証は適用されず、攻撃者が機密データにアクセスできるようになります。
依存関係と前提条件	なし

ベアラー承認の場合の制限付きバインディング機能 **(Security Center 7.0** の新機能)

システムプロパティと制限付きバインディングを使用して、そのエンティティを使用して生成されたアクセストークンを UI 呼び出しに使用できないようにします。

glide.oauth.enforce_restricted_binding_for_ui システムプロパティを使用して OAuth エンティティの制限付きバインディングを有効にし、そのエンティティによって生成されたアクセストークンが UI 呼び出し (incident_list.do など) に使用されないようにします。

制限付きバインディングがオフになっている場合、生成されたアクセストークンは、システムプロパティの値に関係なく UI 呼び出しに使用できます。

すべての OAuth エンティティエントリで *glide.oauth.enforce_restricted_binding_for_ui* が **true** に設定され、 [トークン制限の適用] が **true** に設定されていることを確認します。OAuth エンティティエントリの詳細については、「 OAuth 受信」を参照してください。

詳細情報

属性	説明
構成名	*glide.oauth.enforce_restricted_binding_for_ui*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	false
フォールバック値	false
カテゴリ	アーキテクチャ、設計、および脅威のモデル化
セキュリティリスク	• 重大度スコア:5.0 • CVSS スコア：中 • セキュリティリスクの詳細: *glide.oauth.enforce_restricted_binding_for_ui* システムプロパティが **true** に設定されていないか、制限付きバインディングがオフになっている場合、API (モバイル API など) にアクセスするためのアクセストークンを持つユーザーは、発行されたセッションを取得し、それを使用してインスタンス上の他の制限付きリソース (incident_list.do など) にアクセスできます。
依存関係と前提条件	なし

空の **ACL** でデフォルトで拒否する **(Security Center 1.3** で更新)

*glide.sm.default_mode*プロパティを使用して、既存のアクセス制御リスト (ACL) ルールがワイルドカードテーブルの ACL ルールの一部であることを検出した場合のセキュリティマネージャーのデフォルトの動作を制御します。

リソースに対して ACL が定義されていない場合、またはワイルドカードのテーブルレベルの ACL しかない場合 (incident.* など) に、インスタンスの従来のセキュリティマネージャーがリソースへのアクセスを許可しないようにします。デフォルトでアクセスが許可されている場合、明示的な ACL が設定されていないものはすべて操作の影響を受ける可能性があります。

定義された ACL ルールがない場合、またはワイルドカードのテーブルレベル ACL のみが存在する場合は、 **glide.sm.default_mode** システムプロパティ値を [拒否] に設定してアクセスを禁止します。

⚠ **警告:** これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

詳細情報

属性	説明
プロパティ名	*glide.sm.default_mode*
構成タイプ	システムプロパティ (/sys_properties_list.do)

属性	説明
カテゴリ	アーキテクチャ、設計、および脅威のモデル化
目的	セキュリティ上のベストプラクティスは、無許可のユーザーによるテーブルへのアクセスを制限することです。 • テーブルに ACL ルールがない場合、このプロパティは、テーブル/フィールドで実行される CRUD 操作に対して少なくともワイルドカード ACL が検証されるようにします。 • これらのルールでは、ユーザーが admin ロールを所有しているか、別のテーブル ACL ルールの要件を満たしている場合を除き、読み取り、書き込み、作成、削除の操作をすべてのテーブルで制限します。
推奨値	拒否
機能への影響	このプロパティを [許可] に設定すると、ワイルドカードテーブル ACL ルールは、そのような操作を制限する特定のテーブル ACL ルールが存在する場合を除き、すべてのテーブルで CRUD 操作を許可します。 **注:** このプラグインは、本番環境で既に使用されているテーブルへのセキュリティアクセスを変更する可能性があるため、既存のインスタンスを対象としていません。
セキュリティリスク	6.3
参照	デフォルトの拒否プロパティ

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

トークン認証情報の自動トークンクリーンアップの設定 (Security Center 2.0 の新機能)

com.snc.platform.security.token.auth.cleanup プロパティを使用して、期限切れの API キーと HMAC シークレットを確実に削除することで、トークンの再利用の可能性を制限します。

com.snc.platform.security.token.auth.cleanup プロパティが安全でない値 false に設定されている場合、期限切れの API キーと HMAC シークレットは削除されず、トークンが再利用される可能性があります。トークンが漏洩または侵害により期限切れになった場合、そのトークンを再利用すると、漏洩したトークンを所有するすべてのユーザーにインスタンスが公開される可能性があります。

期限切れのトークンは、*com.snc.platform.security.token.auth.days.expired.hmac_secret.is.kept* と *com.snc.platform.security.token.auth.days.expired.api_key.is.kept* で定義された日数の間保持されます。これらの設定に有効な値は、0 以上の整数です。値が 0 の場合、期限切れのトークンはその日のうちに削除されますが、日数が長い場合は公開期間が長くなります。デフォルト値は 7 日以下にすることをお勧めします。

詳細情報

属性	説明
構成名	*com.snc.platform.security.token.auth.cleanup,* *com.snc.platform.security.token.auth.days.expired.hmac_se*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	整数
推奨値	推奨値は true で、7 以下の整数です。
デフォルト値	7
カテゴリ	アーキテクチャ、設計、および脅威のモデル化
セキュリティリスク	• 重大度スコア:5.1 • CVSS スコア：中 • セキュリティリスクの詳細:このプロパティを推奨値の true に構成しないと、期限切れの API キーと HMAC シークレットが削除されなくなり、トークンが再利用される可能性が高くなります。
依存関係と前提条件	なし

認証

認証カテゴリは、エンティティとその主張が本物で正確であり、偽装を防止し、パスワードの傍受を防止していることを確認するための最新の認証の主要な要素をカバーしています。

ASVS 標準は、このセクションの NIST 800-63b (https://pages.nist.gov/800-63-3/sp800-63b.html) 仕様に基づいてビルドされています。

認証には、パスワードポリシー、コントロールとストレージ、認証システムの適切な実装、帯域外または 1 回限りの検証機能の適切な実装が含まれます。

ロールベースのマルチファクター認証を有効にする (セキュリティセンター **1.3** で更新)

glide.authenticate.multifactor プロパティを使用して、特定のロールにアサインされたすべてのユーザーにロールベースのマルチファクター認証 (MFA) を適用します。

マルチファクター認証は、ユーザーに割り当てられたロールに基づいて適用します。マルチファクターロールリストで「admin」、「security_admin」または「user_admin」のロールがユーザーに割り当てられている場合は、MFA が適用されます。

- 特定のロールに割り当てられたすべてのユーザーに対してロールベースのマルチファクター認証を適用するには、このプロパティを **true** に設定します。
- 特定のロールに割り当てられたすべてのユーザーに対してロールベースのマルチファクター認証を無効にするには、このプロパティを **false** に設定します。

自動翻訳

詳細情報

属性	説明
プロパティ名	*glide.authenticate.multifactor*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	認証
目的	特定のロールに割り当てられたすべてのユーザーに対してロールベースのマルチファクター認証を適用する。
タイプ	True/False
推奨値	*true*
セキュリティ依存関係	[マルチファクター基準] テーブルで、[ロールベースのマルチファクター認証 **(Role-based multi-factor authentication)]** をアクティブ化します。
セキュリティリスク評価	7.2
機能への影響	このプロパティを有効にすると、ユーザーのエクスペリエンスが向上します。これは、侵害された認証情報に対する保護およびセキュリティの追加レイヤーとして機能します。
セキュリティリスク	(中) このプロパティを有効にしないと、機密データに不正にアクセスされるリスクがあります。
参照	ロールベースのマルチファクター基準の設定

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

ロールベースのマルチファクター認証を有効にする **(Security Center 1.3** で更新)

glide.authenticate.multifactor プロパティを使用して、特定のロールにアサインされたすべてのユーザーにロールベースのマルチファクター認証 (MFA) を適用します。

マルチファクター認証は、ユーザーに割り当てられたロールに基づいて適用します。マルチファクターロールリストで「admin」、「security_admin」または「user_admin」のロールがユーザーに割り当てられている場合は、MFA が適用されます。

- 特定のロールに割り当てられたすべてのユーザーに対してロールベースのマルチファクター認証を適用するには、このプロパティを **true** に設定します。
- 特定のロールに割り当てられたすべてのユーザーに対してロールベースのマルチファクター認証を無効にするには、このプロパティを **false** に設定します。

詳細情報

属性	説明
プロパティ名	*glide.authenticate.multifactor*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	認証

属性	説明
目的	特定のロールに割り当てられたすべてのユーザーに対してロールベースのマルチファクター認証を適用する。
タイプ	ブール
推奨値	*true*
デフォルト値	false
セキュリティ依存関係	[マルチファクター基準] テーブルで、[ロールベースのマルチファクター認証 **(Role-based multi-factor authentication)]** をアクティブ化します。
セキュリティリスク評価	7.2
機能への影響	このプロパティを有効にすると、ユーザーのエクスペリエンスが向上します。これは、侵害された認証情報に対する保護およびセキュリティの追加レイヤーとして機能します。
セキュリティリスク	(中) このプロパティを有効にしないと、機密データに不正にアクセスされるリスクがあります。
参照	ロールベースのマルチファクター基準の設定

無効なパスワードリセット試行に対するロックアウト時間を制御する **(Security Center 1.3** および **2.0** で更新)

password_reset.request.max_attempt_window プロパティは、*password_reset.request.max_attempt* プロパティで設定された最大試行失敗回数を超えた場合に、ユーザーがパスワードのリセットまたは変更を待機する必要がある時間 (分数) を制御します。

password_reset.request.max_attempt_window プロパティは、*password_reset.request.max_attempt property* で設定された最大試行失敗回数を超えた場合に、ユーザーがパスワードのリセットまたは変更のために待機しなければならない分数を定義します。*password_reset.request.max_attempt_window*プロパティの時間が短いと、パスワードリセットの試行回数が増えるため、パスワードの総当たり攻撃に成功するリスクが高まります。デフォルトの 1440 分をお勧めします。

プロパティ *password_reset.request.max_attempt_window* が 1440 以上に設定されていることを確認します。

詳細情報

属性	説明
プロパティ名	*password_reset.request.max_attempt_window*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	認証
目的	パスワードリセットの最大試行失敗回数に達した後のロックアウト期間 (分数) を示します。
推奨値	1440
デフォルト値	1440

属性	説明
構成タイプ	正の整数値
セキュリティリスク	(高) プロパティが 1440 以下の推奨値に設定されていない場合、誤った認証試行の最大回数に達してもアカウントがロックされないため、アカウントの総当たりを実行できる可能性があります。
セキュリティリスク評価	7.5
参照	パスワードリセットプロパティの設定

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

受信メールからユーザーを作成するのを無効化する (セキュリティセンター 1.3 で更新)

glide.user.trusted_domain プロパティを使用して、受信メールからユーザーを作成する際に信頼できるドメインのカンマ区切りリストを指定します。

アドミニストレーターは受信メールからユーザーを自動的に作成するようにメール プロパティを設定できます。このプロパティを安全でない値に設定すると、インスタンスは受信メールから自動的にユーザーを作成します。作成された各ユーザーには、ハードコードされた同じデフォルトのパスワードが指定されるため、ブルートフォースによる認証のバイパスが容易になります。

詳細情報

属性	説明
プロパティ名	*glide.pop3readerjob.create_caller*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	認証
推奨値	false
デフォルト値	false
セキュリティリスク評価	5.4
機能への影響	このプロパティが設定されると、インスタンスは信頼できるドメインからのメールのみを受け入れます。信頼できるリストにドメインを含めない場合、アカウントが自動的に作成されるため、ゲストユーザーに影響があります。
セキュリティリスク	(中) このプロパティが有効になっていない場合、攻撃者がなりすまし/スパムキャンペーンを使用して複数のメールを送信した結果、不要なゲストユーザーをさらに作成する可能性があります。
参照	受信メール設定

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

パスワードなしの認証の無効化

glide.login.no_blank_password プロパティを使用すると、ユーザーが NOW プラットフォームに空白のパスワードでログインしたり、 [パスワード] フィールドを空のままにしたりできなくなります。

アドミニストレーターが意図的に空の値または空白のパスワードをユーザーレコードに割り当てた場合でも、ユーザーは [パスワード] フィールドに値を指定しなければログインできなくなります。

詳細情報

属性	説明
プロパティ名	*glide.login.no_blank_password*
構成タイプ	システムプロパティ (/sys_properties_list.do)
インスタンスセキュリティセンターでの構成	あり
目的	組織内ではユーザー名が簡単に推測できることがあるため、強力な認証を確実に実現すること
推奨値	true
機能への影響	空白のパスワードは重大なセキュリティリスクと見なされるため、操作では使用しないでください。ただし、このような使用法に有効なケースがある場合は、機能停止する可能性があります。パスワードが空白のユーザーは、インスタンスにログインできません。
セキュリティリスク	(高) 攻撃者は、デフォルトのユーザー名を使用して、またはパスワードのない特定の個人/グループ (通常は firstname.lastname) を使用してインスタンスにログインできます。この操作は、公開ユーザーがインスタンスデータの機密性と完全性を侵害する可能性があるため、重大なセキュリティリスクと見なされます。

OAuth 2 トークン付与でリソース所有者のパスワード認証情報 (ROPC) を無効にする (Security Center 7.0 の新機能)

リソース所有者のパスワード認証情報 (ROPC) が OAuth 2 トークンを付与しないようにします。

デフォルトでは、クライアントアプリケーションがユーザー名とパスワードを使用してアクセストークンを直接要求するときに、リソース所有者のパスワード認証情報 (ROPC) でインスタンスに OAuth 2 トークンを付与できます。*glide.oauth.inbound.ropc.grant_type.disabled* が **true** に設定されている場合、ROPC は非アクティブになり、OAuth 2 トークンを付与するために使用することはできません。

glide.oauth.inbound.ropc.grant_type.disabled システムプロパティが **true** に設定されていることを確認します。システムプロパティ [sys_properties] テーブルにプロパティが存在しない場合、デフォルト値は **false** です。このプロパティがそのテーブルに存在する場合、デフォルトは **false** です。

詳細情報

属性	説明
構成名	*glide.oauth.inbound.ropc.grant_type.disabled*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	false
フォールバック値	false
カテゴリ	認証
セキュリティリスク	• 重大度スコア:3.3 • CVSS スコア：低 • プロパティが **false** に設定されている場合、ROPC を使用して OAuth 2 トークンを付与することは許可されます。ROPC は、ユーザーの認証情報がアプリケーションに公開されるため、他の認証フローよりも安全性が低いと見なされます。これにより、クライアントが侵害され、ベーシック認証と同様の弱点に悩まされる状況で脆弱性が発生する可能性があります。OAuth 2.1 は ROPC を廃止しました。
機能への影響	このプロパティが **true** に設定されている場合、ROPC は非アクティブになり、OAuth 2 トークンの付与に使用することはできません。これにより、ROPC を使用して OAuth 2 トークンを付与することでプラットフォームにアクセスするアプリケーションが防止されます。
依存関係と前提条件	OAuth 2.0 (com.snc.platform.security.oauth) プラグインがアクティブである必要があります。

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

ログイン時にパスワードポリシーを適用しない (セキュリティセンター **1.5** で更新、2.0 で削除)

インスタンスでパスワードの複雑さの処理方法を管理します。

プロパティ *glide.apply.password_policy.on_login* を false に設定すると、ログイン時にパスワードの複雑さが強制されなくなります。プロパティを true に設定すると、パスワードの複雑さが強制され、組織のポリシーコンプライアンスの問題が発生します。

ASVS 4.03 v2.1.9の推奨事項によると:

「許可される文字の種類を制限するパスワード作成ルールがないことを確認してください。大文字、小文字、数字、特殊文字を必須とするべきではありません。(C6)"

ASVS の推奨事項では、パスワードの複雑さを強制する代わりに、パスワードの長さに最低 12 文字を強制します。

「OWASP ASVS v4.0 認証」を参照してください。

詳細情報

属性	説明
構成名	*glide.apply.password_policy.on_login*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	false
デフォルト値	false
カテゴリ	認証
セキュリティリスク	• 重大度スコア：4.4 • CVSS スコア：中 • セキュリティリスクの詳細：このプロパティを **true** に設定すると、パスワードの複雑さが強制され、組織のコンプライアンスの問題が発生する可能性があります。
依存関係と前提条件	なし

アカウント復旧の有効化 (セキュリティセンター **1.3** および **1.5** で更新)

glide.sso.acr.enabled プロパティは、アカウント復旧機能を制御します。

glide.sso.acr.enabled を推奨値 **true** に設定すると、ユーザー ID によるアカウント復旧が可能になります。ユーザー ID によるアカウント復旧を禁止するには、値を **false** に設定します。

詳細情報

属性	説明
プロパティ名	*glide.sso.acr.enabled*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	認証
目的	ユーザー ID 機能によってアカウント復旧機能を制御します。
推奨値	True (デフォルト)
構成タイプ	ブーリアン
セキュリティリスク	重大 (このプロパティを有効にしないと、ユーザーはユーザー ID でアカウントを復旧できません)。
セキュリティリスク評価	9.1
参照	詳細については、「アカウント復旧 (ACR)」を参照してください。

顧客登録の CAPTCHA を有効にする

顧客登録時に CAPTCHA を有効にすることで、悪意のあるボットによる要求のリスクを軽減します。

sn_customerservice.captchaEnabled システムプロパティを使用して、カスタマーサービス管理 (CSM) ポータルでの顧客登録に対して CAPTCHA 検証を有効にするかどうかを決定します。CAPTCHA 検証を使用して、悪意のあるボットがアプリケーションに対して要求を自動的に送信するのを防ぎます。

CAPTCHA 検証を有効にするには、システムプロパティ **sn_customerservice.captchaEnabled** を **true** に設定します。プロパティがシステムプロパティ [sys_properties] テーブルにない場合、デフォルト値は **true** です。

詳細情報

属性	説明
技術的な構成名	sn_customerservice.captchaEnabled
プラグインの適用性	カスタマーサービス管理 (CSM)
セキュリティリスク	CAPTCHA 検証は、悪意のあるボットがアプリケーションに対して要求を自動的に送信するのを防ぐのに役立ちます。
共通脆弱性スコアリングシステム (CVSS) スコア	3.7
共通脆弱性スコアリングシステム (CVSS) 評価	低
機能への影響	ユーザーの登録は、CAPTCHA 検証に合格する必要があるため、望ましくないエクスペリエンスになる可能性があります。
依存関係と前提条件	なし
データタイプ	ブール
ベースシステム値	正しい
フォールバック値	正しい
推奨値	正しい

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

拒否リストに含まれているパスワードの検証チェックを有効にする

[除外されたパスワード] テーブルで拒否リストに含まれているパスワードを管理します。

glide.enable.blacklist_password プロパティを使用して、拒否リストに含まれているパスワードを監視します。プロパティが **True** に設定されている場合、ユーザーのパスワードは拒否リストに含まれているパスワードのリストと照合され、ユーザーは侵害されたパスワードのセットにあるパスワードを使用できなくなります。アドミニストレーターは、[除外されたパスワード] テーブルにパスワードを挿入することでリストを管理できます。

詳細情報

属性	説明
構成名	*glide.enable.blacklist_password*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	false
カテゴリ	認証
依存関係と前提条件	なし
参照	インスタンスのパスワードポリシーによるパスワードの除外

外部ユーザー登録向けに **Captcha** を有効にする **(Security Center 1.3** および **1.5** で更新)

sn_ext_usr_reg.captchaEnabled は、外部ユーザー登録のために CAPTCHA を検証するかどうかを制御します。

sn_ext_usr_reg.captchaEnabled を推奨値の **true** に設定すると、外部ユーザー登録で CAPTCHA を要求することで自動アカウント作成攻撃を防ぐことができます。外部ユーザー登録に CAPTCHA を要求しない場合は、値を **false** に設定します。

詳細情報

属性	説明
プロパティ名	*sn_ext_usr_reg.captchaEnabled*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	認証
目的	このプロパティは、CSP、Community などのポータルで外部ユーザー登録を実行するときに CAPTCHA 検証を有効または無効にするために使用されます。これは、ワクチン接種アドミニストレーション管理 (VAM) や CSM Guest Walkup などのストアアプリでも CAPTCHA を有効または無効にするために使用されます。
推奨値	true
構成タイプ	ブーリアン
セキュリティリスク	(低) このプロパティは、外部ユーザー登録での CAPTCHA の有効化を制御します。理想的でない値を指定すると、セキュリティの脆弱性をもたらす可能性があります。
セキュリティリスク評価	3.7

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

パスワードリセットで **CAPTCHA** を有効にする

password_reset.captcha.ignore プロパティを使用して、ユーザーがパスワードをリセットするときに CAPTCHA チャレンジの要求を有効または無効にします。

ユーザーがパスワードをリセットする際に CAPTCHA チャレンジを要求するには、*password_reset.captcha.ignore* を推奨値の **false** に設定します。パスワードリセットの CAPTCHA オプションを無視するには、値を **true** に設定します。

CAPTCHA は、自動化されたシステムでは簡単に回答できないチャレンジ応答をユーザーに表示することで、自動化攻撃を防止するために役立ちます。CAPTCHA を無効にすると、パスワードリセット機能に対する自動攻撃が成功する可能性が高くなります。

注: このプロパティは、パスワードリセットの自動化にのみ使用されます。

詳細情報

属性	説明
プロパティ名	*password_reset.captcha.ignore*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	認証
目的	このプロパティは、パスワードリセット時の CAPTCHA 検証を有効または無効にするために使用されます。
推奨値	false
構成タイプ	ブーリアン
セキュリティリスク	(中) 理想的でない値を指定すると、セキュリティの脆弱性をもたらす可能性があります。
セキュリティリスク評価	5.5

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

マルチファクター認証のメール **OTP** の有効化

インスタンスで 2 要素認証の適用方法を管理します。

glide.authenticate.multifactor.email.otp.enabled プロパティを使用して、2 要素認証のトークンをメールを使用して送信するかどうかを制御します。メールは脆弱な MFA 要素と見なされており、MFA をバイパスするために攻撃者がアクセス権を取得する可能性があります。このプロパティを **false** に設定すると、攻撃者がユーザーのパスワードを侵害したときに MFA をバイパスするリスクが軽減されます。

詳細情報

属性	説明
構成名	*glide.authenticate.multifactor.email.otp.enabled*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン

属性	説明
推奨値	false
デフォルト値	true
カテゴリ	認証
セキュリティリスク	• 重大度スコア：3.1 • CVSS スコア：低 • セキュリティリスクの詳細：このプロパティを **false** に設定すると、攻撃者が 2 要素認証をバイパスするリスクが軽減されます。
依存関係と前提条件	なし
参照	MFA 要素としてのメール

パスワードリセットポリシーチェックを有効にする (Security Center 2.0 で更新)

glide.enable.password_policy プロパティを使用して、ユーザーがユーザーインターフェイスを使用してパスワードを変更するたびに、パスワードポリシーのチェックを有効にします。

このプロパティを有効にした後に使用するパスワードポリシーを定義するには、「インスタンスでのパスワードポリシーの有効化」を参照してください。Glide プロパティ *glide.enable.password_policy* が存在し、値が true に設定されていることを確認します。プロパティがsys_propertiesテーブルに含まれていない場合は、新しいレコードを追加してください。

注: **glide.enable.password_policy** は、アドミニストレーターがパスワードを変更する場合や、スクリプトを使用してユーザーを追加する場合には適用されません。

警告: これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

詳細情報

属性	説明
プロパティ名	*glide.enable.password_policy*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	認証
目的	パスワード変更時にパスワードポリシーを適用すること
推奨値	true (より強力なパスワードの場合)
セキュリティリスク評価	7.4
機能への影響	プロパティを true に設定すると、ユーザーがパスワードをリセットするときにパスワードポリシーチェックがオンになります。
セキュリティリスク	(中) パスワードポリシーがない場合、ユーザーが脆弱なパスワードを作成するため、攻撃者がインスタンスにアクセスする可能性が高くなります。

設定手順

インスタンスセキュリティセンターの [ハードニングコンプライアンス構成] ページでこの設定を構成する場合は、次の手順を実行します。

1. [中] で、[セッション管理] を選択します。
2. [パスワードリセットポリシーチェックの有効化 **(Enable Password Reset Policy Checks)**] 設定で、パスワードの強度を中程度にする場合は [中] を、より強力なパスワードにする場合は [強 **(strong)**] を選択します。これらのオプションのいずれかを選択すると、**glide.enable.password_policy** プロパティが true に設定され、パスワードポリシーを自動的に更新するワークフローが開始されます。

さらに、*glide.apply.password_policy.on_login* システムプロパティを設定して、ログイン時のパスワードポリシーチェックを有効にすることもできます。

モバイルのポリシーベースのセッションアクセスを有効にする **(Security Center 1.5** の新機能)

Zero Trust - Policy Based Session Access プラグインを使用して、モバイルアプリを通じて認証するユーザーのロールを削減するかどうかを制御します。

Zero Trust - Policy Based Session Access プラグインを使用すると、セキュリティアドミンが、適応認証ポリシーを使用して、IP アドレス、場所、ID プロバイダー属性、ユーザー属性などのパラメーターに基づいて、セッション内のユーザーアクセスを削減できます。このプラグインが有効になっているか true に設定されている場合、モバイルデバイスを介して認証するユーザーのロールは、プラグインのポリシーに従って制限されます。ユーザーがモバイルデバイスを介して認証する場合、機密性の高い操作に関して安全でない環境であることを示す可能性があるため、インスタンスアドミンは、高い特権アクセスを制限したい場合があります。

詳細情報

属性	説明
構成名	*glide.authenticate.session_access.mobile.enabled*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	アクセス制御
セキュリティリスク	• 重大度スコア：4.7 • CVSS スコア：中 • セキュリティリスクの詳細：このハードニング設定が true に設定されている場合、ポリシーベースのセッションアクセスがモバイルログインのインスタンスに強制され、信頼できる環境からアクセスしていないユーザー、または信頼できるデバイスを使用していないユーザーのロールの特権が削減されます。true は安全な設定です。この設定が false に構成されている場合、ポリシーベー

属性	説明
	スのセッションアクセスは無効になり、ユーザーは常に、アドミンなどの高い特権ロールを含む完全なロールを持ち続けます。
依存関係と前提条件	なし
機能への影響度	アドミンがインスタンスにセッションアクセスポリシーを構成している場合、ユーザーが信頼できる環境からアクセスしていないか、信頼できるデバイスを使用していない場合、モバイルログイン後にユーザーのロールが減らされます。
参照	適応認証

SAML 要求のリレーステータスを有効にして、リプレイ攻撃を防止する

SAML 要求のリレーステータスを有効にしてリプレイ攻撃を防ぐことで、リプレイ攻撃のリスクを軽減します。

glide.authenticate.sso.saml2.enable_relay_state_with_id システムプロパティを使用して SAML リプレイ攻撃から保護します。このプロパティが **true** に設定されている場合、リレーステータスパラメーターには、リレーステータス URL のリダイレクト先であるマルチ SSO 要求パラメーター [multisso_request_parameter] テーブルのレコードsys_idが含まれます。

システムプロパティ **[glide.authenticate.sso.saml2.enable_relay_state_with_id**] を **true** に設定します。これにより、SAML 要求にアクセスした攻撃者が有効な要求を再送信して、インスタンスにアクセスするのを防ぐことができます。

詳細情報

属性	説明
技術的な構成名	glide.authenticate.sso.saml2.enable_relay_state_with_id
プラグインの適用性	Multi-Provider SSO プラグイン (com.snc.integration.sso.multi.installer)
セキュリティリスク	このシステムプロパティによって有効化されるリレーステータスは、リプレイ攻撃からインスタンスを保護するのに役立ちます。このプロパティを有効にすると、SAML 要求にアクセスした攻撃者が有効な要求を再送信してインスタンスにアクセスするのを防ぐことができます。
共通脆弱性スコアリングシステム (CVSS) スコア	3.8
共通脆弱性スコアリングシステム (CVSS) 評価	低
機能への影響	このプロパティが true に設定されている場合、SAML 要求のリレーステータスには、リダイレクト先のリレーステータス URL を含むマルチ SSO 要求パラメーター [multisso_request_parameter] テーブルのレコードsys_idが含まれます。
依存関係と前提条件	なし

属性	説明
データタイプ	ブール
ベースシステム値	true
フォールバック値	false
推奨値	true

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

登録および検証の際の SMS コード通知を有効化する (セキュリティセンター 1.3 で更新)

password_reset.sms.use_notify プロパティは、パスワードリセット用の SMS コード通知の使用を制御します。

password_reset.sms.use_notify プロパティが推奨値の **true** に設定されている場合、SMS 検証とメールよりも安全な新しいデバイス登録を使用して、パスワードリセットをするようにユーザーに通知されます。

詳細情報

属性	説明
構成名	*password_reset.sms.use_notify*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	認証
セキュリティリスク	• 重大度スコア：3.7 • CVSS スコア：低 • セキュリティリスクの詳細：このプロパティを **false** に設定すると、SMS よりも安全性が低いメールがパスワード回復のデフォルト方法となります。
依存関係と前提条件	なし

LDAP 認証で SSL を有効にする (Security Center 1.5 および 2.0 で更新)

インスタンスで LDAP 認証要求の暗号化を管理します。

glide.ldap.use.ssl プロパティを使用して、ネットワーク経由で送信される LDAP 認証要求の TLS 暗号化を有効または無効にします。このプロパティが推奨値の **true** に設定されていない場合、LDAP 認証は中間者攻撃の影響を受けやすくなります。

詳細情報

属性	説明
構成名	*glide.ldap.use.ssl*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	認証
セキュリティリスク	• 重大度スコア：8.1 • CVSS スコア：高 • セキュリティリスクの詳細：このプロパティを **false** に設定すると、LDAP 認証は中間者攻撃に対して脆弱になります。
依存関係と前提条件	なし
参照	ライトウェイトディレクトリアクセスプロトコル統合

ログイン時に現在のパスワードポリシーコンプライアンス要件を強制する

ログイン時に現在のパスワードポリシーコンプライアンス要件を適用することで、総当たりでアカウントログインのリスクを軽減します。

インタラクティブユーザーが **glide.apply.password_policy.on_login** システムプロパティを使用して、現在のアドミニストレーター要件を満たしていないパスワードでインスタンスにログインすることを防ぎます。

現在のパスワードポリシーコンプライアンス要件を適用するには、**glide.apply.password_policy.on_login** システムプロパティを **true** に設定します。このプロパティがシステムプロパティ [sys_properties] テーブルに存在しない場合、デフォルト値は **false** です。

詳細情報

属性	説明
技術的な構成名	glide.apply.password_policy.on_login
プラグインの適用性	なし
セキュリティリスク	インタラクティブユーザーは、現在のアドミニストレーター要件を満たしていないパスワードを使用してインスタンスに引き続きログインする可能性があります。これは、ユーザーが最新のセキュリティ要件を満たしていない脆弱なパスワードを使用していることを意味し、悪意のあるユーザーによるブルートフォースアカウントログインのリスクが高まる可能性があります。
共通脆弱性スコアリングシステム (CVSS) スコア	4.4

属性	説明
共通脆弱性スコアリングシステム (CVSS) 評価	中
機能への影響	既存のパスワードが現在のパスワードポリシーに準拠していない場合、このプロパティを有効にすると、ユーザーは次回ログイン時にパスワードの変更を強制されます。このプロパティは自動的に **false** に設定されます。値を **true** に設定すると、ログイン時にパスワードポリシーが適用されます。 **注:** このプロパティを有効にすると、新しいパスワードポリシーに準拠していないかなりの数のユーザーがパスワードの変更を強制される可能性があります。
依存関係と前提条件	なし
データタイプ	ブール
ベースシステム値	false
フォールバック値	false
推奨値	true

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

デバイスの暗号化とパスコードの要件を強制する (Security Center 1.3 の新機能)

glide.sg.device_encryption_enabled プロパティは、連邦情報処理標準 (FIPS 140-2) 暗号化を強制します。モバイルデバイスの暗号化とパスコードは、デバイスが物理的に取得された場合でも、権限のないユーザーがデバイスのコンテンツにアクセスできないようにします。

glide.sg.device_encryption_enabled が true に設定されている場合、ServiceNow モバイルアプリは、デバイスの暗号化とデバイスのパスコードが有効になっていることを確認します。

詳細情報

属性	説明
構成名	*glide.sg.device_encryption_enabled*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	false
カテゴリ	認証
セキュリティリスク	• 重大度スコア：4.2 • CVSS スコア：中 • セキュリティリスクの詳細：暗号化またはパスコードが有効になっていない場合、ユーザーはモバイルでインスタンスにログインできません。

属性	説明
依存関係と前提条件	なし
機能への影響	このプロパティが true に設定されている場合、モバイルアプリはデバイスの暗号化が有効になっているかどうかを確認します。暗号化が有効になっていない場合、ユーザーはモバイルで現在のインスタンスにログインできません。 ユーザーはログアウトし、次の警告メッセージが表示され、デバイス PIN を設定するか、デバイスを暗号化してログインを再試行するように提案されます。 You were logged out You need a passcode in order to use this instance on this device. Go to your device's settings to set one up

無効なパスワードリセットの試行回数を制限する (セキュリティセンター 1.3 で更新、2.0 で更新)

password_reset.request.max_attempt は、指定された期間ロックアウトされるまでにユーザーが実行できるパスワードのリセットまたは変更の最大試行失敗回数を制御するために使用されます。

詳細情報

属性	説明
プロパティ名	*password_reset.request.max_attempt*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	認証
目的	ユーザーがパスワードリセットプロセスからロックアウトされるまでに実行できる、パスワードリセットの最大試行失敗回数を示します。ロックアウト期間は *password_reset.request.max_attempt_window* の値によって決まります。
推奨値	3 未満の正の整数値に設定します。デフォルト値は **[3]** です。プロパティの上限を決定するときは、ユーザーが実行しているタスクを考慮してください。
構成タイプ	正の整数値
セキュリティリスク	(高) プロパティが推奨値の「**3**」またはその他の妥当な小さい値に設定されていない場合、パスワードリセットプロセスに対して総当たり攻撃が行われる可能性があります。
セキュリティリスク評価	7.5
参照	パスワードリセットプロパティの設定

ログイン失敗後のロック解除タイムアウトの管理 (Security Center 1.3 で更新)

ユーザーがパスワードの入力を何度間違えると ServiceNow AI Platform からロックアウトされるかをサイトアドミニストレーターが管理できるようにするために、2 つのスクリプトアクションが用

意されています。これらのスクリプトアクションのいずれかを有効にして、ログイン試行の失敗を管理できます。

詳細情報

属性	説明
プロパティ/プラグイン名	適用外
構成タイプ	[システムポリシー] > [スクリプトアクション]
カテゴリ	認証
目的	ログイン試行の失敗に対して厳格なポリシーを適用し、認証情報に対する総当たり攻撃を防止すること
推奨値	有効
セキュリティリスク評価	7.3
機能への影響	この修正により、インスタンスのアドミニストレーターは悪意のあるユーザーアクセスを監視および報告できるようになります。機能への影響はなく、ユーザーエクスペリエンスのみが変わります。
セキュリティリスク	(中) 定義されたログ記録と監査戦略を適用して、不審なアクティビティを適切なタイミングで特定して対処できるようにします。

設定手順

1. 移動先 システムポリシー > スクリプトアクション。
2. 名前「*`SNC User`*」を検索します。
3. 失敗したログイン試行を管理できるようにするには、*`SNC User Lockout Check with Auto Unlock`* または *`SNC User Lockout Check`* スクリプトアクションのいずれかの [アクティブ] ステータスを **false** から **true** に変更します。
4. ログイン成功後にログイン失敗回数のカウンターをリセットするには、*`SNC User Clear`* スクリプトアクションをアクティブ化します。

失敗したログインのロック解除タイムアウト期間を最大化する (セキュリティセンター **1.3** で更新)

ユーザーがパスワードの入力を何度間違えると ServiceNow AI Platform からロックアウトされるかをサイトアドミニストレーターが管理できるようにするために、2 つのスクリプトアクションが用意されています。これらのスクリプトアクションのいずれかを有効にして、ログイン試行の失敗を管理できます。

ロックアウトされた後にユーザーがログインできない期間を定義することで、総当たり攻撃からインスタンスを保護します。**glide.user.unlock_timeout_in_mins** システムプロパティは、その値で指定された期間後にユーザーアカウントのロックを解除します。値が指定されていない場合、インスタンスはデフォルトの 15 分後にユーザーアカウントのロックを解除します。

glide.user.unlock_timeout_in_minsシステムプロパティ値を最小の **15** に設定します。**glide.user.unlock_timeout_in_mins**存在しない場合、デフォルトのロックアウト時間は 15 分に設定されます。

SNC ユーザーロックアウトチェックと自動ロック解除スクリプトアクション (スクリプトアクション [sysevent_script_action] テーブルにあります) が存在し、アクティブであることを確認しま

す。**[SNC** ユーザーロックアウトチェック (自動ロック解除あり)] スクリプトアクションは、高セキュリティ設定 (com.glide.high_security) プラグインとともにインストールされます。

詳細情報

属性	説明
構成名	• glide.user.unlock_timeout_in_mins (システムプロパティ) • sysevent_script_action (スクリプトアクション)
構成タイプ	[システムポリシー] > [スクリプトアクション]
カテゴリ	認証
目的	ログイン試行の失敗に対して厳格なポリシーを適用し、認証情報に対する総当たり攻撃を防止すること
推奨値	• **glide.user.unlock_timeout_in_mins** システムプロパティの場合は **15** • **SNC** ユーザーロックアウトチェックと自動ロック解除スクリプトアクションに対してアクティブです。
機能への影響	この修正により、インスタンスのアドミニストレーターは悪意のあるユーザーアクセスを監視および報告できるようになります。機能への影響はなく、ユーザーエクスペリエンスのみが変わります。
セキュリティリスク	• 重大度スコア:6.8 • セキュリティリスクの詳細:プロパティが安全な値に設定されておらず、ロックアウト期間が有効になっていない場合は、より短い時間枠でアカウントログインの総当たり攻撃をすることが容易になります。これにより、悪意のあるユーザーが最終的にインスタンスへの不正アクセスを取得する可能性があります。インスタンスへの影響は、影響を受けるユーザーの権限に絞られ、総当たりでログインされます。

設定手順

1. 移動先 システムポリシー > スクリプトアクション。
2. 名前「*`*SNC User`*」を検索します。
3. 失敗したログイン試行を管理できるようにするには、*`SNC User Lockout Check with Auto Unlock`* または *`SNC User Lockout Check`* スクリプトアクションのいずれかの [アクティブ] ステータスを **false** から **true** に変更します。
4. ログイン成功後にログイン失敗回数のカウンターをリセットするには、*`SNC User Clear`* スクリプトアクションをアクティブ化します。

自動翻訳

パスワードリセット要求の再試行ウィンドウの持続期間を最大化する (セキュリティセンター **1.3** で更新)

password_reset.request.retry_window プロパティは、パスワードリセットの試行回数が更新されるまでの時間 (分数) を制御します。

詳細情報

属性	説明
プロパティ名	*password_reset.request.retry_window*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	認証
目的	再試行回数が 0 にリセットされるまでの、最後の要求からのパスワード試行回数の更新までの時間 (分数) を示します。
推奨値	**1440** 以上の正の整数値に設定します。デフォルト値は **1440** 分です。
構成タイプ	正の整数値
セキュリティリスク	(高) プロパティが推奨値の 1440 以上に設定されていない場合、パスワードリセットプロセスに対してアカウントの総当たり攻撃が行われる可能性があります。
セキュリティリスク評価	7.5
参照	パスワードリセットプロパティの設定

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

パスワードリセット要求のロック解除ウィンドウの持続期間を最大化する (セキュリティセンター **1.3** で更新)

password_reset.request.unlock_window プロパティは、最後にアカウントのロック解除アクションに成功した後でユーザーがリセット要求を開始するために待機しなければならない時間 (分数) を制御します。

このプロパティは、ユーザーが最後にアカウントのロック解除に成功した後、要求を開始するまでに待機する必要がある時間 (分) を制御します。*password_reset.request.unlock_window*が推奨値の 1440 以上に設定されていない場合、悪意のある攻撃者が自動ツールを使用してユーザーのパスワードの総当たり攻撃を行う機会が増えます。

詳細情報

属性	説明
プロパティ名	*password_reset.request.unlock_window*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	認証

属性	説明
目的	パスワードを正常にリセットした後で、ユーザーがパスワードを再度リセットするために待機しなければならない時間 (分数) を示します。
推奨値	1440
デフォルト値	1440
構成タイプ	正の整数値
セキュリティリスク	(高) プロパティが 1440 以上の推奨値に設定されていない場合、悪意のある攻撃者が自動ツールを使用してパスワードの総当たり攻撃を行う機会が増えます。
セキュリティリスク評価	5.9
参照	パスワードリセットプロパティの設定

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

パスワードリセット **SMS** の複雑さを最大化する (セキュリティセンター **1.3** で更新)

password_reset.sms.default_complexity プロパティは、パスワードのリセット時に必要な SMS コード検証の最小サイズを制御します。

詳細情報

属性	説明
プロパティ名	*password_reset.sms.default_complexity*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	認証
目的	パスワードのリセット時に必要な SMS コード検証サイズを制御します。
推奨値	6
デフォルト値	4
構成タイプ	0 より大きい整数値
セキュリティリスク	(低) プロパティが推奨値に設定されていない場合、弱い SMS 検証トークンが使用されます。これにより、アカウントの乗っ取りにつながる可能性があるトークン推測の可能性が高まります。
セキュリティリスク評価	3.8

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

パスワードリセット **SMS** の一時停止ウィンドウの持続期間を最大化する (セキュリティセンター **1.3** で更新)

ユーザーが新しいパスワードリセットコードを要求するまでに待機しなければならない時間を分単位で管理します。

このプロパティが推奨値の **2** 分以上に設定されていない場合、悪意のあるユーザーが短い時間に多くのパスワードリセットコードを開始する可能性があります。これにより、攻撃者が SMS リセットコードを予測できる可能性が高くなります。

詳細情報

属性	説明
構成名	*password_reset.sms.pause_window*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	整数
推奨値	2
デフォルト値	2
カテゴリ	認証
セキュリティリスク	• 重大度スコア：4.8 • CVSS スコア：中 • セキュリティリスクの詳細：*password_reset.sms.pause_window* が 2 以上の値に設定されていることを確認してください。
依存関係と前提条件	なし

パスワードリセット検証遅延期間を最大化する (セキュリティセンター **1.3** で更新)

新しいパスワードリセット要求を送信する前にユーザーが待機する必要がある遅延をミリ秒単位で設定します。

攻撃者は、 Bot などの自動化ツールを使用して、ログイン認証情報の総当たり攻撃を試みる可能性があります。[パスワードリセット検証遅延 **(Reset Password Verification Delay)]** プロパティはこの防御に役立ちます。このプロパティ値は、ユーザーがパスワードリセット要求を行うまでに待機しなければならない遅延をミリ秒単位で表します。このプロパティが推奨値の **1000** 以上に設定されていない場合、ログインは総当たり攻撃に対してより脆弱になります。

詳細情報

属性	説明
構成名	*password_reset.verification.delay*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	文字列
推奨値	1000
デフォルト値	1000
カテゴリ	認証

属性	説明
セキュリティリスク	• 重大度スコア：5.9 • CVSS スコア：中 • セキュリティリスクの詳細：プロパティ値を 1000 未満に設定すると、ログインが総当たり攻撃に対してより脆弱になります。
依存関係と前提条件	なし
参照	ユーザーのパスワードの設定

外部ユーザー登録リンクの有効期限を最小化する (セキュリティセンター **1.3** および **1.5** で更新)

登録リンクにアクセスできる日数を管理します。

sn_ext_usr_reg.Reg_link_expiration_days プロパティを使用して、登録リンクにアクセスできるユーザーを管理します。リンクが推奨値の **3** に設定されている場合、目的のユーザー以外のユーザーが後で登録リンクを見つけたときに、その登録リンクが使用される可能性があります。

詳細情報

属性	説明
構成名	*sn_ext_usr_reg.Reg_link_expiration_days*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	整数
推奨値	3
デフォルト値	3
カテゴリ	認証
セキュリティリスク	• 重大度スコア：中 • CVSS スコア：6.6 • セキュリティリスクの詳細：このプロパティを整数 3 に設定しないと、意図しないユーザーによって登録リンクが使用される可能性があります。
依存関係と前提条件	なし

1 日あたりのパスワードリセット **SMS** の最大数を最小化する (セキュリティセンター **1.3** で更新)

検証目的で送信される 1 日あたりの SMS コードの最大数を管理します。

password_reset.sms.max_per_day プロパティは、ユーザーが検証のために送信できる 1 日あたりの SMS コードの最大数を表します。

詳細情報

属性	説明
構成名	*password_reset.sms.max_per_day*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	整数
推奨値	10
デフォルト値	10
カテゴリ	認証
セキュリティリスク	• 重大度スコア：5.9 • CVSS スコア：中 • セキュリティリスクの詳細：このプロパティが推奨値の **10** 以下に設定されていない場合は、SMS コードの総当たり攻撃が容易になります。
依存関係と前提条件	なし

高保証セッションのログイン試行失敗を最小限に抑える

ハイアシュアランスセッションのログイン試行の失敗を最小限に抑えることで、総当たり攻撃の可能性を低減します。

継続的認証による再認証時にユーザーがログアウトされるまでに許容される認証試行の失敗回数を制限するには、 **glide.zta.high_assurance.session.max.login.failed_attempts** システムプロパティを使用します。

総当たり攻撃の可能性を減らすには、このシステムプロパティの値を低い値 (5 など) に設定します。

詳細情報

属性	説明
技術的な構成名	glide.zta.high_assurance.session.max.login.failed_attempts
プラグインの適用性	ゼロトラスト:継続的認証 (com.snc.zero_trust_continuous_authentication)
セキュリティリスク	許可された認証試行回数が多いと、総当たり攻撃の可能性が高くなります。
共通脆弱性スコアリングシステム (CVSS) スコア	3.3
共通脆弱性スコアリングシステム (CVSS) 評価	低
機能への影響	ユーザーは、プロパティで選択された認証失敗回数を超えると、セッションからログアウトされます。
依存関係と前提条件	なし

属性	説明
データタイプ	整数
ベースシステム値	5
フォールバック値	5
推奨値	5

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

パスワードリセット要求の有効期限を最小化する (セキュリティセンター **1.3** で更新)

password_reset.request.expiry は、ユーザーがパスワードリセットプロセスを実行する必要がある期間 (分単位)を指定します。

注: *password_reset.request.expiry* プロパティの設定は、デフォルトで 12 時間に設定される *glide.pwd_reset.onetime.token.validity* プロパティの設定よりも優先されます。

詳細情報

属性	説明
プロパティ名	*password_reset.request.expiry*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	認証
目的	ユーザーがパスワードリセットプロセスを実行する必要がある期間 (分単位) を指定します。
推奨値	**10** 以下の整数に設定します。デフォルト値は 10 です。
構成タイプ	整数値
セキュリティリスク	(中) プロパティが 10 以下の推奨値に設定されていない場合、他のユーザーがその要求を推測して使用し、パスワードのリセットを試行する可能性が高くなります。
セキュリティリスク評価	4.2
参照	パスワードリセットプロパティの設定

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

パスワードリセット要求の成功ウィンドウの持続期間を最小化する (セキュリティセンター **1.3** で更新)

password_reset.request.success_window プロパティは、パスワードが正常にリセットされた後で、ユーザーがパスワードのリセットまたは変更を待機する必要がある時間 (分数) を制御します。指定された期間、ユーザーはパスワードを再度リセットすることができません。

詳細情報

属性	説明
プロパティ名	*password_reset.request.success_window*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	認証
目的	パスワードを正常にリセットした後で、ユーザーがパスワードを再度リセットするために待機しなければならない時間 (分数) を示します。
推奨値	1440
デフォルト値	1440
構成タイプ	正の整数値
セキュリティリスク	(高) プロパティが 1440 以下の推奨値に設定されていない場合、他の誰かがパスワードリセット機能を悪用してユーザーアカウントに不正にアクセスする可能性が高くなります。
セキュリティリスク評価	4.9
参照	パスワードリセットプロパティの設定

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

パスワードリセットの **SMS** 有効期限を最小化する (セキュリティセンター **1.3** で更新)

SMS コードの有効期限が切れるまでの時間 (分) をコントロールします。

password_reset.sms.expiry プロパティは、SMS コードの有効期限が切れるまでの時間 (分) を表します。

詳細情報

属性	説明
構成名	*password_reset.sms.expiry*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	整数
推奨値	5
デフォルト値	5
カテゴリ	認証
セキュリティリスク	• 重大度スコア：5.6 • CVSS スコア：中

属性	説明
	• セキュリティリスクの詳細：このプロパティが推奨値の **5** 以下に設定されていない場合、攻撃者が SMS コードを推測して使用し、パスワードをリセットする可能性が高くなります。
依存関係と前提条件	なし

ワンタイム帯域外検証のライフタイム持続期間を最小化する **(Security Center 1.3** で更新)

帯域外検証ツールの期間を管理します。

帯域外検証ツールは、1 回限りのコードでの代替配信方法です。たとえば、マルチファクタートークンのリセットなどです。多要素認証 プラグインでアドミニストレーターがこのメソッドを有効にしている場合、ワンタイムコードがメールで配信されます。1 回限りの帯域外検証ツールを 10 分後に期限切れになるように設定して、有効期間を制限します。有効期間が長いほど、フィッシング、ソーシャルエンジニアリング、ショルダーサーフィン攻撃などの不正な手段によってコードが侵害される可能性が高くなります。

詳細情報

属性	説明
構成名	*glide.multifactor.onetime.code.validity*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	整数
推奨値	10
デフォルト値	10
カテゴリ	認証
セキュリティリスク	• 重大度スコア：3.9 • CVSS スコア：低 • セキュリティリスクの詳細：1 回限りの帯域外検証ツールを 10 分後に期限切れになるように設定します。時間が長くなると、攻撃者によってコードが侵害されるリスクが高まります。
依存関係と前提条件	多要素認証
参照	マルチファクター認証基準

SAML の「notBefore」または「notOnOrAfter」の制約期間を最小化する (セキュリティセンター **1.3** および **1.5** で更新)

このプロパティを構成し、SAML 要求と応答が有効であると見なされる猶予期間を追加します。

このプロパティは、SAML 要求と応答が有効であると見なされる猶予期間を追加します。このプロパティ値は、ID プロバイダー (IdP) クロックとサービスプロバイダー (SP) クロックの時間差を考慮して、 *NotBefore* および *NotOnOrAfter* の制約に追加する秒数を表します。これらの制約は、指

定された期間内に行われない要求を拒否することで、リプレイ攻撃から防御します。IdP クロックと SP クロックが大幅に異なる場合、ネットワーク遅延により SAML 要求が認可されないことがあります。

詳細情報

属性	説明
構成名	*glide.authenticate.sso.saml2.clockskew*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	文字列
推奨値	60 未満
デフォルト値	180
カテゴリ	認証
セキュリティリスク	• 重大度スコア：7.5 • CVSS スコア (CVSS score)：高 • セキュリティリスクの詳細:プロパティを 60 以上の値に設定すると、制約によるリプレイ攻撃の防御が妨げられる可能性があります。
依存関係と前提条件	なし

パスワードリセット/変更プロセス中にユーザーに通知する **(Security Center 1.5** で削除)

このプロパティを使用して、エンドユーザーを有効にし、セルフサービスプロセスを使用して、パスワードをリセットまたは変更します。

このプロパティを使用すると、エンドユーザーはセルフサービスプロセスを使用してパスワードをリセットまたは変更できます。または、エンドユーザーのパスワードをリセットするためにサービスデスク担当者を必要とするプロセスを組織で実装することもできます。パスワードの変更やリセットプロセスでパスワードの更新についてユーザーに通知されない場合、知らないうちに攻撃者がユーザーをアカウントからロックアウトできる可能性があります。これにより、攻撃者が悪意のあるアクティビティを実行する時間が長くなります。パスワード変更またはリセットの際にパスワードリセットプロセスがユーザーに通知されることを確認してください。

詳細情報

属性	説明
構成名	*pwd_process.change, pwd_process.reset*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	認証

属性	説明
セキュリティリスク	• 重大度スコア：8.1 • CVSS スコア：高 • セキュリティリスクの詳細：パスワードの変更がリセットされたときにユーザーに通知されない場合、知らないうちに攻撃者がユーザーをアカウントからロックアウトできる可能性があります。
依存関係と前提条件	なし

マルチファクターセットアップで許可されるバイパスを減らす

マルチファクターセットアップで許可されるバイパスを減らすことで、アカウントが侵害されるリスクのある期間を短縮します。

ユーザーがマルチファクターパスコードのセットアップ要件をスキップできる回数を減らして、アカウントが侵害されるリスクにさらされる期間を減らします。マルチファクター認証 (MFA) は、追加の形式の検証を要求することで、パスワード関連の攻撃や脆弱なパスワードから保護します。ユーザーがこのセットアップをスキップできる時間を短縮すると、この脆弱性が軽減されます。

glide.authenticate.multifactor.setup.bypass.count システムプロパティを **0** に設定して、ユーザーがマルチファクターパスコードのセットアップをスキップできないようにします。

詳細情報

属性	説明
技術的な構成名	glide.authenticate.multifactor.setup.bypass.count
プラグインの適用性	なし
セキュリティリスク	マルチファクター認証は、追加の形式の検証を要求することで、パスワード関連の攻撃や脆弱なパスワードから保護します。許可されたマルチファクターセットアップのバイパスの数が多いと、アカウントがマルチファクターで保護されていないため、アカウントの侵害のリスクが高まります。少数のバイパスを許可すると、この期間が短縮されます。
共通脆弱性スコアリングシステム (CVSS) スコア	3.9
共通脆弱性スコアリングシステム (CVSS) 評価	低
機能への影響	ユーザーは、プロパティで指定されたログイン回数を超えた場合、MFA を設定せずにインスタンスにログインすることはできません。
依存関係と前提条件	なし
データタイプ	整数
ベースシステム値	0
フォールバック値	0
推奨値	0

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

ようこそページからの認証情報の削除

ようこそページのデフォルトコンテンツを変更し、デフォルトの認証情報を削除します。

2 つの [ログイン方法 **(How to Login)]** レコードは、CMS プラグインのデモデータの一部としてインストールされます。

注: インスタンスのデモデータをインストールしない場合、これらのレコードは存在しません。その場合、推奨されるセキュリティプラクティスに従って、構成はセキュリティ準拠と見なされます。

詳細情報

属性	説明
名前	*How to login*
構成タイプ	テーブル：sys_home
カテゴリ	認証
目的	デモデータとともに追加されたデフォルトの認証情報をようこそページから削除する。
推奨値	デモデータが利用されなかった場合は False または null。
デフォルト値	なし。これはテーブル構成であり、Glide プロパティではありません。したがって、デフォルト値はありません。
構成タイプ	ブーリアン
セキュリティリスク	(中) デモデータは CMS プラグイン用に提供されています。これには、ようこそページに含まれる 2 つのデフォルトパスワードが含まれています。これが削除されない場合、権限のない攻撃者がインスタンスにアクセスできる可能性があります。
参照	https://support.servicenow.com/kb_view.do?sysparm_article=KB0550107 ようこそページ

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

カスタマーサービスアプリケーションでのゲストウォークアップエクスペリエンスに **Captcha** を必須とする (セキュリティセンター **1.3** の新機能、1.5 で更新)

ゲストウォークアップエクスペリエンス用のキャプチャは、ユーザーにキャプチャ検証の完了を要求することで、認証されていないゲストユーザーが予約を作成するのを防ぎます。

詳細情報

属性	説明
構成名	*sn_guest_walkup_cs.captcha.enabled*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	認証
セキュリティリスク	• 重大度スコア：3.7 • CVSS スコア：低 • セキュリティリスクの詳細：キャプチャが有効になっていない場合、スパム予約が自動的に作成されてシステムが過負荷状態になったり、利用可能なすべての予約スポットがいっぱいになったりするサービス拒否 (DoS) 攻撃が発生する可能性があります。
依存関係と前提条件	なし
機能への影響	このプロパティは、「CSM ゲストウォークアップチェックイン」ウィジェットのキャプチャを有効または無効にします。デフォルトでは true に設定されています。

クラシックモバイルアプリ **UI** の難読化を必須とする (セキュリティセンター **1.3** で更新)

glide.ui.m.blur_ui_when_backgrounded プロパティを使用して、バックグラウンド処理中に画像を保存する際に、スナップショットのすべてのフィールドを難読化します。

Android デバイスでは、アプリケーションがバックグラウンドに送信されるときに、Android オペレーティングシステムによって最新のタスクメニューで使用されるスクリーンショットが取得されます。ユーザーは、アプリケーションのスクリーンショットを手動で撮影し、公開状態でデバイスに保存することもできます。

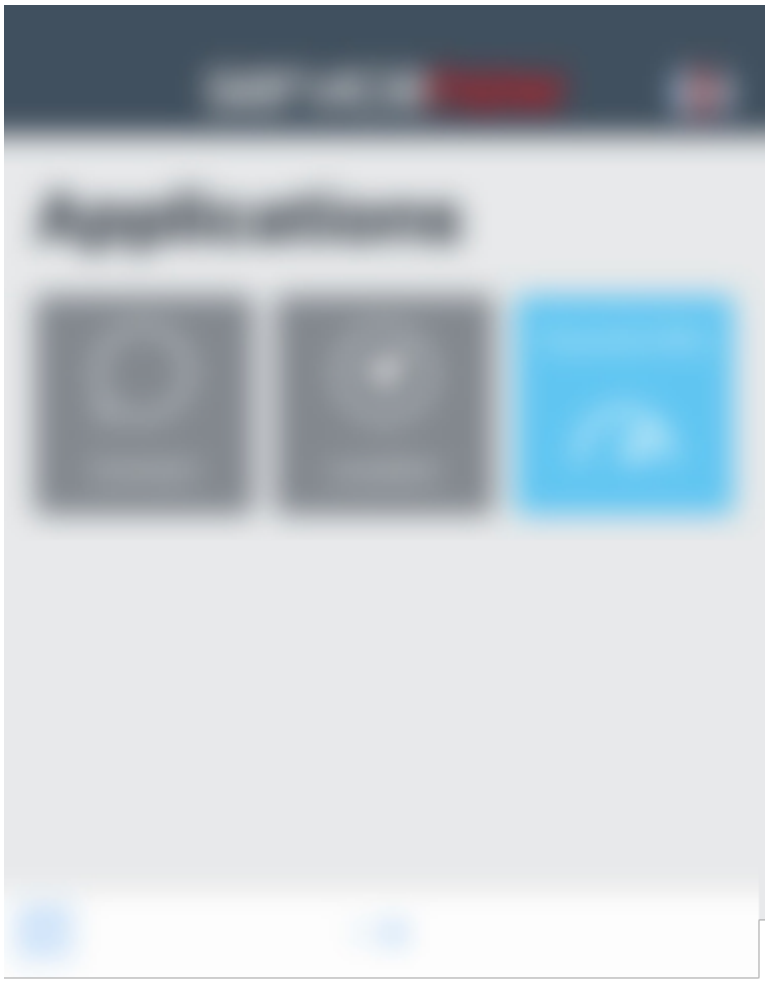
iOS デバイスでは、iOS オペレーティングシステムによってアプリケーションで画像ファイルを保存できるようにすることもできます。このファイルは、アプリケーションがバックグラウンドに送信されたときにユーザーが最後に表示した画面です。ユーザーエクスペリエンスを向上させることが目的ですが、画像が PNG 画像ファイルとして保存されるため、セキュリティ上のリスクも生じます。

注: この設定や構成はインスタンス単位で行われるため、ユーザーは プロパティを設定してインスタンスに接続する必要があります。

ServiceNow Classic アプリのスナップショットのフィールドをすべて難読化するには、「セキュリティ向上のためのアプリをぼかすオプションの設定」を参照してください。

例

このプロパティを true に設定すると、バックグラウンドアプリケーションは、iOS デバイスでは難読化され、Android iOS デバイスではブラックアウトされます。



詳細情報

属性	説明
プロパティ/プラグイン名	*glide.ui.m.blur_ui_when_backgrounded*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	認証
目的	バックグラウンド処理中に画像を保存する際に、スナップショットのすべてのフィールドを難読化すること
推奨値	true
セキュリティリスク評価	2.4
機能への影響	*glide.ui.m.blur_ui_when_backgrounded* プロパティが true に設定されている場合、ネイティブアプリはサーバーで定義されたパラメーターを使用して、アプリがバックグラウンドに入ったときに画面をぼかします。

属性	説明
	• アプリがバックグラウンドに入ると、iOS と Android で撮影されたスクリーンショットがぼやけて表示されます。 • アプリがバックグラウンドに送信されるとコンテンツが表示されなくなるため、ユーザーエクスペリエンスに悪影響を与える可能性があります。
セキュリティリスク	(中) 侵害された (ジェイルブレイクされた) デバイスは、ファイルシステムへのフルアクセス権を持ち、機密情報が埋め込まれたファイル/スナップショットにアクセスできます。

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

モバイルアプリ **UI** の難読化を必須とする (セキュリティセンター **1.3** で更新)

アプリがバックグラウンドで実行されているときにアプリの UI がぼやけるように、*glide.sg.blur_ui_when_backgrounded* プロパティを設定します。

このプロパティが推奨値の **true** に設定されていない場合、アプリスイッチャーから表示するとモバイルアプリのユーザーインターフェイスが表示されます。UI は、アプリがバックグラウンドで実行されている場合でも表示され、エンドユーザーに低レベルのセキュリティと機密性を提供します。

詳細情報

属性	説明
構成名	*glide.sg.blur_ui_when_backgrounded*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	false
カテゴリ	ファイルとリソース
セキュリティリスク	• 重大度スコア：2.4 • CVSS スコア (CVSS score)：低 • セキュリティリスクの詳細 (Security risk details)：値を **true** に設定すると、アプリがバックグラウンドで実行されているときに UI がぼやけて表示されるため、ローカルデバイスの機密性とプライバシーが向上します。
依存関係と前提条件	なし
参照	クラシックモバイルアプリ UI の難読化を必須とする (セキュリティセンター 1.3 で更新)

パスワードの最小長を設定する (セキュリティセンター 2.2 で更新)

コンプライアンスの問題を回避し、総当たり攻撃が成功するリスクを軽減するために、パスワードの最小長を設定します

コンプライアンスの問題を回避し、総当たり攻撃が成功するリスクを軽減するには、パスワードの長さを 12 文字以上にする必要があります。

インスタンスで使用されているすべての使用中のパスワード認証情報ストアについて、パスワードリセットの認証情報ストア [pwd_cred_store] テーブルの関連レコードで [パスワードポリシーを有効にする] フィールドを選択して、パスワードポリシーが適用されていることを確認します。

次に、パスワードポリシー [password_policy] レコードテーブルでレコードを開き、[パスワードの最小長] フィールドを **12** 以上に設定します。関連するパスワードポリシーレコードは、パスワードリセットの認証情報ストア [pwd_cred_store] レコードの [パスワードポリシー] フィールドにあります。

パスワードポリシーの設定の詳細については、「 インスタンスでのパスワードポリシーの有効化」を参照してください。

詳細情報

属性	説明
技術的な構成名	パスワードリセットの認証情報ストア [pwd_cred_store] テーブルとパスワードポリシー [password_policy] テーブルのレコード。
プラグインの適用性	なし
セキュリティリスク	パスワードの最小長ポリシー [password_policy] レコードの値を 12 未満に設定すると、コンプライアンスの問題が発生し、攻撃者がパスワードの総当たり攻撃に成功するリスクが高まる可能性があります。
共通脆弱性スコアリングシステム (CVSS) スコア	5.9
共通脆弱性スコアリングシステム (CVSS) 評価	中
機能への影響	インスタンスは、パスワードの最小長が 12 であっても影響を受けません。
依存関係と前提条件	なし
データタイプ	整数
ベースシステム値	8
フォールバック値	8
推奨値	12

パスワードリセットの OTP 期限を 1 時間に設定 (Security Center 2.0 で更新)

Password Reset メールのリンクの期限を制御します。

glide.pwd_reset.onetime.token.validity システムプロパティにより、プロパティで指定された時間数の後にパスワードリセットメール内のリンクが期限切れになります。パスワードリセット

トークンの有効期限は、通常のユーザーエクスペリエンスを妨げないように、できるだけ短く保つ必要があります

プロパティ値を 1 (時間単位) に設定します。

詳細情報

属性	説明
構成名	*glide.pwd_reset.onetime.token.validity*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	整数
推奨値	1
デフォルト値	1
フォールバック値	1
カテゴリ	認証
セキュリティリスク	• 重大度スコア：4.6 • CVSS スコア：中 • セキュリティリスクの詳細:パスワードリセットトークンの有効期間が長いほど、リセットトークンを含むメールが漏洩または侵害された場合に、悪意のあるアクターがアカウント乗っ取りを実行するための期間が広がります。
依存関係と前提条件	なし

ビジネスロジック

このカテゴリでは、一般的な安全原則に基づいて各アプリケーションに固有のロジックとフローを確認します。具体的には、ビジネスロジックフローの意図された順序がバイパスできないこと、自動攻撃を検出して防止するための制限が存在すること、およびなりすまし、改ざん、情報開示、および権限昇格攻撃に対する保護が存在することを確認します。

ServiceNow AI Platform 内の機密エンティティへの不正アクセスを制限するためにアドミニストレーターが設定できるセキュリティコントロールは次のとおりです。

1 日あたりのユーザー 1 人あたりのコメント最大数の制限

1 日あたりの QA コメントの数を制限するように *sn_kb_social_qa.max_comments_per_user_daily* プロパティを設定します。

このプロパティが、推奨値の **500** 以下に設定されていない場合、1 日あたりの QA コメント数に制限がなくなり、これはリソースの枯渇につながる可能性があります。

詳細情報

属性	説明
構成名	*sn_kb_social_qa.max_comments_per_user_daily*

属性	説明
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	整数
推奨値	500
デフォルト値	500
カテゴリ	ビジネスロジック
セキュリティリスク	• 重大度スコア：3.7 • CVSS スコア：低 • セキュリティリスクの詳細：このプロパティが、推奨値の 500 以下に設定されていない場合、1 日あたりの QA コメント数に制限がなくなり、リソースの枯渇につながる可能性があります。
依存関係と前提条件	なし

1 日あたりのユーザー 1 人あたりのサブスクリプション最大数の制限

この *sn_kb_social_qa.max_subscriptions_per_user_daily* プロパティを構成して、ユーザーが 1 日に登録できるサブスクリプションの最大数を制限します。

このプロパティが推奨値の **500** 以下に設定されていない場合、ユーザーが 1 日に登録できる Q&A 質問の最大数に制限はありません。この制限がないとリソースが枯渇し、ご自分のインスタンスの可用性に影響を与える可能性があります。

詳細情報

属性	説明
構成名	*sn_kb_social_qa.max_subscriptions_per_user_daily*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	整数
推奨値	500
デフォルト値	500
カテゴリ	ビジネスロジック
セキュリティリスク	• 重大度スコア：3.7 • CVSS スコア：低 • セキュリティリスクの詳細：リソースの枯渇を防ぐには、このプロパティの値を 500 以下に設定します。
依存関係と前提条件	なし

SMTP 受信者の数量を最小化する (セキュリティセンター 1.3 で更新)

glide.email.smtp.max_recipients は、1 件のメール通知でインスタンスが **To:** の行に一覧表示できる受信者の最大数を指定します。

glide.email.smtp.max_recipients を **100** 以下の推奨値に設定します。この制限を超える通知については、代わりに受信者リストの一部に宛てた複製メール通知を作成します。各メール通知には同じ最大数の受信者が指定されています。

詳細情報

属性	説明
プロパティ名	*glide.email.smtp.max_recipients*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	ビジネスロジック
目的	このプロパティがデフォルト値の 100 を超える安全でない値に設定されている場合、インスタンスでサービス拒否が発生する可能性があります。
推奨値	100
デフォルト値	100
構成タイプ	整数
セキュリティリスク	(中) この制限を超える通知については、代わりに受信者リストの一部に宛てた複製メール通知を作成します。
セキュリティリスク評価	4.9

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

タイムアウトゲストセッション

システムプロパティを使用して、非認証ユーザーの非アクティブセッションタイムアウトを制御します。

glide.guest.session_timeout システムプロパティを使用して、非認証ユーザーの非アクティブセッションタイムアウト期間 (分) を設定します。このプロパティの値を増やすと、インスタンスのセッション保持時間がデフォルトの 30 分を超えて延長されます。タイムアウト値が大きいと、インスタンスによって永続化するセッション数が増加し、可用性に関する軽微な問題が発生する可能性があるため、避けてください。

詳細情報

属性	説明
構成名	*glide.guest.session_timeout*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	整数 (分)
推奨値	30
デフォルト値	30
フォールバック値	0
カテゴリ	ビジネスロジック

属性	説明
セキュリティリスク	• 重大度スコア:4.3 • CVSS スコア：中 • セキュリティリスクの詳細:タイムアウト値を大きくすると、インスタンスでの同時セッション数が増加し、可用性に関する軽微な懸念が生じる可能性があります。
依存関係と前提条件	なし

リモートホストの検証

攻撃者がネットワークで内部ポートスキャンを使用するのを防ぐには、このプロパティを true に設定します。

glide.update_set.remote.check_host プロパティが 推奨値の **true** に設定されていない場合、 リモートインスタンステスト機能では、悪意のある攻撃者がネットワークの脆弱性を発見するために使用できる方法である内部ポートスキャンが許可されます。その後、特定のホストで開いているすべてのポートを列挙し、場合によっては応答データを抽出して、情報漏えいや不正なデータアクセスにつながる可能性があります。

⚠ **警告:** これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

詳細情報

属性	説明
構成名	*glide.update_set.remote.check_host*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	ビジネスロジック
セキュリティリスク	• 重大度スコア：6.3 • CVSS スコア：中 • セキュリティリスクの詳細：このプロパティを推奨値の **true** に設定しないと、攻撃者が内部ポートスキャンを使用して許可されていないデータにアクセスできる可能性があります。
依存関係と前提条件	なし
参照	https://support.servicenow.com/kb?id=kb_article_view&sysparm_article=KB0755132 リモートインスタンスを定義

通信

このコントロールにより、強力なアルゴリズムと暗号を使用して適切な暗号化が行われます。これには、クライアント接続での推奨されるバージョンの TLS の使用、強力な暗号スイートの使用、信頼できる署名付き証明書の使用の確認、およびコンポーネント間での接続の暗号化と接続エラーのログ記録の確認が含まれます。

証明書の信頼を強制する (Security Center 1.3 で更新、 2.0 で削除、7.0 で追加)

システムプロパティを使用して、ホスト検証が実行されないときに、送信 HTTPS コールエンドポイントから受信した証明書について、証明書の有効期限と信頼性がチェックされるようにします。

*com.glide.communications.trustmanager_trust_all*が **true** に設定されている場合、ホスト検証が実行されないと、送信 HTTPS コールエンドポイントから受信した証明書の有効期限と信頼はチェックされません。

com.glide.communications.trustmanager_trust_all システムプロパティが推奨値の **false** に設定されていることを確認します。これにより、インスタンスは JVM 証明書ストアに対して検証できる証明書のみを信頼するようになります。自己署名証明書およびエンタープライズ署名証明書は信頼されません。このプロパティは、*com.glide.communications.httpclient.verify_hostname*が **false** に設定されている場合にのみ適用されます。

注: これらのプロパティの値は であるため、一度変更すると変えることができなくなります (元に戻すことはできません)。セキュリティ上の理由から、このプロパティ値は変更しないでください。さらに質問がある場合は、カスタマーサービス & サポートにお問い合わせください。

詳細情報

属性	説明
プロパティ名	*com.glide.communications.trustmanager_trust_all*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	通信
目的	送信要求の証明書の検証を適用すること
推奨値	false
セキュリティリスク評価	5.7
機能への影響	この修正では、証明書 CA (認証局) フィールドに厳格な検証が適用されます。信頼できるエンティティ (CA) が証明書を発行した場合、インスタンスはそれを受け入れてその後使用します。
セキュリティリスク	(中) 機密性と完全性の理由から、アプリケーションはどのトランザクション操作に対しても、証明書を使用する前に証明書の CA を検証する必要があります。
参照	証明書 証明書チェーンとホスト名の検証 (Security Center 1.3 の新機能、2.0 で更新)

発信 SSLv2/SSLv3 接続を無効化する (Security Center 1.3 で更新)

glide.outbound.sslv3.disabled プロパティを設定して、REST 要求や SOAP 要求などの送信接続を行うときに、MID サーバーが TLS を使用するように強制します。通常は、インスタンスからの送信接続で SSL の代わりに TLS が使用されます。

詳細情報

属性	説明
プロパティ名	*glide.outbound.sslv3.disabled*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	通信
目的	ServiceNow インスタンスからのすべての送信接続で TLS の使用を強制すること
推奨値	true
デフォルト値	false **重要:** *glide.outbound.sslv3.disabled*プロパティの値は安全な上書きであり、一度変更すると変更できません。
セキュリティリスク評価	6.5
機能への影響	この修正では、HTTPS での通信時に TLS プロトコルバージョンの使用が強制されます。インスタンスの顧客/ユーザーが TLS 通信をサポートしていないデバイスを使用している場合は、機能停止する可能性があります。
セキュリティリスク	(中) SSL の従来のバージョンは、HTTP セキュアシェル実装に使用した場合、BEAST や SSL ハートブリードなどのクライアント側の攻撃のためにセキュアでないことが確認されています。

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

アクティブな SAML 構成でデモ認証を使用しない (セキュリティセンター 1.5 で更新)

本番 SAML 構成でデモ証明書を使用するかどうかを制御します。

ServiceNow によって提供されるデモ証明書は、既知のパスフレーズを持つすべてのインスタンスに共通するものであるため、本番 SAML 構成では使用しないでください。証明書キーストアを使用する SAML プロパティのいずれかがアクティブ (*require_signed_authnrequest*、*require_signed_logoutrequest*、または *encrypt_assertion*) の場合は、デモデータを使用しないでください。デモデータはすべてのインスタンスで共有されるため、共有証明書で署名された要求の整合性は保証されません。したがって、IDP によって暗号化されたメッセージが、インターセプトされた場合、攻撃者によって復号化される可能性があります。

詳細情報

属性	説明
構成名	*glide.authenticate.sso.saml2.keystore*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	文字列
推奨値	カスタムキーストアの sys_id
デフォルト値	空白文字列
カテゴリ	通信
セキュリティリスク	• 重大度スコア：3.9 • CVSS スコア：低
依存関係と前提条件	なし

廃止された **TLS** バージョンを無効にする

非推奨の TLS バージョンを無効にすることで、機密データの損失や漏洩を回避します。

弱い TLS 1 および TLS 1.1 プロトコルバージョンでのデータ転送を防止するために、他のサーバーと通信するときに、インスタンスがトランスポートレイヤーセキュリティ (TLS) バージョン 1.2 を介した通信のみをネゴシエートすることを確認します。

TLS バージョン 1.2 以降のみを使用するには、**com.glide.communications.disable.deprecated.tls** を **true** に設定します。

詳細情報

属性	説明
技術的な構成名	com.glide.communications.disable.deprecated.tls
プラグインの適用性	なし
セキュリティリスク	古いサポート対象外の TLS バージョンを使用すると、機密データが失われたり漏洩したりする可能性があります。
共通脆弱性スコアリングシステム (CVSS) スコア	4.4
共通脆弱性スコアリングシステム (CVSS) 評価	中
機能への影響	このプロパティが **true** に設定されている場合、より弱い TLS 1 または 1.1 プロトコルを必要とする古くて安全でないサーバーは、インスタンスと通信できません。
依存関係と前提条件	なし
データタイプ	ブール
ベースシステム値	true
フォールバック値	true

属性	説明
推奨値	true

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

ネットワークエラーに OCSP チェックを強制する (Security Center 1.3 の新機能、2.0 で更新)

攻撃者がオンライン証明書ステータスプロトコル (OCSP) チェックをバイパスしないように *com.glide.communications.httpclient.ocsp_allow_network_error* プロパティを構成する方法について説明します。

com.glide.communications.httpclient.ocsp_allow_network_error が推奨値の false に設定されていないときに 、オンライン証明書ステータスプロトコル (OCSP) チェックでネットワークエラー (タイムアウトや失効情報の取得に関する問題など) が発生した場合、OCSP セキュリティチェックがバイパスされ、チェックに成功したと見なされます。これにより、失効した証明書を持つ攻撃者が、Web の基盤となる公開鍵インフラストラクチャ (PKI) およびデジタル証明書の信頼を破壊する可能性があります。失効した証明書の使用は、サーバーが同期していない場合を除いて、多くの場合、悪意のあるアクティビティのインジケーターになります。

プロパティ *com.glide.communications.httpclient.ocsp_allow_network_error* が存在し、false に設定されていることを確認します。プロパティがsys_propertiesテーブルに含まれていない場合は、新しいレコードを追加してください。

詳細情報

属性	説明
構成名	*com.glide.communications.httpclient.ocsp_allow_network_er*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	false
デフォルト値	true
カテゴリ	通信
セキュリティリスク	• 重大度スコア：5.9 • CVSS スコア：中 • セキュリティリスクの詳細：このプロパティを false に設定しないと、攻撃者が OCSP セキュリティチェックをバイパスできる可能性があります。
依存関係と前提条件	なし
機能への影響	このプロパティは、接続エラーまたはタイムアウトエラーが発生した場合に、機関情報アクセス (AIA) オンライン証明書ステータスプロトコル (OCSP) URI に対する要求が成功または失敗の結果になるかどうかを決定します。false に設定すると、提示されたサーバー証明書の失効ステータスを検証できないときに、そのエンドポイントとの通信エラーが発生します。プロパティがデフォルト値の true に

属性	説明
	設定されているときにネットワークエラーが発生した場合、証明書は失効しているという観点から有効として扱われます。

証明書チェーンとホスト名の検証 (Security Center 1.3 の新機能、2.0 で更新)

com.glide.communications.httpclient.verify_hostname プロパティを構成して、証明書検証プロセスが確実に実行されるようにすることで、中間者攻撃を防ぎます。

Glide プロパティ *com.glide.communications.httpclient.verify_hostname* が安全な値である true に設定されていない場合、ServiceNow インスタンスから開始された TLS 接続中にリモートホストによって提示されたホスト名と証明書チェーンは検証されません。これにより、TLS 接続のセキュリティが侵害され、2 者間の通信が傍受される中間者攻撃が可能になります。これにより、機密データが開示される可能性があります。

詳細情報

属性	説明
構成名	*com.glide.communications.httpclient.verify_hostname*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	false
カテゴリ	通信
セキュリティリスク	• 重大度スコア：高 • CVSS スコア：7.4 • セキュリティリスクの詳細：*com.glide.communications.httpclient.verify_hostname* を推奨値の true に設定しないと、インスタンスが中間者攻撃に対して脆弱になる可能性があります。
依存関係と前提条件	なし
機能への影響	リモート SSL (Secure Socket Layer) ホストによって提示されたホスト名と証明書チェーンを検証します。このプロパティを true に設定し、中間者 (MITM) 攻撃から保護します。 **注:** このプロパティは *com.glide.communications.trustmanager_trust_all* プロパティを上書きします。

証明書の失効を検証する (Security Center 1.3 の新機能)

com.glide.communications.httpclient.verify_revoked_certificate プロパティは 、トランスポートレイヤーセキュリティ (TLS) ハンドシェイク中に証明書の失効をチェックし、セキュリティチェックがバイパスされていないことを確認します。

com.glide.communications.httpclient.verify_revoked_certificate が推奨値の **true** に設定されていない場合、TLS ハンドシェイク中に証明書の失効がチェックされません。TLS は、インターネット経由で送信されるデータを暗号化して、悪意のある人物にパスワードやクレジットカード番号などの機密情報が表示されないようにします。失効した証明書を持つ攻撃者が、有効な証明書の提供を省略することができ、公開鍵インフラストラクチャ (PKI) とデジタル証明書の信頼を破る可能性があるため、TLS ハンドシェイクのバイパスはセキュリティ上のリスクです。

詳細情報

属性	説明
構成名	*com.glide.communications.httpclient.verify_revoked_certif*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	通信
セキュリティリスク	• 重大度スコア：6.5 • CVSS スコア：中 • セキュリティリスクの詳細：*com.glide.communications.httpclient.verify_revoked_c* を推奨値の true に設定しないと、TLS ハンドシェイク中に証明書の失効がチェックされません。
依存関係と前提条件	なし
機能への影響	認証されたエンドポイントとトランスポートレイヤーセキュリティ (TLS) セッションを確実に開始するために、このプロパティは true に設定する必要があります。このプロパティが false に設定されている場合、証明書がチェックされず、インスタンスのセキュリティが侵害される可能性があります。

設定

構成カテゴリは、アプリケーションにセキュアなビルド環境と強化されたサードパーティのライブラリーコンポーネントがあることを確認します。具体的には、ビルドと展開のパイプラインが繰り返し可能で自動テストが含まれていることを確認し、既知のセキュリティの問題が展開されないようにします。これには、依存関係を最新の状態に保ち、既知の脆弱性を排除することが含まれます。

コンテンツタイプの自動設定オプション (セキュリティセンター **1.3.3** で削除)

インスタンスでコンテンツタイプオプションの自動設定プロパティを設定して、MIME 混乱攻撃を防止します。

このプロパティを使用して、X-Content-Type-Options 応答 HTTP ヘッダーを制御します。X-Content-Type-Options 応答 HTTP ヘッダーは、Content-Type ヘッダーで公示される MIME タイプに従う必要があることを示すためにサーバーで使用されます。このプロパティが false に設定されている場合、攻撃者は MIME 混乱攻撃を行う可能性があります。true に設定されている場合、この

ヘッダーにより、ブラウザが HTTP ヘッダーのコンテンツタイプ以外のものとしてファイルを解釈するのを防止します。

⚠ **警告:** このプロパティの値は、DB オーバーライドなしです。変更またはオーバーライドすることはできません。

詳細情報

属性	説明
構成名	*glide.security.header.auto_set_x_content_type_options*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	false
カテゴリ	設定
セキュリティリスク	• 重大度スコア：7.3 • CVSS スコア (CVSS score)：高 • セキュリティリスクの詳細 (Security risk details)：このプロパティを **false** に設定すると、攻撃者が MIME 混乱攻撃を行う可能性があります。
依存関係と前提条件	なし
参照	システムプロパティを追加する

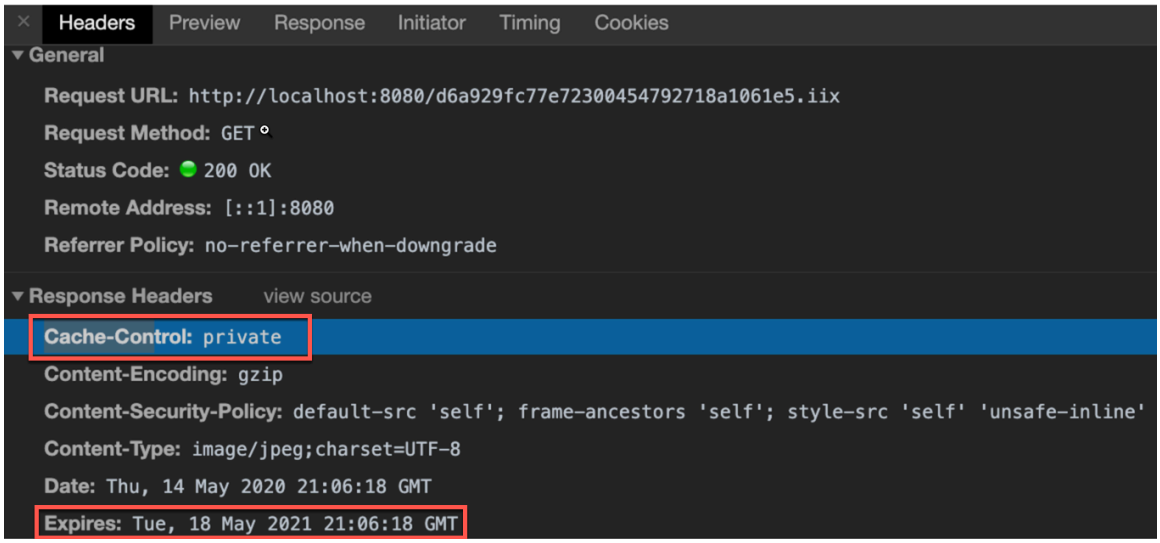
キャッシュ制御 **HTTP** ヘッダー値 **(Security Center 1.3** で更新、1.5 で削除)

glide.http.cache_control プロパティを使用して、ページの静的コンテンツデータを要求するときに ServiceNow AI Platform から送信される HTTP 応答ヘッダーのデフォルトのキャッシュ制御値を設定します。静的コンテンツには、たとえばページで内部からレンダリングされる画像、CSS、および JavaScript などがあります。

glide.http.cache_control プロパティは、HTTP 応答ヘッダーのデフォルトのキャッシュ制御値を **private** または **public** に設定します。デフォルトは **private** です。

値	説明
private	静的コンテンツはブラウザー (クライアント) レベルでキャッシュできますが、プロキシサーバーレベルではキャッシュできません。
公開	静的コンテンツはブラウザー (クライアント) レベルでキャッシュでき、またプロキシサーバーレベルでもキャッシュできます。

HTTP 応答ヘッダーの Expires の値は、静的コンテンツの有効期限を制御します。デフォルト値は 369 日です。デフォルト値を手動で上書きするには、*glide.http.expire.days* プロパティを使用します。



注: *glide.http.cache* プロパティを使用して、HTTP 応答ヘッダーの Cache-Control と Expires の値の設定を有効にするか無効にするかを指定できます。デフォルトは **true** で、以下の項目を設定できます。

- *glide.http.cache_control* プロパティを使用してデフォルトの Cache-Control (キャッシュ制御) 値を設定可能
- *glide.http.expire.days* プロパティを使用してデフォルトの Expires (有効期限) 値を設定可能

詳細情報

属性	説明
プロパティ名	*glide.http.cache_control*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	設定
目的	静的コンテンツのキャッシュ制御 HTTP 応答ヘッダー値を設定すること
推奨値	プライベート
デフォルト値	プライベート
セキュリティリスク評価	4.3
機能への影響	HTTP 応答ヘッダーのデフォルトのキャッシュ制御値を設定します。
セキュリティリスク	(高) このプロパティを **public** に設定すると、CDN/プロキシを備えたインスタンスが静的コンテンツをキャッシュし、認証なしでレンダリングする場合があります。 • **private** は、CDN/プロキシセットアップを使用するインスタンスに適した設定です。 • インスタンスに CDN/プロキシが設定されていない場合は、どちらの値でもかまいません。

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

HTTP 応答ヘッダー構成を有効にする

システムプロパティを使用して、Web アプリの cookie/セッション関連のハイジャックのリスクを軽減します。

*glide.http.headers_config.enabled*が **true** に設定されていない場合、HTTP 応答ヘッダー [sys_response_header] テーブルで定義された応答ヘッダー構成は使用されません。セキュリティ関連の HTTP 応答ヘッダーには、XSS 関連の保護に役立つコンテンツセキュリティポリシーが含まれています。HTTP 応答ヘッダーの詳細については、「HTTP 応答ヘッダー」を参照してください。

プロパティ *glide.http.headers_config.enabled* が **true** に設定されていることを確認します。

詳細情報

属性	説明
構成名	*glide.http.headers_config.enabled*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
フォールバック値	true
カテゴリ	セッション管理
セキュリティリスク	• 重大度スコア:5.5 • CVSS スコア：中 • セキュリティリスク:HTTP 応答ヘッダーの欠落、不正確、または脆弱なセキュリティリスクにより、XSS、CSRF、および Web アプリの cookie/セッション関連のハイジャックが発生する可能性があります。
依存関係と前提条件	なし

チャットサーバーのデバッグの無効化

このプロパティを設定して、インスタンスのシステムログを有効または無効にします。

glide.cs.debug プロパティを使用して、インスタンスでのシステムログの処理方法を管理します。プロパティが推奨値の **false** に設定されている場合、システムログは有効になりません。**true** に設定されている場合、システムログが有効になります。トラブルシューティングが必要な場合にのみこのプロパティを **true** に設定し、完了したら無効にします。システムログを有効にすると、多くのメッセージが生成され、システムログが過負荷になる可能性があります。また、ログメッセージによって機密情報が誤って公開される可能性があります。

詳細情報

属性	説明
構成名	*glide.cs.debug*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	false
デフォルト値	false
カテゴリ	設定
セキュリティリスク	• 重大度スコア：2.3 • CVSS スコア：低 • セキュリティリスクの詳細：システムログを有効にすると、多くのメッセージが生成され、ログメッセージでシステムが過負荷になり、ログメッセージから機密情報が誤って公開される可能性があります。
依存関係と前提条件	なし

従来の **JQuery UI** の使用を無効にする

従来の JQuery UI の使用を無効にすることで、ライブラリにパッチ未適用の脆弱性が入らないようにします。

ライブラリにパッチ未適用の脆弱性をもたらす、古いパッチが事前に適用された JQuery UI バージョンの使用を防止します。古いバージョンを使用すると、古いバージョンの JQuery UI ライブラリで発見された脆弱性に対する攻撃によってセキュリティリスクが生じる可能性があります。

古いパッチが事前に適用された JQuery UI バージョンが使用されないように、**glide.jquery_ui.legacy** システムプロパティが **false** に設定されていることを確認します。このシステムプロパティは、組織がカスタム実装を実行するためにパッチ適用されていないバージョンに依存している場合のフェイルセーフです。

詳細情報

属性	説明
構成名	*glide.jquery_ui.legacy*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	false
デフォルト値	false
フォールバック値	true
カテゴリ	設定

属性	説明
セキュリティリスク	• 重大度スコア：3.9 • CVSS スコア：低 • 古いパッチが事前に適用された JQuery UI バージョンが使用されないようにすると、ライブラリにパッチが適用されていない脆弱性が発生する可能性があります。
機能への影響	このシステムプロパティは、組織がカスタム実装を実行するためにパッチ適用されていないバージョンに依存している場合のフェイルセーフです。
依存関係と前提条件	なし

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

ロックされたフォーム要素のデバッグを無効にする

glide.security.explain.write.locks の説明です。

ロックされたフォーム要素の説明を表示しないようにするには、*glide.security.explain.write.locks* を推奨値の **false** に設定します。ロックされたフォーム要素の説明を表示するには、値を **true** に設定します。

詳細情報

属性	説明
プロパティ名	*glide.security.explain.write.locks*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	設定
目的	他のプロパティに依存しない SecurityDebugger の表示動作を制限します。
推奨値	false
デフォルト値	false
構成タイプ	ブーリアン
セキュリティリスク	(低) ロックされたフォーム要素に説明が表示されないようにします。これにより、セキュリティデバッガーによって提供される情報が少なくなるため、アプリケーションの安全性がわずかに向上します。
セキュリティリスク評価	3.5

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

MultiSSO のデバッグを無効にする (Security Center 1.3 および 1.5 で更新)

glide.authenticate.multisso.debug プロパティは、マルチ SSO のデバッグログ記録を制御します。

詳細情報

属性	説明
プロパティ名	*glide.authenticate.multisso.debug*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	設定
目的	マルチ SSO デバッグを無効にします。
推奨値	false
デフォルト値	false
構成タイプ	ブーリアン
セキュリティリスク	(高) プロパティを推奨値の「False」に設定します。そのようにしないと、MultiSSO デバッグが有効になり、意図しない機密情報の漏洩が発生する可能性があります。
セキュリティリスク評価	4.0
参照	複数プロバイダー SSO のプロパティ、テーブル、およびスクリプト

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

ターゲットのクローンを禁止する (Security Center 1.3 の新機能)

glide.db.clone.allow_clone_target プロパティを構成して、インスタンスがクローンターゲットとして使用されるのを防ぎます。

glide.db.clone.allow_clone_target システムプロパティを **false** に設定して、インスタンスがクローンターゲットとして使用されないように保護します。システムクローンは、データベース内のすべてのものをソースインスタンスからターゲットインスタンスにコピーします。クローン作成プロセスでターゲットインスタンスのインスタンスデータベースが上書きされ、データが失われてデータの整合性が失われるため、これはセキュリティ上のリスクです。

本番インスタンスで **glide.db.clone.allow_clone_target** システムプロパティを **false** に設定して、インスタンスがクローンターゲットとして選択されないようにします。

詳細情報

属性	説明
構成名	*glide.db.clone.allow_clone_target*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	false
デフォルト値	true
カテゴリ	設定

属性	説明
セキュリティリスク	• 重大度スコア：4.4 • CVSS スコア：中 • セキュリティリスクの詳細：このプロパティを推奨値の false に設定しないと、インスタンスをクローンターゲットとして使用できるようになります。クローン作成プロセスでインスタンスデータベースが上書きされる可能性があるため、これはセキュリティ上のリスクです。
依存関係と前提条件	なし
参照	• クローン (従来の UI) • クローンターゲット (登録と認証)
機能への影響度	このプロパティは、本番インスタンスがクローンされないようにする追加の安全策を提供します。デフォルト値は、本番インスタンスの場合は false、dev や qa などの準本番インスタンスの場合は true です。インスタンスをクローンターゲットとして使用できるようにするには、このプロパティを true に設定します。

SOAP フォールトスタックトレースディスプレイの無効化

スタックトレースをインスタンスに表示する方法を管理します。

glide.soapfault.display_stack_trace プロパティを使用して、インスタンスのスタックトレースを管理します。このプロパティが **false** に設定されている場合、機密情報が漏洩する可能性があります。これが **true** に設定されている場合、スタックトレースは表示されません。

詳細情報

属性	説明
構成名	*glide.soapfault.display_stack_trace*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	false
デフォルト値	false
カテゴリ	設定
セキュリティリスク	• 重大度スコア：7.3 • CVSS スコア (CVSS score)：中 • セキュリティリスクの詳細 (Security risk details)：このプロパティを **false** に設定すると、スタックトレースから機密情報が公開される可能性があります。
依存関係と前提条件	なし

パフォーマンス監視のアクセスを制限する (Security Center 1.3 で更新)

glide.security.diag_txns_acl プロパティを使用して、認証されていない接続からの stats.do、threads.do、thread_pool_stats および replication.do アクセスを制御します。

このプロパティを **true**に設定すると、*glide.security.diag_txns_acl* プロパティはアドミニストレーターアカウントによる以下へのアクセスのみを許可します。

- https://<instancename>.servicenow.com/stats.do
- https://<instancename>.servicenow.com/threads.do
- https://<instancename>.servicenow.com/replication.do
- https://<instancename>.servicenow.com/thread_pool_stats.do

この設定を有効にしない場合、認証されていない接続からこれらのリソースに引き続きアクセスできます。

詳細情報

属性	説明
プロパティ名	*glide.security.diag_txns_acl*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	設定
目的	設定ページへのアクセスをアドミンアカウントのみに制限すること
推奨値	true
デフォルト値	true
セキュリティリスク評価	5.3
機能への影響	この修正では、ログ記録とトラブルシューティングの目的で、アドミニストレーターアカウントのみがアプリケーションの機密データにアクセスできるようにします。
セキュリティリスク	(中) サーバーの詳細、スレッド、サーバーで実行されるプロセスなどの機密データは、適切な権限のないエンドユーザーが表示したりアクセスしたりすることはできません。

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

MultiSSO プラグインの更新バージョンを有効にする (Security Center 1.3 および 1.5 で更新)

セキュリティの脆弱性を軽減するには、MultiSSO プラグインの v2 を使用していて、true に設定されていることを確認します。

インスタンスでマルチ SSO プラグインが有効になっている場合は、v2 バージョンが有効になっていることを確認して、セキュリティの脆弱性を軽減します。最新バージョンではセキュリティが強化され、アサーション暗号化のサポートや IDP によって開始されるシングルログアウト (SLO) など、より多くの機能が追加されています。最新バージョンが有効になっていない場合、新しいセキュリティ機能は使用できず、インスタンスは廃止されたプラグインを使用するリスクがあります。

KB0756504 の手順に従って、最新バージョンにアップグレードします。このプロセスには、カスタマイズ関連の変更の確認と移行、そしてバージョンのアップグレードが含まれます。完了すると、 **glide.authenticate.multissov2_feature.enabled** システムプロパティは自動的に **true** に設定されます。

詳細情報

属性	説明
構成名	glide.authenticate.multissov2_feature.enabled
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	設定
セキュリティリスク	最新バージョンが有効になっていない場合、新しいセキュリティ機能は使用できず、インスタンスは廃止されたプラグインを使用するリスクがあります
依存関係と前提条件	なし
参照	https://support.servicenow.com/kb?id=kb_article_view&sysparm_article=KB0756504

安全なリファラーポリシーを強制する (Security Center 1.3 の新機能)

com.glide.security.referrerpolicy プロパティを使用して、Referrer-Policy HTTP ヘッダーが各 ServiceNow　ページに適切なレベルのデータを送信するようにすることで、データ漏洩を防止します。

com.glide.security.referrerpolicy プロパティを default に設定すると、ServiceNow AI Platform　要求ページに合わせて特別に調整された、適切なレベルの送信済み情報を使用して Referrer-Policy HTTP ヘッダーが管理されるようになります。これにより、パスやクエリ文字列など、完全な URL の他の部分からアクセスできる可能性のある不正なデータ漏洩を防ぐことができます。

詳細情報

属性	説明
構成名	*com.glide.security.referrerpolicy*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	文字列
推奨値	デフォルト
デフォルト値	default
カテゴリ	設定
セキュリティリスク	• 重大度スコア：7.3 • CVSS スコア：中

属性	説明
	• セキュリティリスクの詳細：不正なデータの漏洩を防ぐために、*com.glide.security.referrerpolicy* プロパティが default に設定されていることを確認してください。
依存関係と前提条件	なし
参照	Referrer-Policy
機能への影響	このプロパティは、ページから要求が送信されたときに、「referrer」ヘッダーを介して送信される情報の量を制御します。 • default：インスタンスがリファラーヘッダーを処理します。 • same-origin：インスタンス/同じドメイン内で完全なリファラー URL を送信し、外部の送信元にはリファラーを送信しません。 • origin：送信元のみをリファラーとして送信元の内部と外部に送信します。 • origin-when-cross-origin：完全なリファラー URL をインスタンス/同じドメイン内および外部の送信元のみに送信します。

秘密鍵の最小サイズを確保する

システムプロパティを使用して、証明書インベントリ管理アプリケーションでの証明書署名要求 (CSR) の生成に使用する秘密鍵の最小サイズを決定します。

sn_disco_certmgmt.private_key_size システムプロパティは、証明書インベントリ管理アプリケーションでの CSR の生成に使用される秘密鍵の最小サイズを決定します。有効な選択肢は、512、1024、2048、または 4096 です。

このプロパティの値が 2048 以上に設定されていることを確認します。このプロパティの有効な選択肢は、512、1024、2048、または 4096 です。プロパティがシステムプロパティ [sys_properties] テーブルに存在しない場合、または値が無効な場合、値はデフォルトで 2048 です。

詳細情報

属性	説明
構成名	*sn_disco_certmgmt.private_key_size*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	生成された秘密キーのサイズを表す整数。512、1024、2048、または 4096 の値を使用します。
推奨値	2048
デフォルト値	2048
フォールバック値	2048

属性	説明
カテゴリ	通信
セキュリティリスク	• 重大度スコア：3.1 • CVSS スコア：低 • セキュリティリスク:2048 より小さいキーを使用すると、キーがブルートフォース攻撃を受けた場合に、将来情報が公開される可能性があります。2048 以上の有効な値を使用して、キーを長期間にわたって将来にわたって保証します。
機能への影響	レガシーシステムおよびアプリケーションは、2048 以降のキーを処理できない場合があります。
依存関係と前提条件	なし

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

X-Frame-Options：SAMEORIGIN セキュリティヘッダーを実装 (セキュリティセンター 1.3 で更新)

glide.set_x_frame_options プロパティを使用して、すべての UI ページの SAMEORIGIN に対して X-Frame-Options 応答ヘッダーを設定します。

X-Frame-Options HTTP 応答ヘッダーは、<frame> または <iframe> 内のページをブラウザーでレンダリングできるかどうかを示すために使用されます。サイトでは、この機能を使用して、そのコンテンツがその他のサイトに組み込まれることがないようにすることで、クリックジャッキング攻撃を回避することができます。攻撃者は、お客様のページを自分のページに埋め込み、ページ要素が悪意を持って実行されるようにする可能性があります。エンドユーザーは、ページが自分のページに似ているため、そのページが正当だと考えるかもしれません。エンドユーザーが通常どおりに要素をクリックするだけで、悪意のあるスクリプトまたは要素が実行される可能性があります。

詳細情報

属性	説明
プロパティ名	*glide.set_x_frame_options*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	設定
目的	クリックジャッキング攻撃の軽減
推奨値	true
デフォルト値	true
セキュリティリスク評価	5.9
機能への影響	この修正では、サードパーティアプリケーションで ServiceNow AI Platform アプリケーションを iFrame の形式でレンダリングするための制限が適用されます。このような統合がある場合、アプリケーションはカスタマイズされたサードパーティアプリにはレンダリングされません。

属性	説明
セキュリティリスク	(中) 同一生成元ポリシーを使用すると、ドメインが別のドメインからスクリプトやリソースを取得できないようにすることができます。最新のブラウザーはすべてこの機能をサポートしています。 このポリシーは、プロトコル、ポート、およびホストに基づいて接続を検証します。CORS (クロスオリジン要求) は同一生成元ポリシーを修正したものであり、ヘッダー値の一部として明示的に指定されている場合に、別のドメインのリソース/スクリプトにアクセスできるようにします。 • この場合、X-Frame-Options ヘッダーは、ServiceNow AI Platform アプリケーションをサードパーティの Web サイトでレンダリングできるかどうかを制御します。 • プロパティ値を **SAMEORIGIN** に設定するとレンダリングが行われないため、機密情報が公開されることが減少します。
参照	利用可能なシステムプロパティ iFrame の設定

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

サービスカタログアイテム追加ページへのアクセスに書き込みアクセスを必須とする (Security Center 1.3 の新機能)

glide.sc.request.add_item_write_access プロパティを使用して、カタログアイテムに対して不正な操作が実行されないようにします。

詳細情報

属性	説明
構成名	*glide.sc.request.add_item_write_access*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	false
カテゴリ	設定
セキュリティリスク	• 重大度スコア：7.3 • CVSS スコア：中 • セキュリティリスクの詳細：*glide.sc.request.add_item_write_access* プロパティが false に設定されている場合、ログインしているユーザーは誰でも [カタログアイテムの追加] UI ページにアクセスできます。これにより、カタログアイ

属性	説明
	テムに対して不正な操作が実行される可能性があります。このセキュリティリスクを修復するには、このプロパティを true に設定します。
依存関係と前提条件	なし
機能への影響	プロパティが true の場合、ユーザーは UI ページのコンテキストでレコードへの書き込みアクセス権を持っている必要があります。

サードパーティ **Web** サイトの埋め込みを防ぐために **Xframe** オプションを設定する **(Security Center 1.3** で更新)

Web アプリケーションのコンテンツがサードパーティのサイトに埋め込まれないようにするには、このプロパティを構成します。

com.glide.cs.embed.xframe_options が推奨値の DENY または SAMEORIGIN に設定されていない場合、Web アプリケーションのコンテンツは ALLOW-FROM URI を使用してサード パーティ サイトに埋め込まれる可能性があります。信頼できないサードパーティサイトを許可すると、クリックジャッキングなどの攻撃が可能になる可能性があります。

詳細情報

属性	説明
構成名	*com.glide.cs.embed.xframe_options*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	文字列
推奨値	sameorigin
デフォルト値	sameorigin
カテゴリ	設定
セキュリティリスク	• 重大度スコア：3.1 • CVSS スコア：低 • セキュリティリスクの詳細：このプロパティを推奨値に設定しないと、Web アプリケーションのコンテンツがサードパーティのサイトに埋め込まれ、クリックジャッキングなどの攻撃が可能になる可能性があります。
依存関係と前提条件	なし

データ保護

データ保護カテゴリは、データの機密性、完全性、および可用性 (CIA) の要素を対象としています。

CIA コンポーネントは次のとおりです。

- 機密性：データは転送中および停止中に不正アクセスから保護されます。
- 完全性：データは不正な作成、削除、変更から保護されます。
- 可用性：必要に応じてデータにアクセスできます。

[記憶する] の削除

glide.ui.forgetme プロパティを使用して、ログインページから [記憶する] チェックボックスを削除して、ログイン情報がキャッシュされないようにします。

詳細情報

属性	説明
プロパティ名	*glide.ui.forgetme*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	データ保護
目的	認証情報がキャッシュされないようにすること。
セキュリティリスク評価	3.5
推奨値	true
デフォルト値	true
機能への影響	この修正では、セッションが期限切れになるとインスタンスから自動的にログアウトされるため、ユーザーエクスペリエンスが変わります。セッションの有効期限は、「ユーザーセッションの管理」で詳述されているようにシステムプロパティに設定された値にのみ依存します。
セキュリティリスク	(低) [記憶する] チェックボックスをログイン時にオンにした場合、ユーザーのコンピューターに追加の Cookie が保存されます。 • その目的は、ログイン中のユーザーが再度アクセスした場合にセッションを自動的に再確立することです。 • ユーザーが意図的にログアウトするまでユーザーセッションはアクティブにできるため、セキュリティ上のリスクがあります。エンドユーザーがブラウザーをそのままにして席を離れた場合、または別の攻撃によって侵害された場合、このシナリオでの攻撃の可能性が高くなります。
参照	[記憶する] チェックボックスを削除する

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

モバイルアプリケーションのバックグラウンド処理中にペーストボードをクリアすることを必須とする **(Security Center 1.3** の新機能、1.5 で更新)

*glide.sg.clear_pasteboard_when_backgrounded*プロパティは、ServiceNow モバイルアプリからコピーしたテキストを、アプリがバックグラウンドモードになった後もクリップボードとペーストボードに保持するかどうかを制御します。推奨値の true に設定されていない場合、機密情報が

Android または iOS クリップボードに開示され、デバイス上の他のアプリケーションに公開される可能性があります。

詳細情報

属性	説明
構成名	*glide.sg.clear_pasteboard_when_backgrounded*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	false
カテゴリ	データ保護
セキュリティリスク	• 重大度スコア：3.5 • CVSS スコア：低 • セキュリティリスクの詳細：このプロパティが推奨値の true に設定されていない場合、機密情報が Android または iOS のクリップボードに開示され、デバイス上の他のアプリケーションに開示される可能性があります。
依存関係と前提条件	なし
機能への影響	このプロパティは、ServiceNow アプリがバックグラウンドに入ったときにコピーおよび貼り付けクリップボードをクリアします。

私用メールからの **HR** ケースの更新を制限する (セキュリティセンター **1.3** の新機能、1.5 で更新)

sn_hr_core.restrict_guest_email プロパティを使用して、ユーザーが個人メールで HR ケースに返信できるかどうかを制御します。

sn_hr_core.restrict_guest_email プロパティが true に設定されていない場合、ユーザーは、作業メモに含める HR ケースを参照するメールを個人アカウントから送信できます。これにより、個人のメールが侵害された場合や安全でない通信が行われた場合に、機密性や整合性に関する軽微な問題が発生する可能性があります。admin は、どのユーザーが個人メールアカウントにアクセスしているか確かでないため、ユーザーが個人メールから HR ケースに応答する機能を制限することができます。

詳細情報

属性	説明
構成名	*sn_hr_core.restrict_guest_email*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	false

属性	説明
カテゴリ	データ保護
セキュリティリスク	• 重大度スコア：3.5 • CVSS スコア：低 • セキュリティリスクの詳細：このプロパティを true に設定しないと、個人のメールが侵害された場合や安全でない通信が行われた場合に、機密性や整合性に関する軽微な問題が発生する可能性があります。
依存関係と前提条件	なし
機能への影響	このプロパティは、個人メールアドレスからの返信で HR ケースを更新するかどうかを制御します。true に設定すると、個人用メールからの返信はすべてケースノートに追加されます。false の場合、ケースとメモは更新されません。

OAuth パラメーターを POST 本文に制限する (Security Center 1.3 の新機能)

glide.oauth.allow.parameters.in.post.body.only プロパティを使用して、受信 OAuth 認証によるアクセストークンの受け入れを制御します。アクセストークンは機密性が高く、POST 要求本文内にある場合にのみ受け入れるようにする必要があります。

詳細情報

属性	説明
構成名	*glide.oauth.allow.parameters.in.post.body.only*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	データ保護
セキュリティリスク	• 重大度スコア：4.2 • CVSS スコア：中 • セキュリティリスクの詳細: *glide.oauth.allow.parameters.in.post.body.only* が推奨値の **true** に設定されていない場合、アクセストークンが GET 要求パラメーターに存在する可能性があります。これらのアクセストークンは、クライアントログとインフラストラクチャログに残り、これらのログが漏洩した場合、アカウントの乗っ取りにつながる可能性があります。
依存関係と前提条件	なし

属性	説明
参照	• OAuth 2.0 • OAuth トークンの管理
機能への影響度	oauth_token.do プロセッサが、サポートされているすべての権限許可タイプに対する入力として POST 本文パラメーターのみを受け付けることを確認します。

エラー処理とログ記録

エラー処理とログ記録のカテゴリは、ステークホルダーに公開されるログ情報の品質と詳細度に対応します。

これには、ログとエラーメッセージが機密情報を収集せず、分類に従ってデータを正しく保護し、適切な有効期間を設定していることの確認が含まれます。また、このカテゴリは、適切なエラー処理に関連し、さらにセキュリティに影響を与える未処理の例外の詳細なスタックトレースなどの機密性の高いエラーをエンドユーザーに公開しないことに関連しています。

スクリプトサンドボックスで優先度「低」のユーザーのロガーを無効にする **(Security Center 1.3** で更新)

サンドボックス環境で実行されているスクリプトをログ記録する Glide システムの機能を管理します。

glide.security.sandbox_no_logging プロパティを使用して、サンドボックス環境で実行されているスクリプトをログ記録する Glide システムの機能を制御します。*glide.security.sandbox_no_logging* が **false** に設定されている場合、サンドボックススクリプトを使用して、権限の低いユーザーがログ記録を利用できます。権限の低いユーザーがログを挿入して、攻撃者が攻撃を難読化できる可能性があるため、これは潜在的なセキュリティの脆弱性です。サンドボックススクリプトを使用している権限の低いユーザーがログ記録機能を使用しないようにするには、プロパティを **true** に設定します。

詳細情報

属性	説明
構成名	*glide.security.sandbox_no_logging*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	エラー処理とログ記録
セキュリティリスク	• 重大度スコア：2.2 • CVSS スコア：低 • セキュリティリスクの詳細：このプロパティを **false** に設定すると、権限の低いユーザーのログ記録が有効になり、攻撃者が攻撃を難読化する可能性があります。

属性	説明
依存関係と前提条件	なし

安全な **Cookie** のデバッグの無効化

インスタンス内の Cookie に関連するログメッセージを管理します。

glide.secure_cookie.debug プロパティを使用して、Cookie に関連するログメッセージを管理します。このプロパティを **false** に設定すると、ログメッセージは表示されません。**true** に設定すると、SecureUserCookie クラスおよび Cookie クラスのメッセージがログに記録されます。これにより、機密情報がインスタンスで公開される可能性があります。

詳細情報

属性	説明
構成名	*glide.secure_cookie.debug*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	false
デフォルト値	false
カテゴリ	エラー処理とログ記録
セキュリティリスク	• 重大度スコア：4.2 • CVSS スコア：中 • セキュリティリスクの詳細：このプロパティを **true** に設定すると、機密情報が公開される可能性があります。
依存関係と前提条件	なし

SQL エラーメッセージを無効にする **(Security Center 1.3** および **1.5** で更新)

glide.db.loguser プロパティを使用して、SQL エラーメッセージがブラウザーでレンダリングされないようにします。

*glide.db.loguser*が推奨値の false に設定されていない場合、機密性の高いサーバー側のエラーメッセージがエンドユーザーに表示される可能性があります。エラーメッセージには、スタックトレースやデータベースの構造に関する情報が含まれており、前提条件が存在する場合、SQL インジェクションを成功させるために必要な知識を攻撃者に提供する可能性があります。多層防御のため、これらのエラーメッセージはエンドユーザーに表示されません。

詳細情報

属性	説明
プロパティ名	*glide.db.loguser*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	エラー処理とログ記録

属性	説明
目的	ブラウザー内での SQL エラーメッセージの表示を無効にすること
タイプ	ブール
推奨値	false
デフォルト値	true
セキュリティリスク評価	3.1
機能への影響	この修正により、SQL エラーメッセージの表示が無効になります。機能への影響はありません。
セキュリティリスク	(中) 攻撃者に役立つ可能性のある機密の SQL 情報は、Web ページのエラーメッセージに含まれて表示されることはありません。

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

MID 監査ログを有効化する (セキュリティセンター 1.3 の新機能、1.5 で更新)

MID サーバーコマンド監査ログには、コマンド名、コマンドハッシュ、使用された認証情報の名前、実行状況などの詳細が記録されます。

有効にすると、ecc_agent_command_audit_logテーブルでagent_security_adminロールを持つユーザーが、または **MID** サーバー > コマンド監査ログ.

MID サーバープロパティ [ecc_agent_property] テーブルで *mid.log.command_audit.enable* を true に設定すると、MID サーバーによって実行されるコマンドの監査がオンになります。

詳細情報

属性	説明
構成名	*mid.log.command_audit.enable*
構成タイプ	MID サーバープロパティ [ecc_agent_property] レコード
データタイプ	ブーリアン
推奨値	true
デフォルト値	false
カテゴリ	エラー処理とログ記録
セキュリティリスク	• 重大度スコア：2.2 • CVSS スコア：低 • セキュリティリスクの詳細：セキュリティ調査の際に、インシデント応答チームはこのテーブルを使用して、MID サーバーで実行されるコマンドを監査できます。このログがないと、アカウントの不正使用などの状況に対応するための十分な詳細情報がない可能性があります。
依存関係と前提条件	なし

Protected Tables プラグインの有効化 (Security Center 1.3 の新機能)

com.glide.security.protected_table.enabled プロパティを使用して、特権の高いユーザーによってログテーブルが改ざんされるのを防ぎます。

com.glide.security.protected_table.enabled プロパティが **true** に設定されている場合、Protected Tables プラグインを使用して、権限の高いユーザーによってインスタンスでログテーブルが改ざんされることを防ぎます。このプロパティが **true** に設定されている場合、次のログテーブルには特別な保護が適用されます。

- syslog (データベース上書きなし)
- syslog_transaction
- sys_outbound_http_log
- sysevent
- sys_audit
- sys_push_notification
- protected_table_configuration (データベース上書きなし)

ログの完全性は、カスタマーアドミンがインスタンスで悪意のあるアクティビティを判断するために重要です。

詳細情報

属性	説明
構成名	*com.glide.security.protected_table.enabled*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	false
カテゴリ	エラー処理とログ記録
セキュリティリスク	• 重大度スコア：4.5 • CVSS スコア：中 • セキュリティリスクの詳細：*com.glide.security.protected_table.enabled* を推奨値の true に設定しないと、高い権限を持つユーザーによってインスタンスでログテーブルが改ざんされる可能性があります。
依存関係と前提条件	なし
参照	システムログ

すべての送信 HTTP 要求フィールドをログに記録する [Security Center v1.3.2 で削除]

glide.outbound_http.security.log.allow.all.fields プロパティを false に設定し、機密性の高い送信 HTTP フィールドがプレーンテキストでログに記録されないようにします。

このプロパティが推奨値の **false** に設定されていない場合、機密性の高い送信 HTTP フィールドがプレーンテキストでログに記録される可能性があります。これにより、機密データや認証情報を含む送信要求が暗号化されていないプレーンテキストでログに記録され、特権の低いユーザーが表示できるため、企業ネットワークのセキュリティ体制が低下する可能性があります。

詳細情報

属性	説明
構成名	*glide.outbound_http.security.log.allow.all.fields*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	整数
推奨値	30
デフォルト値	30
カテゴリ	エラー処理とログ記録
セキュリティリスク	• 重大度スコア：6.8 • CVSS スコア：中 • セキュリティリスクの詳細：このプロパティを **false** に設定しないと、機密性の高い送信フィールドがプレーンテキストでログに記録される可能性が高くなり、セキュリティリスクを生じます。
依存関係と前提条件	なし

HTML のサニタイズをログに記録 **(Security Center 2.0** で削除)

glide.html_sanitize.discarded_log.enable プロパティを設定して、HTML サニタイゼーションイベントをインスタンスに記録するかどうかを決定します。

このプロパティが推奨値の **true** に設定されていない場合、HTML サニタイゼーションイベントは *sys_log* テーブルに記録されません。ログ記録がないと、インスタンスの自動セキュリティ検出および調査機能に悪影響を与える可能性があります。

詳細情報

属性	説明
構成名	*glide.html_sanitize.discarded_log.enable*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	エラー処理とログ記録
セキュリティリスク	• 重大度スコア：2.4 • CVSS スコア (CVSS score)：低

属性	説明
	• セキュリティリスクの詳細 (Security risk details)：このプロパティが推奨値の **true** に設定されていない場合、HTML サニタイゼーションイベントは *sys_log* テーブルに記録されず、自動セキュリティ検出および調査機能に影響を与える可能性があります。
依存関係と前提条件	なし
参照	HTML サニタイザーの有効化

セッション監査イベントのログ記録 (Security Center 1.3 の新機能、1.5 で更新)

glide.authenticate.session_access.log_audit_event プロパティを **true** に設定すると、セッション監査イベントが *sys_session_access_audit* テーブルに作成されます。

Glide プロパティ *glide.authenticate.session_access.log_audit_event* が true に設定されている場合、セッション監査イベントがsys_session_access_auditテーブルに作成されます。悪意のあるアクターの調査を支援するために、セッションにアクセスしたユーザーに関する情報をログに記録することをお勧めします。ログに記録される情報には、ユーザー、セッション ID (非機密)、IP アドレス、ロール、ポリシーが含まれます。

注: *glide.authenticate.session_access.log_audit_event* システムプロパティはゼロトラストアクセスに固有です。詳細については、「ゼロトラストアクセス (ZTA)」を参照してください。

詳細情報

属性	説明
構成名	*glide.authenticate.session_access.log_audit_event*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	エラー処理とログ記録
セキュリティリスク	• 重大度スコア：6.3 • CVSS スコア：中 • セキュリティリスクの詳細：このプロパティを推奨値の true に設定しないと、イベントをログに記録できなくなります。これにより、サイバー攻撃が発生した場合に攻撃者を見つけられなくなる可能性があります。
依存関係と前提条件	なし

ユーザーの代理操作をログ記録 (Security Center 1.3 および 2.0 で更新)

glide.sys.log_impersonation を構成して、ユーザー代理操作イベントをインスタンスのログに記録するかどうかを制御します。

このプロパティが推奨値の **true** に設定されていない場合、ユーザー代理操作イベントがログに記録されなくなります。ログ記録がないと、インスタンスの自動セキュリティ検出および調査機能に影響を与える可能性があります。プロパティ *glide.sys.log_impersonation* が存在し、true に設定されていることを確認します。プロパティがsys_propertiesテーブルに含まれていない場合は、新しいレコードを追加してください。

詳細情報

属性	説明
構成名	*glide.sys.log_impersonation*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	エラー処理とログ記録
セキュリティリスク	• 重大度スコア：6.4 • CVSS スコア：中 • セキュリティリスクの詳細：このプロパティが **true** に設定されていない場合、ユーザー代理操作イベントがログに記録されなくなり、インスタンスのセキュリティ検出および調査機能に影響を与える可能性があります。
依存関係と前提条件	なし
参照	ユーザーの代理操作

詳細な **HTTP** 要求のログ記録を防止する

詳細な HTTP 要求ログ記録を減らすことで、機密情報へのアクセスを防止します。

送信 HTTP 要求のログ記録レベルを制御して、認証ヘッダーや cookie などの機密情報へのアクセスを防止します。この情報は、要求されたリソースにアクセスするための認証情報のように使用できます。

これらの要求のログ記録レベルは、 **glide.outbound_http_log.override** プロパティと **glide.outbound_http_log.override.level** プロパティによって制御されます。**glide.outbound_http_log.override** が **true** に設定されている場合、要求と応答のログレベルは **glide.outbound_http_log.override.level** プロパティによって制御されます。**glide.outbound_http_log.override.level** が **all** または **elevated** に設定されている場合、要求ヘッダーと応答ヘッダーがログに記録されます。

glide.outbound_http_log.override を **false** に、**glide.outbound_http_log.override.level** を **basic** に設定します。これらのプロパティがシステムプロパティ [sys_properties] テーブルに表示されない場合、デフォルトでは安全な状態になっています。

自動翻訳

詳細情報

属性	説明
技術的な構成名	• glide.outbound_http_log.override • glide.outbound_http_log.override.level
プラグインの適用性	なし
セキュリティリスク	詳細な設定でログに記録された送信 HTTP 要求ヘッダーには、認証ヘッダーや cookie などの機密情報が含まれている可能性があります。この情報は、要求されたリソースにアクセスするための認証情報のように使用できます。 送信 HTTP ログ [sys_outbound_http_log] テーブルへのアクセス権を持つユーザーは、この情報を表示できます。重大度は、作成される送信要求のタイプによって異なります。
共通脆弱性スコアリングシステム (CVSS) スコア	5
共通脆弱性スコアリングシステム (CVSS) 評価	中
機能への影響	**glide.outbound_http_log.override** システムプロパティを使用すると、送信 http 要求ログレベルを上書きできます。値が **false** の場合、ログレベルはデフォルトで basic に設定されます。 **glide.outbound_http_log.override** が **true** に設定されている場合、ログ記録のレベルは **glide.outbound_http_log.override.level** プロパティの値によって決まります。この値は、 基本、 昇格、または すべてにすることができます。3 つすべてが文字列/テキストベースの値です。これら以外の値は 、基本として解釈されます。 その他の詳細については、「送信ログ記録の構成」を参照してください。
依存関係と前提条件	なし
データタイプ	• ブール • 文字列
ベースシステム値	• false • <空>
フォールバック値	• false • <空>

属性	説明
推奨値	• false • basic

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

インポートプロセッサの詳細な SQL エラーメッセージをオフにする (Security Center 1.3 で更新)

このプロパティを構成して、詳細な SQL エラーメッセージを表示するかどうかを制御します。

このプロパティが **false** に設定されている場合は、機密情報が漏洩する可能性がある詳細な SQL エラーメッセージが表示されます。これを回避するには、プロパティを **true** に設定して汎用メッセージを表示します。

詳細情報

属性	説明
構成名	*glide.import.error_message.generic*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	エラー処理とログ記録
セキュリティリスク	• 重大度スコア：3.1 • CVSS スコア：低 • セキュリティリスクの詳細：このプロパティを **false** に設定すると、情報漏えいにつながる可能性がある詳細な SQL メッセージが有効になります。
依存関係と前提条件	なし

ファイルとリソース

ファイルとリソースカテゴリにより、アプリケーションが信頼できないファイルデータを安全に処理し、権限が制限された信頼できないソースからの信頼できないデータを適切な場所に保存できるようになります。

これには、サイズが大きいファイルまたは予期しないファイルタイプによるサービス拒否の回避、ファイルタイプの検証、パストラバースの防止などのコントロールが含まれます。

感染したファイルのダウンロードを許可しない (セキュリティセンター 1.5 および 2.0 で更新)

ウイルス対策サービスが停止しているか到達不能な場合に、ユーザーがスキャンされていない添付ファイルをダウンロードできるかどうかを制御します。

com.glide.snap.infected_download_allowed プロパティが true に設定されている場合、ウイルス対策サービスが停止または到達不可能になった場合でも、ユーザーはスキャンされていない添付ファイルをダウンロードできます。この状況では、ユーザーが悪意のあるファイルをダウンロードするリスクにさらされる可能性があり、特にデバイスに他のエンドポイント保護がインストールされていない場合は、デスクトップが感染するリスクがあります。

プロパティ *com.glide.snap.infected_download_allowed* が false に設定されていることを確認します。

詳細情報

属性	説明
構成名	*com.glide.snap.infected_download_allowed*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	false
デフォルト値	false
カテゴリ	ファイルとリソース
セキュリティリスク	• 重大度スコア：6.7 • CVSS スコア：中 • セキュリティリスクの詳細：このプロパティを推奨値の false に設定しないと、インスタンスが悪意のあるファイルのダウンロードにさらされる可能性があります。
依存関係と前提条件	なし

スパムメールのスコアリングとフィルタリングを有効化する (Security Center 1.3 で更新)

メールフィルター (*com.glide.email_filter*) プラグインをインストールして、インスタンス内にメールフィルタリングをインストールします。このフィルタリングでは、既存のヘッダーを識別し、関連するヘッダーに基づいてメールの処理方法を決定できます。または、*com.glide.email_filter* を false に設定します。

ServiceNow AI Platform メールサーバーを介して送信されるメッセージはすべて、スパムメールの可能性について評価されます。

注: インスタンスがプライベートメールサーバーを使用している場合は、このトピックは適用できません。詳細については、「スパムメールのスコアリングとフィルタリング」を参照してください。

必須条件

このプロパティを設定する前に、次の手順を実行してください。

glide.email.read.active プロパティを true に設定します。詳細については、「独自の POP3 サーバーの使用を有効にする」を参照してください。

詳細情報

属性	説明
プラグイン名	com.glide.email_filter、glide.email.read.active
構成タイプ	[システム定義] > [プラグイン]
カテゴリ	ファイルとリソース
目的	フィルタリングを適用してメールのスパム送信を回避すること
推奨値	いずれも： • **glide.email.read.active** プロパティを **false** に設定します • **glide.email.read.active** プロパティを **true** に設定し、メールフィルター (com.glide.email_filter) プラグインを有効にします。 アクティブ
デフォルト値	なし。これはプラグインであり、Glide プロパティではないため、デフォルト値はありません。
セキュリティリスク評価	8.1
機能への影響	スパムスコアリングの一環として、メールがインスタンスからフィルタリング、ブロック、隔離されることはありません。スコアリングのみが行われた後、インスタンスに送信されます。フィルタリングはすべて、Email Filters プラグインを使用して インスタンス内で行われます。
セキュリティリスク	(中) メールフィルターを使用すると、アドミニストレーターは条件ビルダーまたは条件付きスクリプトを使用して、既知/不明の送信者からの悪意のある受信メールを無視するタイミングを指定できます。
参照	メールフィルター https://support.servicenow.com/kb_view.do?sysparm_article=KB0549426

プラグインのアクティブ化の詳細については、「プラグインを有効にする」を参照してください。

ウイルス対策スキャンを有効にする

com.glide.snap.enable_scan プロパティはウイルス対策スキャン機能を有効にします。

ウイルス対策スキャンを有効にするには、*com.glide.snap.enable_scan* を推奨値の **true** に設定します。

⚠ **警告:** これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

詳細情報

属性	説明
プロパティ名	*com.glide.snap.enable_scan*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	ファイルとリソース
目的	特定のインスタンスでアンチウイルススキャンを有効または無効にします。
推奨値	true
デフォルト値	true
データタイプ	ブーリアン
セキュリティリスク	(高) アンチウイルススキャンを使用すると、インシデント、問題、ストーリーなど、システムレコードに添付ファイルによって持ち込まれるウイルス感染からインスタンスを保護できます。
セキュリティリスク評価	7.7

静的コンテンツでダウンロード可能なファイルの種類を制限する (セキュリティセンター **1.3** で更新)

glide.ui.strict_customer_uploaded_static_content プロパティを使用して、ファイルのアップロード機能でアップロードされた場合にダウンロード可能になるファイルタイプを制限できるようにします。

このプロパティを *glide.ui.strict_customer_uploaded_content_types* プロパティとともに使用すると、制限付きダウンロード可能ファイルタイプのカンマ区切りリストが作成されます。

⚠ **警告:** このプロパティの値は、DB オーバーライドなしです。変更またはオーバーライドすることはできません。

詳細情報

属性	説明
プロパティ名	*glide.ui.strict_customer_uploaded_static_content*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	ファイルとリソース
目的	アプリケーションからダウンロードできるファイルタイプを確実に安全なものにすること
推奨値	true
デフォルト値	true
セキュリティリスク評価	3.1
機能への影響	この修正では、*glide.ui.strict_customer_uploaded_content_types* プロパティで指定された値に基づいてファイルのダウンロードが制限されます。

属性	説明
セキュリティリスク	(低) ファイルのダウンロード制限は、信頼できないユーザー入力ソースに適用する必要があります。

GraphQL エンドポイントのトレーニングおよび予測フローの添付ファイルサイズの制限 (Security Center 1.3 の新機能、1.5 で更新)

glide.platform_ml_di.max_attachment_size_graphql プロパティは、トレーニングおよび予測フローの GraphQL エンドポイントで返される添付ファイルの最大サイズをコントロールします。

詳細情報

属性	説明
構成名	*glide.platform_ml_di.max_attachment_size_graphql*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	整数
推奨値	5,242,880
デフォルト値	5,242,880
カテゴリ	ファイルとリソース
セキュリティリスク	• 重大度スコア：7.3 • CVSS スコア：中 • セキュリティリスクの詳細：このプロパティが推奨値である 5,242,880 未満に設定されていない場合、大きなファイルが返されるとサービス拒否 (DoS) が発生する可能性があります。
依存関係と前提条件	なし

トレーニングおよび予測フローの添付ファイルサイズの制限 (Security Center 1.3 の新機能、1.5 で更新)

glide.platform_ml_di.max_attachment_size プロパティは、トレーニングおよび予測フローで返される添付ファイルの最大許容サイズを制御します。

詳細情報

属性	説明
構成名	*glide.platform_ml_di.max_attachment_size*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	整数
推奨値	4,000,000
デフォルト値	4,000,000
カテゴリ	ファイルとリソース

属性	説明
セキュリティリスク	• 重大度スコア：7.3 • CVSS スコア：中 • セキュリティリスクの詳細：*glide.platform_ml_di.max_attachment_size* が推奨値の 4,000,000 以下に設定されていない場合 、大きなファイルが返されるとサービス拒否 (DoS) 攻撃が発生する可能性があります。
依存関係と前提条件	なし

HTTP 応答の本文サイズを制限する (セキュリティセンター 1.3 の新機能、1.5 で更新)

glide.http.response.get_body.limit.enabled および *glide.http.response.get_body.limit* プロパティを構成して、*OutOfMemoryExceptions* からインスタンスを保護します。

glide.http.response.get_body.limit.enabled および **glide.http.response.get_body.limit** システムプロパティを使用して、要求応答本文が大きすぎることが原因で発生する可能性のある OutOfMemoryExceptions を防止します。これらの例外により、サービス拒否 (DoS) 攻撃や、攻撃者がインスタンスを侵害するのに役立つその他の問題が発生する可能性があります。これらのプロパティを推奨値に設定しないと、インスタンスが OutOfMemoryExceptions やサービス拒否攻撃に対して脆弱になる可能性があります。

次のセキュリティの脆弱性からインスタンスを保護するには、次の手順を実行します。

- **glide.http.response.get_body.limit.enabled** システムプロパティを **true** に設定します。
- **glide.http.response.get_body.limit** システムプロパティが 524,288,000 メガバイト (500 MB) 以下に設定されていることを確認します。

詳細情報

属性	説明
構成名	• **glide.http.response.get_body.limit.enabled** • **glide.http.response.get_body.limit**
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	ファイルとリソース

属性	説明
セキュリティリスク	• 重大度スコア：3.1 • セキュリティリスクの詳細：これらのプロパティを推奨値に設定しないと、インスタンスが *OutOfMemoryExceptions* やサービス拒否攻撃に対して脆弱になる可能性があります。
依存関係と前提条件	なし
機能への影響度	このプロパティを使用すると、顧客が誤って大きなファイルをメモリに読み込んだために *OutOfMemoryException* が発生する危険性が低くなります。

メールの最大添付ファイル数の制限

インスタンスで受信メールの添付ファイルの数を設定します。

glide.email.inbound.max_attachment_count プロパティはインスタンスで受信メールの添付ファイルの最大数を制御します。このプロパティが **30** 以下の推奨値に設定されていない場合、受信メールによってインスタンスのパフォーマンスが低下する可能性があります。

詳細情報

属性	説明
構成名	*glide.email.inbound.max_attachment_count*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	整数
推奨値	30
デフォルト値	30
カテゴリ	ファイルとリソース
セキュリティリスク	• 重大度スコア：5.3 • CVSS スコア (CVSS score)：中 • セキュリティリスクの詳細 (Security risk details)：プロパティ値を 30 より大きい値に設定すると、デグレードの問題が発生する可能性があります。
依存関係と前提条件	なし
参照	メール プロパティ

添付ファイルの最大許容サイズ **(Security Center 1.3** で更新)

アップロードされた添付ファイルに許可される最大サイズ (メガバイト単位) を制御するには、 the *com.glide.attachment.max_size* プロパティを構成します。

このプロパティは、アップロードされた添付ファイルの最大サイズ (メガバイト単位) を制御します。このプロパティが推奨値の **1024** (1 ギガバイト) 以下に設定されていない場合、プラットフォー

ムはストレージを満杯にし、サービス拒否 (DoS) 攻撃につながる可能性のある大きなファイルを受け入れる可能性があります。

詳細情報

属性	説明
構成名	*com.glide.attachment.max_size*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	整数
推奨値	1024
デフォルト値	1024
カテゴリ	ファイルとリソース
セキュリティリスク	• 重大度スコア：6.5 • CVSS スコア：中 • セキュリティリスクの詳細:このプロパティが **1024** (1 ギガバイト) 以下に設定されていない場合、プラットフォームは DoS 攻撃につながる可能性のある大きなファイルを受け入れる可能性があります。
依存関係と前提条件	なし
参照	添付ファイル制限プロパティ

許可された **MIME** 子タイプを設定する **(Security Center 2.0** の新機能)

ファイルの種類が MIME (Multipurpose Internet Mail Extensions) の種類チェックに合格しないように、 *glide.security.mime.type.allowed_child_types* プロパティをセキュリティで保護された設定に構成する方法について説明します。これにより、アップロードされたファイルに対してリモートでコードが実行されるリスクが軽減されます。

glide.security.mime.type.allowed_child_types プロパティは、ファイル拡張子がアップロードされたファイル内のデータと一致しない可能性がある MIME ファイルタイプを定義します。これにより、このようなファイルタイプは MIME タイプのチェックをバイパスできます。このプロパティは、ファイルタイプペアのカンマ区切りリスト (*application/zip=application/java-archive* など) を受け入れます。この例では、プロパティがそのような値に設定されている場合、技術的に.jarファイルである拡張子.zipファイルは、不整合があっても MIME タイプのチェックに合格できます。適切に設定されていない場合、このバイパスはアップロードされたファイルのリモートコード実行につながる可能性があります。したがって、有効なユースケースが発生しない限り、常に空の文字列 ("") に設定する必要があります。たとえば、特定の MIME タイプを別のファイル拡張子で許可する必要があり、Tika 構成に従って有効である場合、これらのキーと値のペアはこのプロパティ値の一部として更新されます。

詳細情報

属性	説明
構成名	*glide.security.mime.type.allowed_child_types*
構成タイプ	システムプロパティ (/sys_properties_list.do)

属性	説明
データタイプ	文字列
推奨値	""
デフォルト値	""
カテゴリ	ファイルとリソース
セキュリティリスク	• 重大度スコア：4.6 • CVSS スコア：中 • セキュリティリスクの詳細:このプロパティを安全な値に設定しないと、構成が正しくないファイルによって、アップロードされたファイルがリモートで実行される可能性があります。
依存関係と前提条件	はい。*glide.security.mime.type.detection.allow_child_types* が true に設定されている場合、このプロパティの値は、許可された MIME 子タイプの構成済みリストと照合して検証するために使用されます。
機能への影響	ファイル拡張子がファイルの内容と一致しないが、 *tika-mimetypes.xml* の Tika サブタイプ構成に従って有効な MIME タイプをサポートすること。

AttachmentCreator SOAP Web サービスでファイルの MIME タイプを検証する (Security Center 1.3 の新機能、1.5 で更新)

glide.attachment.enforce_security_validation プロパティは、 MIME (多目的インターネットメール拡張) ファイルが検証を受けるかどうかを決定します。

不適切なファイル拡張子を使用して危険なファイルがインスタンスにアップロードされないように、添付ファイルの MIME タイプが検証されるようにしてください。

glide.attachment.enforce_security_validation システムプロパティを **true** に設定します。**true** に設定すると、ファイルは正しいファイルタイプ拡張子でアップロードされます。

詳細情報

属性	説明
構成名	*glide.attachment.enforce_security_validation*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	ファイルとリソース
セキュリティリスク	• 重大度スコア：6.7 • CVSS スコア：中

属性	説明
	• セキュリティリスクの詳細：プロパティが false に設定されている場合、アップロード中に MIME ファイルは検証されません。これにより、ファイル拡張子を変更することで、悪意のあるファイルが偽装される可能性があります。
依存関係と前提条件	なし
参照	https://developer.mozilla.org/en-US/docs/Web/HTTP/Basics_of_HTTP/MIME_types/Common_types
機能への影響度	アップロードされた添付ファイルに対して MIME タイプとファイル拡張子の検証を実行するには、このハードニング設定を true に設定します。このプロパティが false に設定されている場合、検証は実行されません。このプロパティはデフォルトで [true] に設定されています。

悪意のあるコード

悪意のあるコードのカテゴリでは、コードに脆弱性や不要な機能がないことを確認するための最善の努力が払われます。

これには、悪意のあるアクティビティに対する安全で適切な処理、時間ベースの攻撃がないこと、信頼できない宛先への送信通信がないこと、および承認されていないコードや攻撃者が制御するコードが含まれていないことの確認が含まれます。このカテゴリには、アプリケーションコードベースからの監査ライブラリまたはサードパーティライブラリが含まれます。

ルート化またはジェイルブレイクされたモバイルデバイスのブロック

ジェイルブレイクされたデバイスからの不正アクセスを防止して、インスタンスを保護します。

glide.sg.allow_rooted_jailbroken_device プロパティを使用し、ジェイルブレイクされたデバイスによる不正アクセスからインスタンスを保護します。このプロパティが **false** に設定されているときにユーザーがモバイルアプリを使用してインスタンスへの認証を試みると、次のアラートが表示されます：「This device appears to be jailbroken and cannot be used to access this instance. Please contact your ServiceNow Administrator.)」。アラートメッセージが表示されている間はアプリがフリーズします。このメッセージを消すための唯一の方法は [ログアウト] を選択することです。このプロパティが **true** に設定されている場合、ユーザーはジェイルブレイクされたデバイスを使用してインスタンスに対して認証を行います。

詳細情報

属性	説明
構成名	*glide.sg.allow_rooted_jailbroken_device*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	false
デフォルト値	false
カテゴリ	悪意のあるコード

属性	説明
セキュリティリスク	• 重大度スコア：4.5 • CVSS スコア：中 • セキュリティリスクの詳細：ジェイルブレイクされたデバイスはセキュリティが欠如するため、デバイスは攻撃者の主要な標的になります。承認されていないエンティティが企業データにアクセスすると、企業ネットワークのセキュリティ体制が弱まる可能性があります。
依存関係と前提条件	なし
参照	アクセス制御

アプリケーション構成データとスクリプトのコード署名を有効にする (セキュリティセンター **1.3** で削除)

アプリケーション構成データとスクリプトのコード署名を有効にします。

コード署名を使用すると、機密性の高いアプリケーション構成データとスクリプトを使用する前に検証できるので、セキュリティの強化に役立ちます。コード署名は、データの信頼性と整合性を確認するために後でチェックされるデータのデジタル署名を作成します。この検証により、悪意のあるデータやスクリプトがインスタンスで使用され、インスタンスが完全に侵害される恐れを防ぎます。

com.snc.kmf.signature.validation.flag プロパティを使用して、アプリケーション構成データとスクリプトのコード署名を管理します。このプロパティを **true** に設定すると、アプリケーション構成データとスクリプトのコード署名が有効になります。

⚠ 警告: これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

詳細情報

属性	説明
構成名	*com.snc.kmf.signature.validation.flag*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	false
カテゴリ	悪意のあるコード
セキュリティリスク	• 重大度スコア：6 • CVSS スコア：中 • セキュリティリスクの詳細：このプロパティを **true** に設定すると、コード署名が有効になり、機密性の高いアプリケーション構成データとスクリプトを検証することでセキュリティを強化できます。
依存関係と前提条件	なし

セッション管理

このカテゴリは、ユーザーのアプリケーションステータスのセキュリティを調べます。セッションは、各個人に対して一意で、推測または共有できず、何度かの非アクティブ期間の後や不要な場合は無効にされる必要があります。これには、Cookie ベースのセッションの Cookie 属性、セッショントークンの生成とストレージ、およびフェデレーション再認証の要件などの要素が含まれます。

モバイルセッションへの継続的認証ポリシーの適用

モバイルセッションに継続的な認証ポリシーを適用することで、セッションハイジャックのリスクを軽減します。

glide.zta.high_assurance.mobile.session.allowed システムプロパティを **false** に設定して、モバイルユーザーがゼロトラスト - 継続的認証プラグインのハイアシュアランスセッション機能を使用していることを確認します。モバイルセッションの継続認証ポリシーをバイパスするには、このプロパティを **true** に設定します。

glide.zta.high_assurance.mobile.session.allowed システムプロパティを **false** に設定します。このアクションにより、セッションが侵害された場合のセッションハイジャックと永続的なアクセスのリスクが軽減されます。このプロパティがシステムプロパティ [sys_properties] テーブルに表示されない場合、デフォルト値は **false** です。

詳細情報

属性	説明
技術的な構成名	glide.zta.high_assurance.mobile.session.allowed
プラグインの適用性	ゼロトラスト:継続的認証 (com.snc.zero_trust_continuous_authentication)
セキュリティリスク	このシステムプロパティが **true** に設定されている場合、ハイアシュアランスセッション機能はモバイルセッションには適用されません。再認証は、高ロールセッションのポリシーで定義されているようには行われません。これにより、セッションが侵害された場合のセッションハイジャックと永続的なアクセスのリスクが高まります。
共通脆弱性スコアリングシステム (CVSS) スコア	3.9
共通脆弱性スコアリングシステム (CVSS) 評価	低
機能への影響	**true** に設定すると、ハイアシュアランスセッション機能はモバイルセッションに適用されず、セッションが侵害された場合にセッションハイジャックや永続的なアクセスのリスクが高まります。
依存関係と前提条件	なし
データタイプ	ブール
ベースシステム値	false
フォールバック値	false
推奨値	false

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

絶対的なセッションタイムアウト時間を最小化する (セキュリティセンター 1.3 で更新)

glide.ui.user_cookie.max_life_span_in_days プロパティを使用して、ユーザーが [記憶する] チェックボックスを選択してログインしたときに作成される Cookie の最大有効期間を設定します。Cookie の有効期間が切れると、[記憶する] チェックボックスを選択したユーザーはインスタンスへの再認証を強制されます。

これによりユーザー Cookie の有効期間は、Cookie が最初に発行されてからの指定の日数になります。デフォルト値は 30 日で、上限は 365 日です。

注: 任意のアクティブユーザーセッションに対して最大セッション時間を適用する場合は、「ユーザーセッションの管理」を参照してください。

詳細情報

属性	説明
プロパティ名	*glide.ui.user_cookie.max_life_span_in_days*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	セッション管理
目的	特定の日数後に、[記憶する] チェックボックスをオンにしたユーザーに再認証を強制します。
推奨値	30 以下
デフォルト値	30 日間
機能への影響	このプロパティは、指定された期間が経過した後、すべての種類の Cookie ローテーションを回避することによって再ログインを強制します。
セキュリティリスク評価	4.2
セキュリティリスク	(中) ユーザーの Cookie が無期限にアクティブになるとセキュリティリスクが発生するため、時間ベースの構成で期限が切れるようにする必要があります。
参照	利用可能なシステムプロパティ 記憶する

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

アクティブセッションタイムアウト例外ロールを定義 (Security Center 1.3 の新機能)

システムプロパティを使用して、ロールをアクティブセッションのタイムアウト制限から除外します。

glide.active.session.timeout.exception.roles システムプロパティを使用して、ロールをアクティブセッションのタイムアウト制限から除外します。アクティブセッションタイムアウト機能により、認証情報を提供せずにハイジャックされたセッションを無期限に使用できないようにすることができます。内部統合アカウントロールのアクティブセッションタイムアウト制限の例外のみを考慮することがベストプラクティスです。

アクティブセッションタイムアウトから除外する必要があるロールに *glide.active.session.timeout.exception.roles* プロパティを設定します。このプロパティ値は、ロールのカンマ区切りリストです。デフォルト値は *edge_encryption,mid_server,maint* です。

詳細情報

属性	説明
構成名	*glide.active.session.timeout.exception.roles*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	文字列
推奨値	edge_encryption、mid_server、メンテナンス
デフォルト値	edge_encryption、mid_server、メンテナンス
フォールバック値	edge_encryption、mid_server、メンテナンス
カテゴリ	セッション管理
セキュリティリスク	• 重大度スコア：6.4 • CVSS スコア：中 • アクティブセッションタイムアウト制限の例外は、内部統合アカウントロールについてのみ考慮してください。ユーザーがセッションハイジャックの試みの被害者であり、例外のあるロールを持っている場合、そのセッションを使用する攻撃者は、そのセッションに対して無期限に認証を続けることができます。これにより、攻撃者がハイジャックされたアカウントを使用する時間が増えるため、セキュリティインシデントの影響が高くなる可能性があります。
依存関係と前提条件	なし

UserCookie バージョン 3.1 を有効にする (Security Center 2.0 で更新)

インスタンスで有効になっている UserCookie のバージョンを管理して、ソースコード内の秘密キーのストレージを保護します。

UserCookie v3 は、property *glide.ui.secure.cookies.use_kmf is disabled* の場合にのみ生成されます。UserCookie v3 は、HMAC の秘密鍵をソースコードに保存し、すべての顧客に対して同一であるため、安全ではありません。これにより、悪意のあるアクターがこの1つの秘密鍵を使用してユーザーセッションを乗っ取ることができます。プロパティ *glide.ui.secure.cookies.use_kmf* を true に設定すると、UserCookie v3.1 が使用され、秘密鍵が KMF などのセキュリティストレージに保存されます。

詳細情報

属性	説明
構成名	*glide.ui.secure.cookies.use_kmf*
構成タイプ	システムプロパティ (/sys_properties_list.do)

属性	説明
データタイプ	ブーリアン
推奨値	true
デフォルト値	false
カテゴリ	セッション管理
セキュリティリスク	• 重大度スコア：7.1 • CVSS スコア：高 • セキュリティリスクの詳細：これを **false** に設定すると、ハッシュベースのメッセージ認証コード (HMAC) の秘密キーがソースコードに格納されるため、セキュリティの脆弱性が発生します。
依存関係と前提条件	なし

API 要求でパスワードリセットを強制する (Security Center 1.5 で更新)

インスタンスでのパスワードリセット機能の動作方法を管理します。

ユーザーが [パスワードのリセットが必要 **(Password needs reset)**] とマークされている場合、次の認証試行時に新しいパスワードを入力する必要があります。このプロパティは、API 呼び出しを行う前にパスワードリセットが必須かどうかを制御します。このプロパティが推奨値の **true** に設定されていない場合でも、[パスワードのリセットが必要 **(Password needs reset)**] とマークされたユーザーアカウントは、ベーシック認証を通じてテーブル API をクエリすることで操作を実行できます。このセキュリティの脆弱性により、非アクティブなアカウントが侵害された場合に情報が漏洩する可能性があります。

詳細情報

属性	説明
構成名	`glide.authenticate.api.user.reset_password.mandatory`
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	false
カテゴリ	セッション管理
セキュリティリスク	• 重大度スコア：8.1 • CVSS スコア：高 • セキュリティリスクの詳細：このプロパティを **false** に設定すると、非アクティブなアカウントが侵害された場合に情報が漏洩する可能性があります。
依存関係と前提条件	なし

HTTP のみの Cookie フラグを有効にする (Security Center 1.3 で更新)

glide.cookies.http_only プロパティを使用して、機密性の高い Cookie の HTTPOnly 属性を有効にします。

HTTPOnly 属性は、JavaScript などのクライアント側スクリプトを使用した Cookie へのアクセスを許可しないため、クロスサイトスクリプティングなどの攻撃を防止するために使用します。クロスサイトスクリプティングのリスクは排除されませんが、一部の悪用ベクトルは排除されます。

⚠ **警告:** これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

詳細情報

属性	説明
プロパティ名	*glide.cookies.http_only*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	セッション管理
目的	クライアント側スクリプトが保護された Cookie にアクセスするリスクを軽減すること
推奨値	true
デフォルト値	true
セキュリティリスク評価	8
機能への影響	この修正では、セッション Cookie に HTTPOnly フラグが追加されるため、Cookie が盗まれるのを防ぎます。 • ユーザーの Cookie にアクセスするために JavaScript を必要とするカスタム機能がある場合、その機能は損なわれます。通常の状況では発生しないはずです。 • ServiceNow AI Platform がセッション管理を処理するため、カスタムスクリプトがユーザーの Cookie にアクセスする理由はありません。
セキュリティリスク	(中) アプリケーションのセッション Cookie はエンドユーザーを認証し、アプリケーションに対する暗黙的なアクセス権限を付与します。つまり、セッション Cookie の盗難やエクスポートの防止が必要になります。HTTP Only フラグは、JavaScript インジェクションまたはクロスサイトスクリプティングの脆弱性による盗難からセッション Cookie を保護します。
参照	利用可能なシステムプロパティ

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

OAuth トークンの有効期限が切れた後にセッションを無効にする (Security Center 2.0 の新機能)

glide.authenticate.oauth.post.token.expiration.cookie_auth.disabled プロパティを安全な値に設定して、セッションの作成に使用された OAuth トークンの有効期限が切れた後も、ユーザーが Cookie を介してセッションを引き続き使用できないようにします。

glide.authenticate.oauth.post.token.expiration.cookie_auth.disabled プロパティが安全な値 true に設定されていない場合、セッションの作成に使用された OAuth トークンの有効期限が切れた後も、ユーザーは Cookie を介してセッションを引き続き使用できます。これにより、Cookie が漏洩し、悪意のあるユーザーによってセッションがハイジャックされ、不正なリソースにアクセスするリスクが高まります。glide property *glide.authenticate.oauth.post.token.expiration.cookie_auth.disabled* が true に設定されていることを確認します。レコードがsys_propertiesテーブルに存在しない場合、デフォルト値は false です。

詳細情報

属性	説明
構成名	*glide.authenticate.oauth.post.token.expiration.cookie_aut*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	zboot インスタンスの場合、このプロパティは true です。更新インスタンスの場合、このプロパティはデフォルトで false です。
カテゴリ	セッション管理
セキュリティリスク	• 重大度スコア:5.4 • CVSS スコア：中 • セキュリティリスクの詳細:このプロパティが安全な値の true に設定されていない場合、セッションの作成に使用された OAuth トークンの有効期限が切れた後でもユーザーがセッションを引き続き使用する可能性があり、悪意のあるユーザーによってセッションが乗っ取られる可能性が高くなります。
依存関係と前提条件	なし
機能への影響	true:Cookie 認証は、OAuth アクセストークンの有効期限が切れるまでのみ有効です。有効期限が切れると、認証は優先されません。 false:OAuth アクセストークンの有効期限が切れた後でも Cookie 認証が優先されます。

同時インタラクティブセッションの量を最小化する (Security Center 1.3 で更新)

このプロパティを同時セッション制限プラグインとともに使用して、ユーザーが開くことができるアクティブなセッションの数を制御します。

[Glide 認証：同時インタラクティブセッションの最大数] プロパティを **Limit Concurrent Sessions** (*com.glide.limit.concurrent.sessions*) プラグインと共に使用して、ユーザーに開かれるアクティブなセッションの数を制御します。推奨値は **1** です。これにより、開かれるセッションの数が減ります (数が多いほど、セッションがハイジャックされる可能性が高くなります)。

詳細情報

属性	説明
構成名	*glide.authenticate.max.concurrent.interactive.sessions*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	整数
推奨値	1
デフォルト値	1
カテゴリ	セッション管理
セキュリティリスク	• 重大度スコア：3.7 • CVSS スコア：低 • セキュリティリスクの詳細： プロパティのデフォルト値を 1 より大きく設定すると、セッションハイジャックの可能性が高まります。
依存関係と前提条件	**Concurrent Interactive Sessions** (*com.glide.limit.concurrent.sessions*) プラグインがアクティブになっている必要があります。
参照	同時セッションの制限

全ノードを対象とした同時セッション制限 **(Security Center 1.3** で更新)

glide.authenticate.limit.concurrent.sessions.across.all.nodes プロパティを同時セッション制限プラグインとともに使用して、すべてのノードで追跡されるセッションの数を管理します。

同時セッションの制限 プラグインがアクティブな場合、ユーザーごとに開いているセッションの数を制限できます。このプラグインがアクティブなときに、(**[Glide** 認証：全ノードを対象とした同時セッション制限]) プロパティが **true** に設定されていることを確認します。これにより、開いているセッションの数が単一のアプリケーションノードではなくすべてのノードで追跡されるようになります。このプロパティが **false** に設定されている場合、複数のノードで複数のセッションを開くことができるため、セッションハイジャックの可能性が高くなります。

詳細情報

属性	説明
構成名	*glide.authenticate.limit.concurrent.sessions.across.all.n*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true

属性	説明
デフォルト値	true
カテゴリ	セッション管理
セキュリティリスク	• 重大度スコア：3.7 • CVSS スコア：低 • セキュリティリスクの詳細：同時セッション制限プラグインを使用している場合、このプロパティを **false** に設定すると、複数のノードに渡って複数のセッションが開かれ、セッションハイジャックなどのセキュリティ脆弱性の可能性が高くなります。
依存関係と前提条件	なし
参照	同時セッションの制限

Limit concurrent sessions プラグイン

com.glide.limit.concurrent.sessions プラグインを設定して、インスタンスでのセッションハイジャックの可能性を減らします。

このプラグインを使用すると、アドミニストレーターはユーザーまたはロールごとのアクティブセッションの数を制限できます。セッションハイジャックの可能性を減らすために、このプラグインを有効にして構成することをお勧めします。このプラグインが有効になっていて構成されている場合、ハイジャックされる可能性がある開かれたセッションの数が制限されます。

詳細情報

属性	説明
構成名	*com.glide.limit.concurrent.sessions*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	プラグイン
推奨値	com.glide.limit.concurrent.sessions を有効に構成
デフォルト値	com.glide.limit.concurrent.sessions を有効に構成
カテゴリ	セッション管理
セキュリティリスク	• 重大度スコア：3.7 • CVSS スコア：低 • セキュリティリスクの詳細：このプラグインがアクティブでない場合、セッションハイジャックの可能性が高くなります。
依存関係と前提条件	なし

ゲストのアクティブセッションのライフスパンを制限する (Security Center 1.3 の新機能)

glide.guest.active.session.life_span プロパティを使用して、アクティブなゲストの HTTP セッションの期間を制御します。

glide.guest.active.session.life_span プロパティは、セッションの非アクティブ状態や、ユーザーが非アクティブ状態のままでセッションがタイムアウトしてクローズするまでの時間に関係なく、アクティブなゲスト HTTP セッションに最大有効期間を適用します。設定値は分単位です。値が 0 の場合はアクティブセッションのタイムアウトが無効になります。値が大きいほど、攻撃者が盗んだセッションをより長く保持できるので、セキュリティインシデントの危険性が高まります。このプロパティは、インスタンスへの低いアクセス権を持つゲストユーザーに制限されています。

このセキュリティの脆弱性を修正するには、*glide.guest.active.session.life_span* を 0 より大きく 720 以下の値に設定します。

詳細情報

属性	説明
構成名	*glide.guest.active.session.life_span*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	整数
推奨値	1 ～ 720 (分)
デフォルト値	0
カテゴリ	セッション管理
セキュリティリスク	• 重大度スコア：4.2 • CVSS スコア：中 • セキュリティリスクの詳細： 最大有効期間を大きな値に設定すると、攻撃者がセッションを盗んだ場合にインスタンス内で操作できる時間がより多くなります。
依存関係と前提条件	なし
機能への影響	この構成では、非アクティブタイムアウトに関係なく、アクティブなゲスト HTTP セッションに最大の有効期間が適用されます。設定値は分単位です。値が 0 の場合はアクティブセッションのタイムアウトが無効になります。最大有効期間は、非アクティブタイムアウト *glide.guest.session_timeout* (デフォルトは 30 分) より長くする必要があります。

同時インタラクティブセッションを制限する (セキュリティセンター 1.3 で更新)

インスタンス上のインタラクティブセッションの数を管理します。

このプロパティは、同時セッション制限 (*com.glide.limit.concurrent.sessions*) プラグインで使用するためのものです。プラグインがアクティブで、プロパティが false に設定されている場合、ユーザーはインスタンス上で任意の数の同時インタラクティブセッションを使用することができます。開いているセッションの数が多いほど、セッションハイジャックが発生する可能性が高くなります。

詳細情報

属性	説明
構成名	*glide.authenticate.limit.concurrent.interactive.sessions*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	false
カテゴリ	セッション管理
セキュリティリスク	• 重大度スコア：3.7 • CVSS スコア：低 • セキュリティリスクの詳細：プラグインがアクティブで、プロパティが **false** に設定されている場合、ユーザーはインスタンス上で任意の数の同時インタラクティブセッションを使用することができ、そのためにセッションハイジャックの可能性が高くなります。
依存関係と前提条件	なし
参照	同時インタラクティブセッションを制限する (セキュリティセンター 1.3 で更新)

統合のアクティブなセッションの有効期間を制限する **(Security Center 1.3** の新機能)

glide.integrations.active.session.life_span プロパティは、非アクティブタイムアウトに関係なく、アクティブなゲスト HTTP セッションに最大の有効期間を適用します。設定値は分単位です。値が 0 の場合はアクティブセッションのタイムアウトが無効になります。

詳細情報

属性	説明
構成名	*glide.integrations.active.session.life_span*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	整数
推奨値	0 ～ 720
デフォルト値	0
カテゴリ	セッション管理
セキュリティリスク	• 重大度スコア：4.2 • CVSS スコア：中 • セキュリティリスクの詳細：最大有効期間が長いほど、攻撃者が盗んだセッションをより長く保持できるので、セキュリティインシデントの範囲が拡大し

自動翻訳

属性	説明
	ます。このプロパティは、インスタンスへの低いアクセス権を持つ統合に限定されます。glide プロパティ *glide.integrations.active.session.life_span* の値を 0 より大きく、720 以下に設定します。
依存関係と前提条件	なし

ポリシーベースのセッションアクセスのモバイルリフレッシュトークンの間隔を制限する **(Security Center 1.5** の新機能)

glide.authenticate.session_access.mobile.refresh_token_interval プロパティを使用して、モバイルデバイスユーザーが再認証を強制されるまでに必要な経過時間を制御します。

ユーザーは、アドミンがセッションポリシーで ID プロバイダー属性を構成し (属性はログインごとに異なる場合があります)、ユーザーがシングルサインオン (SSO) を使用して認証する場合にのみ再認証を求められます。デフォルト値は、ユーザーが再認証されるまでの時間 (秒単位) を表します。デフォルト値を大きくすると、セッションハイジャックが発生した場合に、攻撃者がセッションにアクセスできる時間が長くなります。

詳細情報

属性	説明
構成名	*glide.authenticate.session_access.mobile.refresh_token_in*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	整数
推奨値	1800 秒
デフォルト値	1800 秒
カテゴリ	セッション管理
セキュリティリスク	• 重大度スコア：7.3 • CVSS スコア：中 • セキュリティリスクの詳細：インスタンスで ZTA ポリシーが有効になっている場合、モバイルログイン中に SSO を使用しているユーザーは、デフォルト値の 1800 秒 (30 分) が経過すると、強制的にログアウトされ、再ログインする必要があります。より大きい値を使用すると、ユーザーはその経過時間まで待機する必要があります。
依存関係と前提条件	Zero Trust - Policy Based Session Access
機能への影響度	この設定は、認証にシングルサインオンを使用しており、アドミンがセッションアクセスポリシーで ID プロバイダー属性を構成している場合に、ログイン後のユーザーがモバイルデバイスから強制的にログアウトされるまでの時間 (秒単位) を制御します。

UI のアクティブセッションのライフスパンを制限する (Security Center 1.3 の新機能)

glide.ui.active.session.life_span プロパティは、非アクティブタイムアウトに関係なく、アクティブな認証済み HTTP セッションに最大の有効期間を適用します。

アクティブな HTTP セッションの有効期間を短くすることで、潜在的なセキュリティインシデントの範囲を縮小します。**glide.ui.active.session.life_span** システムプロパティは、非アクティブタイムアウトに関係なく、アクティブな HTTP セッションに最大有効期間を適用します。最大有効期間が長いほど、攻撃者が盗んだセッションを長時間使用できるようになり、セキュリティインシデントの範囲が拡大します。デフォルト値の **0** は、アクティブセッションのタイムアウトを無効にします

glide.ui.active.session.life_spanを 1 ～ 720 の値に設定します。この値は、HTTP セッションをアクティブにしておくことができる時間 (分単位) を表します。

注: **glide.ui.active.session.life_span**は UI セッションタイムアウトに制限されます。

詳細情報

属性	説明
構成名	*glide.ui.active.session.life_span*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	整数
推奨値	1 ～ 720
デフォルト値	0
カテゴリ	セッション管理
セキュリティリスク	• 重大度スコア：4.2 • CVSS スコア：中 • セキュリティリスクの詳細：最大有効期間が長いほど、攻撃者が盗んだセッションをより長く保持できるので、セキュリティインシデントの危険性が高まります。
依存関係と前提条件	なし
機能への影響	非アクティブタイムアウトに関係なく、アクティブな認証済み HTTP セッションに最大の有効期間を適用します。設定値は分単位です。値が 0 の場合はアクティブセッションのタイムアウトが無効になります。最大有効期間は、非アクティブタイムアウト glide.ui.session_timeout (デフォルトは 30 分) より長くする必要があります。

高保証セッションのセッション長を制限する

セッションの長さを制限することで、高保証セッションでのアカウント乗っ取りのリスクを低減します

セッションの長さを制限することで、高保証セッションでのアカウント乗っ取りのリスクを軽減します。指定された時間が経過すると、エンドユーザーは再認証する必要があります。

glide.zta.high_assurance.session.timeout システムプロパティを使用して、ユーザーの再認証が必要になるまでの時間を分単位で設定します。このプロパティの値は 1 分から 480 分の間でなけれ

ばなりません。アカウントの乗っ取りのリスクを減らすために、この値を 30 以下に制限することを検討してください。

詳細情報

属性	説明
技術的な構成名	glide.zta.high_assurance.session.timeout
プラグインの適用性	ゼロトラスト:継続的認証 (com.snc.zero_trust_continuous_authentication)
セキュリティリスク	セッションの長さを短くすると、ユーザーに再認証が強制されるため、アカウントの乗っ取りのリスクが軽減されます。
共通脆弱性スコアリングシステム (CVSS) スコア	3.3
共通脆弱性スコアリングシステム (CVSS) 評価	低
機能への影響	ハイアシュアランスセッションのユーザーは、このプロパティで設定された間隔で再認証する必要があります。
依存関係と前提条件	なし
データタイプ	整数
ベースシステム値	30
フォールバック値	30
推奨値	30 以下

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

非アクティブなセッションを積極的に無効化する (Security Center 1.3 の新機能、1.5 および 2.0 で更新)

glide.active.session.timeout.invalidate.session プロパティは、Tomcat サーバーの前にタイムアウトセッションを積極的に無効にするかどうかを制御します。

glide.active.session.timeout.invalidate.session が **true** に設定されていない場合、タイムアウトになったセッションが無効にならない短い期間 (キューサイズに応じて 60 秒以上) が存在する可能性があります。セッションがハイジャックされた場合、攻撃者はこの短い期間にセッションを使用できる可能性があります。このセキュリティリスクを修復するには、*glide.active.session.timeout.invalidate.session* を **true** に設定します。

詳細情報

属性	説明
構成名	*glide.active.session.timeout.invalidate.session*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true

属性	説明
デフォルト値	false
カテゴリ	セッション管理
セキュリティリスク	• 重大度スコア：4.6 • CVSS スコア：中 • セキュリティリスクの詳細： このプロパティを推奨値の true に設定しないと、タイムアウトセッションが無効になる可能性があります。これにより、攻撃者がセッションをハイジャックする可能性が高くなります。
依存関係と前提条件	なし

HTTP セッション識別子をローテーションする

glide.ui.rotate_sessions プロパティを使用して HTTP セッション識別子のローテーションを有効にし、セキュリティの脆弱性を軽減します。

認証後に非認証ユーザーのセッション ID が変更されない場合、Web アプリケーションはセッション固定化攻撃 に対して脆弱になります。悪意のあるユーザーが非認証セッションを開始し、関連するセッション ID を被害者に提供する可能性があります。被害者が認証されると、悪意のあるユーザーがその認証済みセッションを共有するようになります。

詳細情報

属性	説明
プロパティ名	*glide.ui.rotate_sessions*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	セッション管理
目的	より安全なセッション認証を実現すること
推奨値	true
デフォルト値	true
セキュリティリスク評価	8.8
機能への影響	この修正では、ユーザーが非認証ページから認証済みページに移動したときに SessionID が変更されました。 • ユーザーの初回ログイン時に、または何らかの目的でプロキシを使用したり、SessionID をハードコーディングしたりすると、機能に影響を与える可能性があります。 • Single Sign-on 認証に SAML 2.0 プラグインを使用している場合、インスタンスと ID プロバイダーの間で行われるセッション情報の共有が妨げられる可能性があります。そのような場合、このプロパティを false に設定できます。
セキュリティリスク	(中) SessionID は、ブラウザーでセッションステータスを維持することによって、インスタンスユーザーを処理および認証するために使用されます。したがって、SessionID

属性	説明
	は機密データと見なされ、デフォルトで安全である必要があります。セッションローテーションは、ユーザーが非認証ページから認証ページへ移動するたびに SessionID の変更を適用するセキュリティコントロールです。
参照	SAML による認証

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

同時インタラクティブセッションの量を最小化する (Security Center 1.3 で更新)

このプロパティを同時セッション制限プラグインとともに使用して、ユーザーが開くことができるアクティブなセッションの数を制御します。

[Glide 認証：同時インタラクティブセッションの最大数] プロパティを **Limit Concurrent Sessions** (*com.glide.limit.concurrent.sessions*) プラグインと共に使用して、ユーザーに開かれるアクティブなセッションの数を制御します。推奨値は **1** です。これにより、開かれるセッションの数が減ります (数が多いほど、セッションがハイジャックされる可能性が高くなります)。

詳細情報

属性	説明
構成名	*glide.authenticate.max.concurrent.interactive.sessions*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	整数
推奨値	1
デフォルト値	1
カテゴリ	セッション管理
セキュリティリスク	• 重大度スコア：3.7 • CVSS スコア：低 • セキュリティリスクの詳細： プロパティのデフォルト値を 1 より大きく設定すると、セッションハイジャックの可能性が高まります。
依存関係と前提条件	**Concurrent Interactive Sessions** (*com.glide.limit.concurrent.sessions*) プラグインがアクティブになっている必要があります。
参照	同時セッションの制限

セッションアクティビティのタイムアウト時間を最小化する (セキュリティセンター 1.3 で更新)

glide.ui.session_timeout プロパティを使用して、アクティビティのタイムアウト値を分単位で指定します。

このプロパティを設定すると、機能にいくつかの影響が生じます。

- 指定されたセッションタイムアウトが長いほど、セッションの処理中に使用されるメモリ量が多くなります。ベースシステムでは、デフォルトの Apache Tomcat タイムアウト時間の 30 分を使用します。
- ServiceNow AI Platform は [記憶する] によってもユーザーをログアウトします。アプリケーションで 30 分間操作がなかった場合、ログインページの [記憶する] チェックボックスがオンになっていないかぎり、プラットフォームはユーザーを自動的にログアウトします。違いは、再度ログインして続行することがないことです。
- ユーザーのホームページに自動的に更新されるゲージまたはコンテンツがある場合は、このタイムアウト値に一度も達しない可能性があります。

詳細情報

属性	説明
プロパティ名	*glide.ui.session_timeout*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	セッション管理
目的	セッションタイムアウトを適用すること
推奨値	ユーザー指定のタイムアウト (分)。推奨値は 60 分ですが、この値は機能とセキュリティ要件によって異なる場合があります。この値の設定は 1 日を超えないようにしてください。
セキュリティリスク評価	7.5
機能への影響	この修正では、ユーザーアカウントのタイムリーな有効期限が適用されます。機能への影響はありませんが、ユーザーエクスペリエンスは変わります。
セキュリティリスク	(高) 無期限にアクティブであるユーザーセッションはセキュリティリスクです。時間ベースの設定で期限切れにする必要があります。
参照	ユーザーセッションの管理

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

セッションウィンドウのタイムアウト時間を最小化する (セキュリティセンター 1.3 で更新)

glide.ui.user_cookie.life_span_in_days プロパティを使用して、「記憶する」Cookie の有効期限を設定します。デフォルト値は 15 日で、上限は 30 日です。

詳細情報

属性	説明
プロパティ名	*glide.ui.user_cookie.life_span_in_days*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	セッション管理

属性	説明
目的	「記憶する」Cookie のデフォルトの有効期限を有効にすること
データタイプ	整数
推奨値	15
デフォルト値	15
セキュリティリスク評価	4.9
機能への影響	このプロパティは、エンドユーザーがログインページから [記憶する] チェックボックスをオンにして ServiceNow AI Platformにログインすると有効になります。
セキュリティリスク	(中) ユーザーの Cookie が無期限にアクティブになるとセキュリティリスクが発生するため、時間ベースの構成で期限が切れるようにする必要があります。
参照	利用可能なシステムプロパティ 記憶する

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

保存された暗号化

このカテゴリは、保存されたデータの暗号化に焦点を当てています。これには、確立されたアルゴリズムと暗号化モジュールの採用、疑似ランダム値の適切な生成、データ分類に基づく暗号化の実装、キーマテリアルの安全な保存と分離など、いくつかの重要な側面が含まれます。

Glide KMF エンクリプターを有効にする [Security Center 1.3.2 で削除]

インスタンスの Password2 フィールドに使用するエンクリプターを管理します。

glide.kmf.encrypter.enabled プロパティを使用して、KMF エンクリプターを Password2 フィールドのデフォルトのエンクリプターとして設定します。このプロパティにより、従来のエンクリプターではなく、強力で規制に準拠した暗号化標準を確実に使用することができます。従来のエンクリプターの代わりに KMF エンクリプターが使用されるようにするには、このプロパティを **true** に設定します。

詳細情報

属性	説明
構成名	*glide.kmf.encrypter.enabled*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	false
カテゴリ	保存された暗号化

属性	説明
セキュリティリスク	• 重大度スコア：4.9 • CVSS スコア：中 • セキュリティリスクの詳細：このプロパティを **true** に設定すると、従来のエンクリプターの代わりに KMF エンクリプターが使用されます。
参照	Key Management Framework (KMF) による Password2 暗号化

TripleDES/3DES 暗号化アルゴリズムの使用を無効にする

TripleDES/3DES 暗号化アルゴリズムの使用を無効にすることで、古い暗号化方法によるセキュリティリスクを回避します。

機密情報の損失や漏洩につながる可能性があるため、廃止された従来の Triple Data Encryption Standard (3DES または Triple-DES) の使用をインスタンスで無効にします。

米国国立標準技術研究所 (NIST) は、データの暗号化に Triple DES (3DES) を使用しないよう勧告しています。詳細については、NIST 800-131A Rev 2 を参照してください。連邦システムでは、3DES は暗号化が禁止されました。今後、TDES は、古いメッセージの復号化、キーのラップ解除、MAC 検証などの履歴目的にのみ使用できます。

glide.security.3des.encryption.allow システムプロパティは、インスタンスで 3DES 暗号化を有効にするかどうかを制御しますが、インスタンスを廃止する準備ができていることを確認するためにいくつかの手順が必要になる場合があります。このプロパティを設定する前に、廃止プロセスの詳細については KB1704481 を確認してください。

詳細情報

属性	説明
技術的な構成名	glide.security.3des.encryption.allow
プラグインの適用性	なし
セキュリティリスク	3DES のような古くて弱い暗号化を使用すると、機密情報が失われたり漏洩したりする可能性があります。
共通脆弱性スコアリングシステム (CVSS) スコア	4.2
共通脆弱性スコアリングシステム (CVSS) 評価	中
機能への影響	このプロパティを **false** に設定すると、インスタンスで古い脆弱な暗号化の使用を防ぐことができます。これは、Password2 データの保存方法に影響します。3DES の廃止による機能への影響の詳細については 、KB1704481 を確認してください。
依存関係と前提条件	このプロパティの値を変更する前に、 KB1704481 で概説されているタスクを実行します。
データタイプ	ブール
ベースシステム値	偽

属性	説明
フォールバック値	false
推奨値	false

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

3DES キーの使用を防止する (Security Center 7.0 の新機能)

システムプロパティを使用して、インスタンスでの 3DES 静的キーの使用を無効にします。

glide.security.3des.static_keys_usable システムプロパティを使用して、インスタンスでの 3DES 静的キーの使用を無効にします。このプロパティは、スケジュール済みジョブによって自動的に設定される必要があり、そのステータスはプロパティ *glide.security.3des.removal_job_status*で確認できます。このプロパティが **false** に構成されていない場合、スケジュール済みジョブが実行され、ステータスが KEYS_DEACTIVATED になった後、スケジュール済みジョブはこのプロパティが **false** に構成されるまで待機してから、3DES 静的キーの削除を続行します。

*glide.security.3des.static_keys_usable*がシステムプロパティ [sys_properties] テーブルに存在し、値が **false** に設定されていることを確認します。3DES の廃止の詳細については、KB1704481 を確認してください。

詳細情報

属性	説明
構成名	*glide.security.3des.static_keys_usable*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	false
デフォルト値	true
フォールバック値	true
カテゴリ	保存された暗号化
セキュリティリスク	• 重大度スコア：5 • CVSS スコア：中 • 3DES は非推奨であり、一時的な下位互換性以外で 3DES および 3DES 静的キーを使用することはお勧めしません。継続的な使用は、ユーザーが暗号化されたデータへのアクセスを取得した場合、機密情報が開示される状況につながる可能性があります。
機能への影響	**false** に設定すると、3DES 静的キーが引き続き依存しているインスタンスに存在するコードとデータにアクセスできなくなるという、まれな状況が発生する可能性があります。
依存関係と前提条件	なし

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

検証、サニタイズ、およびエンコーディング

検証、サニタイズ、およびエンコーディングは、クロスサイトスクリプティング (XSS)、SQL インジェクション、その他の攻撃などに対する脆弱性を防ぐために入力の検証を実施します。

このコントロールでは、入力の検証と出力のエンコーディングが行われ、出力データのエンコーディングやエスケープなどが正しく構成されていることを確認できます。このカテゴリには、オブジェクトの逆シリアル化や許可リストによる正の検証などのアイテムのチェックも含まれます。

Impact ワークスペースモジュールの説明フィールドで、信頼できるドメインへの **HTML** リンクを許可する **(Security Center 7.0** の新機能)

システムプロパティを使用して、説明フィールドで許可される HTML をサニタイズします。このプロパティは、許可されるリンクを、プロパティにリストされている信頼できるドメインからのリンクのみに制限します。

> **注:** このハードニング設定は、ハードニングベースラインの一部ではありません。セキュリティセンターの強化ページには表示されず、ハードニングスコアに影響します。

Impact ワークスペースモジュールでは、多くの説明関連フィールドで HTML を使用できます。設定すると、 *sn_impact_common.whitelisted.url_HTML_injection* システムプロパティにはドメイン名のカンマ区切りリストが含まれます。Impact ワークスペースモジュールの説明フィールドには、プロパティにリストされているドメインからの URL を持つ HREF のみを含めることができます。

*sn_impact_common.whitelisted.url_HTML_injection*システムプロパティが、Impact ワークスペースモジュールの説明フィールドの HTTP 参照 URL で許可されているドメインを表すドメイン名のカンマ区切りリストに設定されていることを確認します。

これらのフィールドで HREF を禁止するには、プロパティを空の文字列に設定します。プロパティがシステムプロパティ [sys_properties] テーブルに存在しない場合、デフォルトは servicenow.com、service-now.com、youtube.com、google.com、youtu.be、soti.net、dpdhl.sharepoint.com、documentation.avaya.com、w
my.sharepoint.com、scaledagileframework.com のリストになります。

詳細情報

属性	説明
構成名	*sn_impact_common.whitelisted.url_HTML_injection*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	文字列リスト
推奨値	servicenow.com、service-now.com、youtube.com、google.com、youtu.be、soti.net、dpdhl.sha my.sharepoint.com、scaledagileframework.com
デフォルト値	servicenow.com、service-now.com、youtube.com、google.com、youtu.be、soti.net、dpdhl.sha my.sharepoint.com、scaledagileframework.com
フォールバック値	servicenow.com、service-now.com、youtube.com、google.com、youtu.be、soti.net、dpdhl.sha my.sharepoint.com、scaledagileframework.com

属性	説明
カテゴリ	検証、サニタイズ、およびエンコーディング
セキュリティリスク	• 重大度スコア：4.4 • CVSS スコア：中 • 信頼できないドメインがプロパティに追加されると、これらのフィールドが開き、HTML インジェクション攻撃につながる可能性のある危険なソースへのリンクが表示されます。正確なリスクは、顧客インスタンスによって異なります。
機能への影響	プロパティが空の場合、フィールドテキストで HREF を使用できず、すべての HREF が削除されます。プロパティにリストされていないドメインを使用しているリンクはすべて削除されます。このフィールドの値が不適切な場合、影響を受けるフィールドのデータが破損する可能性があります。
依存関係と前提条件	*sn_impact_common.blacklist_tags_HTML_injection* システムプロパティに HREF リンクを囲む HTML タグが含まれている場合、それらのタグ内のすべてのリンクが削除されます。

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

GlideSystemUserSession スクリプト作成可能 API へのアクセスを制限する (セキュリティセンター 1.3 および 2.0 で更新)

クライアント呼び出し可能な GlideSystemUserSessionSandbox スクリプト可能 API は、GlideSystemUserSession の addErrorMessageNoSainitization メソッドと addInfoMessageNoSainitization メソッドを JavaScript サンドボックスに公開しています。これにより、すべてのユーザーがスクリプトを介してこのメソッドをコールできます。

gs.addErrorMessageNoSanitizationMessaging() と *gs.addInfoMessageNoSanitization()* は、スクリプト環境内でログ記録と通知に使用されます。どちらも、このプロパティが推奨値の false に設定されていない場合にサンドボックスで使用できます。サンドボックスは、非認証ユーザーでロールのないユーザーが利用できる、権限の低いスクリプト環境です。これらのメソッドはどちらも、サニタイズされていない入力をユーザーに表示するために使用できます。サニタイズされていない入力には、ユーザーのブラウザで実行される危険なコードが含まれている可能性があるため、サニタイズされていない入力をユーザーに表示することは危険です。これは、従来の反射型XSS攻撃に利用される可能性があります。反射型 XSS 攻撃は、セッションハイジャックを含む複数のシナリオで使用される可能性があります。

glide.sandbox.usersession.allow_unsanitized_messagesシステムプロパティを **false** に設定します。システムプロパティ [sys_properties] テーブルにこのプロパティのレコードがない場合は、作成します。

⚠ **警告:** これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

自動翻訳

詳細情報

属性	説明
プロパティ名	*glide.sandbox.usersession.allow_unsanitized_messages*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	アクセス制御
目的	このプロパティにより、サンドボックス化されたユーザーセッションでの、サニタイズされていない情報メッセージまたはエラーメッセージのコールが制限されます。
タイプ	ブール
推奨値	false
デフォルト値	true
セキュリティリスク評価	8.1
機能への影響	プロパティを値 false に設定すると、これらの関数が呼び出された場合にメッセージが作成されず、ログも記録されません。
セキュリティリスク	(高) 適切にサニタイズされないと、危険性のあるコンテンツにアクセスする可能性があり、サニタイズされていないエラー関数がスクリプトで使用可能になります。
参照	アクセス制御

埋め込み **HTML** で **Javascript** タグを無効にする (セキュリティセンター **1.3** で更新)

glide.ui.security.codetag.allow_script プロパティを使用して、[code] タグを使用して作成された HTML JavaScript コードの埋め込みサポートを無効にします。

ServiceNow AI Platform は、エスケープおよびエンコード技術を実装することで、多くのインジェクション攻撃とクロスサイト攻撃を軽減します。その結果、ユーザーはジャーナルフィールドに対して HTML 形式の入力を書き込んで送信することができなくなります。しかしジャーナルフィールドでは、コードタグで囲まれたテキストを HTML としてレンダリングできます。*glide.ui.security.codetag.allow_script* プロパティが sys_properties テーブルにあり、false に設定されていることを確認します。

- ただし、関連するセキュリティリスクがあります。**true** に設定すると、悪意のあるユーザーは、ジャーナルフィールドをレンダリングした後に、別のクライアントブラウザーで実行できる有害な HTML JavaScript コードを書き込むことができます。
- このプロパティを **false** に設定すると、[code] タグのサポートを無効にすることで、ジャーナルフィールドで HTML JavaScript コードがレンダリングされないようにすることができます。

⚠ **警告:** これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

詳細情報

属性	説明
プロパティ名	*glide.ui.security.codetag.allow_script*
構成タイプ	システムプロパティ (/sys_properties_list.do)

属性	説明
カテゴリ	検証、サニタイズ、およびエンコーディング
目的	クロスサイトスクリプティングと悪意のあるスクリプトの実行からの保護
推奨値	false
デフォルト値	false
セキュリティリスク評価	8.8
機能への影響	この修正では、UI で強制的に JavaScript エスケープが行われ、エンコードされた結果がユーザーに対してレンダリングされます。これは、結果のデータを用いたインスタンスのユーザーインタラクションに基づいて、機能に影響を与える可能性があります。
セキュリティリスク	(高) クロスサイトスクリプティング攻撃から防御するために、アプリケーションで入力検証を行う必要があります。これらの攻撃により、ログインしているブラウザーのコンテキストで、ユーザーセッションで外部スクリプトが実行される可能性があります。攻撃者はこれを使用してセッション情報と機密データを盗むことができます。
参照	ジャーナルフィールドの CODE タグを制限する ジャーナルフィールドエントリを HTML としてレンダリングする 高セキュリティ設定

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

強化された **Java** セキュリティマネージャーを有効にする **(Security Center 1.3** の新機能)

glide.security.manager プロパティには、現在の Java セキュリティマネージャーの Java クラス名が含まれています。

⚠ **警告:** これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

詳細情報

属性	説明
構成名	*glide.security.manager*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	文字列
推奨値	*com.glide.sys.security.ContextualSecurityManager*
デフォルト値	*com.glide.sys.security.ContextualSecurityManager*
カテゴリ	検証、サニタイズ、およびエンコーディング

属性	説明
セキュリティリスク	• 重大度スコア：7.2 • CVSS スコア：高 • セキュリティリスクの詳細：*glide.security.manager* が推奨値の *com.glide.sys.security.ContextualSecurityManager* に設定されていない場合、インスタンスは、予想されるハードニングポリシーのない古い Java セキュリティマネージャーを使用している可能性があります。このハードニングがないと、スクリプト実行アクセス権を持つ攻撃者がインスタンスでリモートからコードを実行する可能性があります。
依存関係と前提条件	なし

HTML サニタイゼーションを強制する (Security Center 1.3 で更新)

com.glide.security.check_unsanitized_html プロパティを使用して、フィールドアサインのためにグローバルレベルで translated_html フィールドのサニタイズ動作を強制します。

HTML は、辞書フィールドに割り当てることができるタイプの一種です。HTML フィールドを任意のフィールドタイプに割り当てると、HTML タグ (<p>、<a href>、<b>、<font>、<img> など) を使用してコンテンツをフォーマットできるようになります。悪意のあるアクティビティを防ぐために、ブロックリストを使用して特定の HTML タグを禁止することができます。このプロパティは、インスタンスの translated_html フィールドで禁止タグが使用されないようにします。

- このプロパティを **enforce** に設定すると、translated_html フィールドのサニタイズ動作が適用されます。
- プロパティを **disable** に設定すると、html のサニタイズがオフになり、translated_html フィールドでブロックされた html タグが許可されます。

⚠ **警告:** これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

詳細情報

属性	説明
プロパティ名	*com.glide.security.check_unsanitized_html*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	検証、サニタイズ、およびエンコーディング
目的	クロスサイトスクリプティングなどの攻撃から保護するために、安全でない HTML タグの使用を防止すること
タイプ	文字列
推奨値	enforce
デフォルト値	enforce
セキュリティリスク評価	7.3

属性	説明
機能への影響	この修正では、UI で強制的に HTML サニタイズが行われ、翻訳された html フィールドがユーザーに対してレンダリングされます。読みやすさと書式設定に影響を与える可能性があります。
セキュリティリスク	(高) クロスサイトスクリプティング攻撃から防御するために、アプリケーションで入力検証を行う必要があります。これらの攻撃により、ログインしているブラウザーのコンテキストで、ユーザーセッションで外部スクリプトが実行される可能性があります。攻撃者はこれを使用してセッション情報と機密データを盗むことができます。
参照	HTML サニタイザー

コンテキスト検索に未検証のリダイレクトが含まれていないことを確認する **(Security Center 7.0** の新機能)

コンテキスト検索の結果に、システムプロパティを使用して現在のドメイン外の参照リンクが含まれないようにします。

コンテキスト検索プラグインでは、 *cxs_new_window* UI ページを使用して、検索結果を新しいウィンドウに表示します。この UI ページには、*sysparm_url* に値を指定することで設定できる紹介リンクが含まれています。*com.snc.contextual_search.cxs_new_window.force_relative_link* システムプロパティが **true** に設定されている場合、*sysparm_url*には現在のドメインに関連するリンクのみを含めることができます。この制限により、UI ページが攻撃者が制御する Web サイトへの未検証のリダイレクトとして使用されるのを防ぐことができます。プロパティが **false** に設定されている場合、 *sysparm_url* 任意の Web サイトにリンクできます。

com.snc.contextual_search.cxs_new_window.force_relative_link プロパティを **[true]** に設定します。システムプロパティ [sys_properties] テーブルにプロパティが存在しない場合、デフォルト値は **false** です。プロパティがテーブルに存在する場合、デフォルトは **true** です。

詳細情報

属性	説明
構成名	*com.snc.contextual_search.cxs_new_window.force_relative_l*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
フォールバック値	false
カテゴリ	検証、サニタイズ、およびエンコーディング
セキュリティリスク	• 重大度スコア：3.1 • CVSS スコア：中

属性	説明
	• **false** に設定すると、*sysparm_url* は任意の Web サイトにリンクできるため、UI ページが攻撃者が制御する Web サイトへの未検証のリダイレクトとして使用される可能性があります。
機能への影響	**true** に設定すると、現在のドメインに関連するリンクのみを含める*sysparm_url*が許可されます。この制限は、UI ページが現在のドメインの Web ページにのみリンクできることを意味します。ただし、UI ページは現在のドメインからの検索結果を表示することを目的としており、現在のドメインにのみリンクする必要があります。
依存関係と前提条件	コンテキスト検索 (com.snc.contextual_search) プラグインがアクティブである必要があります。

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

AJAXEvaluate を無効にする

glide.script.allow.ajaxevaluate を使用して、AJAX 呼び出しによるクライアントスクリプト実行の脆弱性からシステム API を保護します。

プロパティを編集するには、security_admin ロールへの昇格が必要です。

⚠ 警告: これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

詳細情報

属性	説明
プロパティ名	*glide.script.allow.ajaxevaluate*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	検証、サニタイズ、およびエンコーディング
目的	ユーザーがアドミン権限でスクリプトを実行できないようにする。
推奨値	false
デフォルト値	false
構成タイプ	ブーリアン
機能への影響	この修正により、AJAXEvaluate プロセッサが強制的にオフになります。カスタマイズされたスクリプトの一部として AJAX 評価プロセッサーを明示的に使用している場合は、機能に影響を与える可能性があります。
セキュリティリスク	(高)AjaxEvaluator プロセッサーはサンドボックスでクライアントスクリプトを実行しますが、サンドボックス内のアクティビティの範囲を拡張できる追加のプロパティがいくつかあります。

属性	説明
セキュリティリスク評価	7.3
参照	このプロパティは、*glide.script.allow.ajaxevaluate* などのクライアントからのスクリプトの実行を保護および制限するプロパティと同じファミリーに属します。詳細については、「AJAXEvaluate を有効にする」を参照してください。

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

XMLDocument2 ストリーミングパーサー内のエンティティ拡張を無効にする (セキュリティセンター 1.5 で更新)

カスタマイズでエンティティの拡張が必要ない場合は、*glide.stax.allow_entity_resolution* プロパティを使用して外部エンティティの拡張を完全に無効化します。XML は解析を完了しますが、内部または外部エンティティは含まれません。

インスタンスのエンティティ拡張を無効にして、システムファイルの読み取り機能やサービス拒否などの攻撃からインスタンスを保護します。システム プロパティを使用して、ストリーミング パーサー (XMLDocument2) による解析中に XML エンティティが展開されないようにします。

インスタンスでエンティティ拡張を無効にするには、 **glide.stax.allow_entity_resolution** システムプロパティを **false** に設定します。このプロパティがシステムプロパティ [sys_properties] テーブルに表示されない場合、デフォルト値は **true** です。プロパティレコードを作成し、値を **false** に設定して値を変更します。

必須条件

このプロパティを設定する前に、次の手順を実行してください。

- *glide.xml.entity.whitelist.enabled* および *glide.stax.whitelist_enabled* プロパティを true に設定します。詳細については、「XML 外部エンティティを制限する (セキュリティセンター 1.3 および 2.0 で更新)」と「allowlistDisable エンティティ拡張のある XMLdoc2 エンティティ検証を必須とする (セキュリティセンター 1.3 で更新)」を参照してください。
- *glide.xml.entity.whitelist* プロパティでカンマ区切りの FQDN リストを定義してください。XML エンティティ処理を使用してアクセスできるのは、これらの URL のみになります。詳細については、「XML 外部エンティティを制限する (セキュリティセンター 1.3 および 2.0 で更新)」を参照してください。

⚠ **警告:** これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

詳細情報

属性	説明
プロパティ名	*glide.stax.allow_entity_resolution*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	検証、サニタイズ、およびエンコーディング

属性	説明
目的	この修復コントロールは、XML エンティティ拡張/Billion Laugh 攻撃から防御するために有効にする必要があります。
推奨値	false
デフォルト値	true
機能への影響	カスタマイズでエンティティ拡張を使用している場合、ServiceNow AI Platform が以降の処理をブロックする可能性があります。
セキュリティリスク	(重大) 攻撃者はこの脆弱性を利用してデータを指数関数的に拡張し、すべてのシステムリソースを短時間で消費する可能性があります。
ワークアラウンド	カスタマイズでエンティティ拡張が必要な場合は、このプロパティを true に設定し、「allowlistDisable エンティティ拡張のある XMLdoc2 エンティティ検証を必須とする (セキュリティセンター 1.3 で更新)」に記載されている手順を実行してください。

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

OWASp リソースの詳細については、「OWASp」を参照してください。

外部コンテンツ URL を無効にする (Security Center 2.0 で更新)

コネクトチャットを使用して、インスタンスでの外部リンクメタデータの使用方法を管理します。

glide.ui.url.external.content プロパティを使用して、インスタンス内の外部リンクメタデータを管理します。プロパティが推奨値の **false** に設定されている場合、外部リンクメタデータは表示されません。**true** に設定されている場合、コネクトチャット は、YouTube やニュース記事などのソースから外部リンクメタデータを取得し、よりリッチなメッセージが表示されます。これにより、サーバーサイドリクエストフォージェリ (SSRF) 攻撃が発生する可能性があります。

Glide プロパティ *glide.ui.url.external.content* が存在し、値が false に設定されていることを確認します。プロパティがsys_propertiesテーブルに含まれていない場合は、新しいレコードを追加してください。

詳細情報

属性	説明
構成名	*glide.ui.url.external.content*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	false
デフォルト値	true
カテゴリ	検証、サニタイズ、およびエンコーディング
セキュリティリスク	• 重大度スコア：7.2 • CVSS スコア (CVSS score)：高

属性	説明
	• セキュリティリスクの詳細：このプロパティを **true** に設定すると、インスタンスがサーバーサイドリクエストフォージェリ (SSRF) 攻撃にさらされる可能性があります。
依存関係と前提条件	なし
参照	コネクトチャット

ダウンロード可能な **MIME** タイプを制限する (セキュリティセンター **1.3** および **2.0** で更新)

glide.ui.attachment.download_mime_types プロパティは、指定された危険なファイルタイプのリストのファイルを強制的にクライアントにダウンロードし、ブラウザーにインラインで表示しないようにします。

プロパティ *glide.ui.attachment.force_download_all_mime_types* が true に設定されている場合、 *glide.ui.attachment.download_mime_types* プロパティはオーバーライドされ、ブラウザーによってレンダリングされるのではなく、すべての MIME タイプがダウンロードされます。たとえば、text/html をダウンロードすると、HTML ファイルはブラウザーでインライン表示されるのではなく、ファイルとしてクライアントに強制的にダウンロードされ、XSS 攻撃を防ぐことができます。XSS により、admin などの上位ロールへの特権エスカレーションが容易になり、横方向の移動が可能になる可能性があります。

新しい修正:プロパティ *glide.ui.attachment.force_download_all_mime_types* が true に設定されていることを確認します。sys_propertiesテーブルにプロパティが存在しない場合、デフォルト値は false です。

注: プロパティを編集するには、security_admin ロールが必要です。

詳細情報

属性	説明
プロパティ名	*glide.ui.attachment.download_mime_types*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	検証、サニタイズ、およびエンコーディング
目的	ブラウザーで表示できない危険なファイルタイプのリストを適切に管理すると、クロスサイトスクリプティング攻撃 (XSS) を防ぐことができます。
推奨値	適用可能な MIME タイプまたは推奨値のリスト： text/html,image/svg,image/svg+xml,application/xml
デフォルト値	適用可能な MIME タイプのデフォルト値のリスト： text/html,image/svg,image/svg+xml,application/xml
構成タイプ	文字列：アプリケーション MIME タイプのカンマ区切り値。
機能への影響	この修正では、 ServiceNow AI Platform アプリケーションで添付ファイルをクリックしたときに、アクションを実行する前に検証チェックのパフォーマンスが適用されま

属性	説明
	す。潜在的な影響はありませんが、ユーザーエクスペリエンスが変わります。
セキュリティリスク	(中) 攻撃者は MIME タイプを悪用し、意図しないスクリプトコンテンツを被害者側の添付ファイルに配置して機密情報を取得することができます。XSS 機能により、admin などの上位ロールへの権限の昇格が容易になり、横方向の移動を実行しやすくなります。 現在のコンテキストでは、添付ファイル MIME タイプの中でブラウザーでインラインレンダリングしてはいけないものを、カンマ区切りのリストの形式でプロパティに入力します。
セキュリティリスク評価	6.4
関連プロパティ	• *glide.ui.attachment.force_download_all_mime_types* • *glide.ui.attachment.tables_ignore_force_download*
参照	制限されたダウンロード可能な MIME タイプを定義する (セキュリティセンター 1.3、1.5、および 2.0 で更新)。

埋め込み **HTML** コードを無効化する (セキュリティセンター **1.3** で更新)

glide.ui.security.allow_codetag プロパティを使用して、[code] タグを使用して作成された HTML コードの埋め込みサポートを無効にします。

[code] タグを使用して埋め込まれた HTML コードの表示のサポートを無効にします。このタグを使用すると、レンダリングされた HTML をジャーナルフィールドに表示でき、クロスサイトスクリプティング (XSS) 攻撃につながる可能性があります。これらの攻撃により、ログインしているブラウザーのコンテキストで、ユーザーセッションで外部スクリプトが実行される可能性があります。攻撃者はこれらのスクリプトを使用してセッション情報と機密データを盗むことができます。HTML 言語は、スクリプトを書式設定から分離するように設計されていないため、どのシステムでもユーザー制御の HTML を許可することには固有のリスクがあります。

glide.ui.security.codetag.allow_scriptを **false** に設定すると準拠し、このリスクが大幅に軽減されますが、いくつかの小さなリスクが残ります。コードタグのスクリプト部分のみを無効にし、HTML の既知のスクリプトのすべての規則をサニタイズすることに依存します。

glide.ui.security.allow_codetag システムプロパティを **false** に設定すると、ジャーナルフィールドとフォームにレンダリングされた HTML が表示されなくなります。

ServiceNow AI Platform は、エスケープおよびエンコード技術を実装することで、多くのインジェクション攻撃とクロスサイト攻撃を軽減します。その結果、ユーザーはジャーナルフィールドに対して HTML 形式の入力の書き込みや送信ができなくなります。しかしジャーナルフィールドでは、コードタグで囲まれたテキストを HTML としてレンダリングできます。

- ただし、関連するセキュリティリスクがあります。true に設定すると、悪意のあるユーザーは、ジャーナルフィールドをレンダリングした後に、別のクライアントブラウザーで実行できる有害な HTML JS コードを書き込むことができます。
- このプロパティを false に設定すると、[code] タグのサポートを無効にすることで、ジャーナルフィールドで HTML コードがレンダリングされないようにすることができます。

詳細情報

属性	説明
プロパティ名	glide.ui.security.allow_codetag
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	検証、サニタイズ、およびエンコーディング
インスタンスセキュリティセンターでの構成	あり
目的	クロスサイトスクリプティングと悪意のあるスクリプトの実行からの保護
推奨値	false
デフォルト値	true
セキュリティリスク評価	4.2
機能への影響	この修正では、UI で強制的に HTML エンコードが行われ、エンコードされた結果がユーザーに対してレンダリングされます。 このプロパティはデフォルトで [true] に設定されています。このステータスでは、インスタンスはジャーナルフィールドとフォームにレンダリングされた HTML を表示します。 このプロパティを falseに設定すると、HTML が正しくレンダリングされず、フォームのジャーナルフィールドに HTML タグが表示される場合があります。これは、機能に悪影響を及ぼし、結果のデータを用いたユーザーインタラクションに悪影響を与える可能性があります。
セキュリティリスク	(中) クロスサイトスクリプティング攻撃から防御するために、アプリケーションで入力検証を行う必要があります。これらの攻撃により、ログインしているブラウザーのコンテキストで、ユーザーセッションで外部スクリプトが実行される可能性があります。攻撃者はこれを使用してセッション情報と機密データを盗むことができます。

仮想エージェント内で **HTML** サニタイザーを有効にする (セキュリティセンター **1.3** および **1.5** で更新)

com.glide.cs.html.sanitizer.enabled プロパティを使用して HTML SanitizerService を有効にします。

このプロパティは、HtmlSanitizerService が有効かどうかを制御します。*com.glide.cs.html.sanitizer.enabled* が true に設定されていない場合、VA Web クライアントで保存されたクロスサイトスクリプティング (XSS) 攻撃が発生する可能性があります。

詳細情報

⚠ **警告:** これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

属性	説明
プロパティ名	*com.glide.cs.html.sanitizer.enabled*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	検証、サニタイズ、およびエンコーディング
目的	アプリケーションでクロスサイトスクリプティング攻撃および HTML インジェクション攻撃を防ぐこと。
推奨値	true
デフォルト値	true
セキュリティリスク評価	8
機能への影響	この修正では、ユーザーデータがユーザーにレンダリングされる前に、HTML 出力エンコードメカニズムが適用されます。顧客が HTML 属性またはコンテンツデータのレンダリングを伴うカスタマイズを行っている場合は、機能に影響があります。
セキュリティリスク	(高) データがアプリケーションに格納および処理されるときに、ユーザー入力を安全に処理する必要があります。これにより、データの出力エンコードによるクライアント側のクロスサイトスクリプティング攻撃が低減されます。

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

Jelly JS 補間保護を有効にする

glide.ui.jelly.js_interpolation.protect プロパティを使用して、Jelly ページで実行しようとしているすべての JavaScript が Jelly 補間を使用したインジェクションから確実に保護されるようにします。

プロパティを **true** に設定すると、アプリケーションは (ネストされた) Jelly スクリプトツリーを経由します。潜在的に危険な Jelly 式は、次を実行するフィルターでラップされます。

- 結果をエスケープして安全にします。または、
- 安全性を保証できない場合は SecurityException を生成します。これは、評価される予定の式がセキュリティ上の問題を示している可能性があるためです。

⚠ **警告:** これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

詳細情報

属性	説明
プロパティ名	*glide.ui.jelly.js_interpolation.protect*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	検証、サニタイズ、およびエンコーディング
目的	Jelly インジェクションを使用すると発生する可能性がある、悪意のあるコードの実行攻撃を軽減すること

属性	説明
推奨値	true
デフォルト値	false
セキュリティリスク評価	9
機能への影響	このプロパティは、式が引用符で囲まれているかどうかを推測します。正当な式に誤って引用符を付ける可能性があります。その場合、手動で式に安全とマークする必要がある場合があります。
セキュリティリスク	(中) JEXL インジェクションは、ServiceNow AI Platform に固有の入力インジェクションの形式です。これにより、クロスサイト要求の偽造とコード実行の両方が発生する可能性があります。保護を完全にオフにすると、多くの P1 セキュリティ脆弱性がオープンになるおそれがあります。
ワークアラウンド	手動で式に安全とマークするには、SAFE というプリフィックスを Jelly 式に追加します。 `${SAFE:sysparm_input};` すべての式に機械的に SAFE を追加するという方法は、セキュリティの脆弱性をオープンにするおそれがあるため、問題へのアプローチとしては不適切です。 • 式に SAFE を追加するのは、式にクライアントからの入力が含まれていないことが保証できる場合に限ってください。 • 含まれている場合、悪意のあるクライアントによって特権 JavaScript が評価される可能性があります。
参照	Jelly のタグ 高セキュリティ設定

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

入れ子になった式で **Jelly JS** 補間保護を有効にする **(Security Center 2.0** で更新)

インスタンスの補間保護を管理します。

glide.ui.jelly.js_interpolation.protect_nested_expressions プロパティを使用して、補間保護を管理します。補間保護により、JavaScript で Jelly 式を使用する場合、特定のカテゴリに該当するか、式自体で SAFE とマークすることによって、Jelly 式を必ず安全と見なすことができます。この軽減策が有効になっていない場合、攻撃者が GET パラメーターを Jelly ページに送信し、そのパラメーターのコンテンツがアドミン権限を持つサーバー側の JavaScript として評価される可能性があります。このプロパティが推奨値の **true** に設定されていない場合、JavaScript で補間された悪意のある Jelly 式が許可され、ユーザーは Jelly テンプレートを使用してコードを実行できます。

⚠ **警告:** これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

詳細情報

属性	説明
構成名	*glide.ui.jelly.js_interpolation.protect_nested_expressions*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	false
カテゴリ	検証、サニタイズ、およびエンコーディング
セキュリティリスク	• 重大度スコア：9 • CVSS スコア：重大 • セキュリティリスクの詳細：プロパティが **false** に設定されている場合、悪意のある Jelly 式が許可されます。
依存関係と前提条件	なし

相対リンクを強制する (セキュリティセンター **1.3** および **1.5** で更新)

glide.cms.catalog_uri_relative プロパティを使用して、/ess/catalog.do の URL パラメーターからの相対リンクを適用します。

*glide.cms.catalog_uri_relative*プロパティは、/ess/catalog.do の URI パラメーターからの相対リンクを強制します。*glide.cms.catalog_uri_relative* が推奨値の true に設定されていない場合、URL は enforceRelativeURL(url) 関数でサニタイズされません。絶対 URL は、パラメーターまたはフィールド値の一部として使用するとセキュリティリスクを引き起こす可能性があり、ソースページが攻撃者が制御する Web サイトにリダイレクトされます。

詳細情報

属性	説明
プロパティ名	*glide.cms.catalog_uri_relative*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	検証、サニタイズ、およびエンコーディング
目的	外部の未承認コンテンツへのリンクを制限すること
推奨値	true
デフォルト値	false
セキュリティリスク評価	2.6
機能への影響	この修正では、相対 URL のみが許可されるようにカタログページの検証が適用されます。外部 Web アプリケーションへの既存のリンクが切断されます。
セキュリティリスク	(高) 絶対 URL をパラメーターまたはフィールド値の一部として使用すると、セキュリティリスクが生じる可能性が

属性	説明
	あるため、ソースページは攻撃者が制御する Web サイトにリダイレクトされます。

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

URL 許可リストのチェックを強制する (セキュリティセンター 1.3、1.5、および 2.0 で更新)

glide.security.url.whitelist システムプロパティを使用して検証レイヤーを追加し、導入された外部 URL を包含リスト URL に含める必要があるかどうかを確認します。

クライアント側のオープンリダイレクトからユーザーを保護します。これにより、攻撃者はユーザーを信頼できないページや悪意のあるページにリダイレクトできます。

glide.security.url.whitelist.strict_check が推奨値の **true** に設定されていない場合、**glide.security.url.whitelist** システムプロパティが空である限り、すべての外部 URL がリダイレクトに許可されます。**glide.security.url.whitelist** が空でない場合、そのプロパティにリストされている外部 URL のみが許可されます。

オープンリダイレクト攻撃からインスタンスを保護するために、glide.security.url.whitelist.strict_check を **true** に設定するか、許可された外部 URL を使用して **glide.security.url.whitelist** が構成されていることを確認してください。

このプロパティは、次の場合に適用されます。

- /logout.do?sysparm_goto_url={External URL}
- /cms_login_redirect.do?sysparm_goto_url={External URL}

ユーザーがインスタンスからログアウトすると、外部の信頼できるサイトに誘導されます。

- /logout_redirect.do?sysparm_url={External URL}
- /saml_redirector.do?sysparm_uri={External URL}

SAML を有効にすると、ID プロバイダー (IDP) のログアウト URL が呼び出されます。

プロパティ *glide.security.url.whitelist.strict_check* が true に設定されているか、プロパティ *glide.security.url.whitelist* が値に設定されていることを確認します。

詳細情報

属性	説明
プロパティ名	*glide.security.url.whitelist*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	検証、サニタイズ、およびエンコーディング
目的	ログイン、ログアウト、またはその他のリダイレクト時に安全な URL リダイレクトを実装すること。このプロパティは、「未検証のリダイレクトと転送」と呼ばれる OWASP 上位 10 件の攻撃の 1 つを軽減します。
タイプ	文字列
デフォルト値	true

属性	説明
推奨値	true
値	組織の承認済み URL [何らかの定義された FQDN (完全修飾ドメイン名) 例：http://www.servicenow.com]
セキュリティリスク評価	6.3
機能への影響	この修正では、ログアウトページで検証が適用されます。SSO/SAML 構成のインスタンスのユーザーに機能的な影響を与える可能性があります。
セキュリティリスク	(高) クライアント側のオープンリダイレクトにより、攻撃者は、攻撃者が制御する Web サイトに被害者/ユーザーをリダイレクトできるようになります。これはセキュリティリスクと見なされます。
参照	マルチ SSO (SAML 2.0) のエラーと修正

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

Excel 計算式をエスケープ (Security Center 1.3 で更新)

glide.export.escape_formulas プロパティを使用して、Excel のインジェクション (式のインジェクションとも呼ばれる) を防止します。

Excel などのプログラム内の数式をエスケープすることで、ファイルをエクスポートして開いた後に、悪意のある可能性のある数式が実行されないようにします。Excel インジェクションは、Web サイトが Excel ファイル内に信頼できないエントリを埋め込むときに発生します。Microsoft Excel や LibreOffice Call などのスプレッドシートアプリケーションを使用してファイルを開く場合、+、-、=、または @ で始まるセルは、適切にエスケープされない限り、式として解釈されます。悪意のある数式は、スプレッドシートに機密情報が含まれていない場合でも、コード実行を通じて閲覧者のコンピューターを侵害するために使用される可能性があるため、リスクをもたらします。

glide.export.escape_formulas システムプロパティを **true** に設定して、これらの数式が実行されないようにします。

詳細情報

属性	説明
プロパティ名	*glide.export.escape_formulas*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	検証、サニタイズ、およびエンコーディング
目的	アプリケーションで Excel インジェクションまたは式のインジェクションを防止すること
推奨値	true
デフォルト値	false
セキュリティリスク評価	6.4
機能への影響	悪意を持って細工された数式は、スプレッドシートソフトウェアの脆弱性を悪用してユーザーのコンピューターを乗っ取るために使用される可能性があります。

属性	説明
セキュリティリスク	(中) 悪意のある式は、閲覧者のコンピューターの侵害に使用される可能性があるため、埋め込みスプレッドシートに機密情報が含まれていない場合であってもリスクを引き起こします。
ワークアラウンド	代替手段として、可能な場合はすべての末尾の空白を削除し、クライアントが提供するすべてのデータを英数字のみに制限することを検討してください。
参照	利用可能なシステムプロパティ

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

リストビューでの **HTML** をエスケープ (セキュリティセンター **1.3** および **1.5** で更新)

glide.ui.escape_html_list_field プロパティを使用して、リストビューの HTML フィールドで強制的に HTML をエスケープします。

リストビューの HTML フィールドに HTML が表示されないようにするには、*glide.ui.escape_html_list_field*を **true** に設定します。HTML のサニタイズをプラットフォーム全体に (システムプロパティを介して) またはフィールドごとに (スキーマ属性を介して) 非アクティブのままにすると、XSS スタイルの攻撃につながる可能性があります。XSS 攻撃により、権限の低いユーザーが権限の高いユーザーのセッションを乗っ取ったり、リダイレクトや改ざんなどの標準的な Web アプリケーションの動作を妨害したりする可能性があります。

⚠ **警告:** これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

詳細情報

属性	説明
プロパティ名	*glide.ui.escape_html_list_field*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	検証、サニタイズ、およびエンコーディング
目的	アプリケーションでクロスサイトスクリプティング攻撃を防ぐこと
推奨値	true
デフォルト値	true
セキュリティリスク評価	3.1
機能への影響	この修正では、UI で強制的に HTML パーサーレベルで HTML エンコードが行われ、エンコードされた結果がユーザーに対してレンダリングされます。これは、結果のデータを用いたインスタンスのユーザーインタラクションに基づいて、機能に影響を与える可能性があります。
セキュリティリスク	(高) クロスサイトスクリプティング攻撃から防御するために、アプリケーションで入力検証を行う必要があります。これらの攻撃により、ログインしているブラウザーのコンテキストで、ユーザーセッションで外部スクリプトが実

属性	説明
	行される可能性があります。攻撃者はこれを使用してセッション情報と機密データを盗むことができます。
参照	高セキュリティ設定

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

JavaScript をエスケープ (Security Center 1.3 で更新)

glide.html.escape_script プロパティを使用して、リストビュー中に HTML フィールドの JavaScript (<script></script>) タグからのエスケープを適用します。

glide プロパティ *glide.html.escape_script* は、HTML フィールドをサニタイズするのに役立ちます。*glide.html.escape_script* が推奨値の true に設定されていない場合、埋め込み JavaScript を削除することで、バックエンド Java コンテキストからの HTML フィールド (出力エンコーディング) の入力はサニタイズされません。HTML フィールドの Javascript は、XSS を保存して反映させる可能性があります。XSS 機能により、admin などの上位ロールへの特権エスカレーションが容易になり、横方向の移動が容易になる可能性があります。

⚠ **警告:** これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

詳細情報

属性	説明
プロパティ名	*glide.html.escape_script*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	検証、サニタイズ、およびエンコーディング
目的	アプリケーションに対するクロスサイトスクリプティング攻撃を防ぐこと
推奨値	true
デフォルト値	true
セキュリティリスク評価	8.8
機能への影響	この修正では、UI で強制的に JavaScript エスケープが行われ、エンコードされた結果がユーザーに対してレンダリングされます。これは、結果のデータを用いたインスタンスのユーザーインタラクションに基づいて、機能に影響を与える可能性があります。
セキュリティリスク	(高) クロスサイトスクリプティング攻撃から防御するために、アプリケーションで入力検証を行う必要があります。これらの攻撃により、ログインしているブラウザーのコンテキストで、ユーザーセッションで外部スクリプトが実行される可能性があります。攻撃者はこれを使用してセッション情報と機密データを盗むことができます。
参照	利用可能なシステムプロパティ

属性	説明
	高セキュリティ設定

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

Jelly スクリプトをエスケープ (セキュリティセンター 1.3 および 1.5 で更新)

glide.ui.escape_all_script プロパティを使用して、Jelly に挿入されたすべてのスクリプトを強制的にエスケープします。

このプロパティは、<j:jelly> ... に含まれるすべての JS 文字列と HTML 文字列をエスケープします</j:jelly> を実行して、いくつかの XSS の問題が発生するのを防ぎます。*glide.ui.escape_all_script* が推奨値の true に設定されていない場合、Jelly に挿入されたスクリプトのエスケープは無効になります。この軽減策がなければ、プラットフォームはさまざまなスクリプトインジェクション攻撃に対して広くオープンになります。攻撃者はインスタンス上で任意の Rhino スクリプトを実行する可能性があります。

⚠ 警告: これらのタグを使用する場合は注意してください。ここにユーザー入力が表示されると、セキュリティの脆弱性が開かれる可能性があります。

詳細情報

属性	説明
プロパティ名	*glide.ui.escape_all_script*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブール
カテゴリ	検証、サニタイズ、およびエンコーディング
目的	プロパティが **true** に設定されていない場合、開発者は XSS の問題を防ぐために、各カスタム Jelly スクリプトでいくつかの手順を実行する必要があります。これらの手順には、Web ページにレンダリングするために出力ストリームに送信される Jelly 変数の特定と、次の各タグのエスケープが含まれます。 $â {JS:expression} $â {HTML:expression} OR $â {JS,HTML:expression}
推奨値	true
デフォルト値	true
セキュリティリスク評価	7.3
機能への影響	この修正では、パーサーレベルで Jelly エスケープが適用されます。結果のデータを用いたユーザーインタラクションに機能的な影響を与える可能性があります。

属性	説明
セキュリティリスク	(高) アプリケーションで入力されるすべてのユーザー入力で入力検証を行う必要があります。これにより、プラットフォームに対するインジェクション攻撃を防御および保護できます。
ワークアラウンド	Web ページでのレンダリング用に設計された一部のスクリプトと HTML タグが破損したように見えるため、UI が影響を受ける可能性があります。この修正では、出力エンコードされたページをブラウザーに送信してレンダリングします。 たとえば、`<u>` タグが適切にエスケープされると、「`my string here`」ではなく「`<u> my string here</u>`」と表示されます。この場合、エスケープを防ぐには、NOESC プリフィックスを Jelly 式に追加して JS エスケープを防止します。例： • 変更前：($[jvar_context_menus]); • 変更後：($[NOESC:jvar_context_menus]); • 変更前：$[jvar_ui_policy_scripts] • 変更後：$[NOESC:jvar_ui_policy_scripts] ⚠ **警告:** これらのタグを使用する場合は注意してください。ここにユーザー入力が表示されると、セキュリティの脆弱性が開かれる可能性があります。
参照	高セキュリティ設定 Jelly のタグ

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

Scratchpad のスクリプトをエスケープ (セキュリティセンター 1.3 で更新)

スクラッチパッドがインスタンスのセキュリティ体制にどのように影響するか、また悪意のあるスクリプトがそこで実行されないように管理する方法について説明します。

スクラッチパッドは、ブラウザでアクセスできるサーバー上の情報を簡単に設定する方法です。アドミニストレーターは、任意のレコードを含むスクラッチパッド上のあらゆるものをスクリプト化できます。このプロパティが推奨値の **true** に設定されていない場合、クロスサイトスクリプティングの脆弱性など、悪意のあるスクリプトが実行される可能性があります。

詳細情報

属性	説明
構成名	*glide.ui.escape_scratchpad*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true

属性	説明
デフォルト値	true
カテゴリ	検証、サニタイズ、およびエンコーディング
セキュリティリスク	• 重大度スコア：6.5 • CVSS スコア：中 • セキュリティリスクの詳細：このプロパティが推奨値の **true** に設定されていない場合、クロスサイトスクリプティングの脆弱性など、悪意のあるスクリプトが実行される可能性があります。
依存関係と前提条件	なし
参照	ワークフロー管理

XML マークアップをエスケープ (セキュリティセンター 1.3 での更新)

glide.ui.escape_text プロパティを使用して、パーサーレベルで XML 値を強制的にエスケープしてから、クライアントのブラウザに送信します。

クロスサイトスクリプティングは、攻撃者が悪意のある JavaScript をエントリポイントに挿入するときに発生します。プラットフォーム/アプリケーションは、被害者のブラウザに送信して実行する前に悪意のある JavaScript をエスケープすることができません。このコンテキストでのエスケープとは、次のことを意味します。

- **&** --> &
- **<** --> <
- **>** --> >
- **"** --> "
- **'** --> '
- **/** --> /

例:<![CDATA[<script>alert('XSS Attack');]]>

エスケープ：<script>alert('XSS Attack');</script>

glide.ui.escape_text プロパティが sys_properties テーブルにあり、true に設定されていることを確認します。

警告: これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

詳細情報

属性	説明
プロパティ名	*glide.ui.escape_text*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	検証、サニタイズ、およびエンコーディング

属性	説明
目的	XML をエスケープすることで、信頼できないデータに埋め込まれた悪意のある JavaScript をブラウザが解析して JavaScript として実行することを確実になくします。 • 悪意のあるユーザーが XSS 攻撃を試みて、他のユーザーのセッションをハイジャクするか、ユーザーを悪意のある Web サイトにリダイレクトする可能性があります。 • NOW Platform には cookie を保護するためのコードが含まれていますが、それをエスケープするにはこのプロパティが **true** に設定されている必要があります。
推奨値	true
セキュリティリスク評価	8.8
機能への影響	この修正では、UI の XML パーサーレベルで XML エンコードが適用されます。これにより、エンコードされた結果がユーザーに対してレンダリングされるので、結果のデータを用いたインスタンスのユーザーインタラクションに基づいて、機能に影響を与える可能性があります。
セキュリティリスク	(高) クロスサイトスクリプティング攻撃から防御するために、アプリケーションで入力検証を行う必要があります。これらの攻撃により、ログインしているブラウザのコンテキストで、ユーザーセッションで外部スクリプトが実行される可能性があります。攻撃者はこれを使用してセッション情報と機密データを盗むことができます。

ワークアラウンド	このプロパティを **true** に設定すると、カタログアイテムの説明またはカタログアイテム変数のヘルプテキストの HTML タグでレンダリングが停止します。フィールドによっては、HTML 形式を使用できない場合があります。
	ただし、*glide.ui.escape_text* プロパティをオフにすると、出力エンコーダーにより、すべての JEXL 式にプリフィックスが付加されます。
	`${JS:expression}`
	`${HTML:expression}`
	または
	`${JS,HTML:expression}`

XML 応答のエスケープ

インスタンスでの XML エスケープの処理方法を管理します。

このプロパティを使用して、XML 応答をエスケープするかどうかを管理します。プロパティが推奨値の **false** に設定されている場合、XML 応答はエスケープされず、XML インジェクション攻撃につながる可能性があります。意図されていない XML コンテンツが XML メッセージに挿入されると、アプリケーションの意図したロジックが変更される可能性があります。

詳細情報

属性	説明
構成名	*glide.soaprequest.unescape_xml_response*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	false
デフォルト値	false
カテゴリ	検証、サニタイズ、およびエンコーディング
セキュリティリスク	• 重大度スコア：6.4 • CVSS スコア：中 • セキュリティリスクの詳細：このプロパティを **false** に設定すると、XML エスケープが無効になり、XML インジェクション攻撃につながる可能性があります。
依存関係と前提条件	なし

HTML サニタイザーを有効にする (セキュリティセンター 1.3 で更新)

glide.html.sanitize_all_fields プロパティを使用して、HTMLSainitizer スクリプトインクルードを有効にし、スクリプトで構成された除外リストおよび包含リストの属性に基づいて HTML 入力をサニタイズします。

辞書/フィールドで利用可能なフィールドタイプには、HTML や翻訳された HTML などがあります。これらの HTML 入力フィールドを使用すると、次のような HTML 形式の入力を記述できます。

<img>、<a href …>、<iframe>などの最も基本的な HTML タグを使用して<h1>テスト</h1>します。

これにより、悪意のある攻撃者が次のような HTML タグを使用して悪意のあるベクトルを挿入する可能性があります。

[<IMG SRC=" &#14; JavaScript:alert('XSS');">][<IMG onmouseover="alert('xss')">],[a href="" onclick=alert(/xss/)]

詳細情報

属性	説明
プロパティ名	*glide.html.sanitize_all_fields*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	検証、サニタイズ、およびエンコーディング
目的	アプリケーションでクロスサイトスクリプティング攻撃および HTML インジェクション攻撃を防ぐこと
推奨値	true

属性	説明
デフォルト値	true
セキュリティリスク評価	8.8
機能への影響	この修正では、ユーザーデータがユーザーにレンダリングされる前に、HTML 出力エンコードメカニズムが適用されます。顧客が HTML 属性またはコンテンツデータのレンダリングを伴うカスタマイズを行っている場合は、機能に影響があります。
セキュリティリスク	(高) データがアプリケーションに格納および処理されるときに、ユーザー入力を安全に処理する必要があります。これにより、データの出力エンコードによるクライアント側のクロスサイトスクリプティング攻撃が低減されます。
ワークアラウンド	このプロパティは、システム内のすべての HTML フィールドをサニタイズします。個々のフィールドで HTML のサニタイズを有効にする必要がある場合は、「個々のフィールドでサニタイズを有効にする」を参照してください。 包含リストまたは除外リストを設定して、組織のポリシーに従って HTML タグと属性をサニタイズすることもできます。
参照	HTML サニタイザーの有効化 HTML サニタイザー

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

許可された **Java** パッケージを制限する (セキュリティセンター **1.3** で更新)

これらのプロパティを設定すると、危険な API がスクリプトエンジンに公開されるのを防ぐことができます。

システムテーブルを設定し、それに応じて推奨されるプラグインをインストールします。

sys_whitelist_member および *sys_whitelist_package* テーブルが空の値でない場合、危険な API がスクリプトエンジンに公開される可能性があります。ServiceNow セキュリティチームによって承認されていない Java 名前空間に対応するエントリ。

パッケージコール削除ツールをインストールします。詳細については、「パッケージコール削除ツール」を参照してください。

これらのテーブルを編集するには、カスタマーサービス & サポートにお問い合わせください。

詳細情報

属性	説明
テーブル、プラグイン名	テーブル：

属性	説明
	• *sys_whitelist_member* • *sys_whitelist_package* プラグイン：com.glide.script.packages_call_removal
構成タイプ	表形式の構成、プラグイン
カテゴリ	検証、サニタイズ、およびエンコーディング
目的	危険な API がスクリプトエンジンに公開されるのを防ぐ。
推奨値	空
デフォルト値	なし。これはテーブル構成であり、Glide プロパティではないため、デフォルト値はありません。
構成タイプ	テーブルリスト、プラグイン
セキュリティリスク	(高) 危険な API がスクリプトエンジンに公開されるのを防ぎます。これらのサポートされている API は、インスタンス内を不安定にし、セキュリティ上の問題をもたらす可能性があります。
セキュリティリスク評価	8.2

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

パッケージコール削除ツール

Packages Call Removal Tool (*com.glide.script.packages_call_removal*) プラグインをアクティブ化して実行し、提案されたそれぞれの変更を完了するか却下するかを検討します。

Packages Call Removal Tool は、次のようなプラグインです。

- ServiceNow AI Platform Java クラスに対するパッケージコールのスクリプトをスキャンします。
- 優先する GlideScriptable 名に置き換える変更を提案します。
- スクリプトを容易に変更できるようにします。

注: このレコードがベースシステムレコードの場合、ツールからの推奨事項を使用すると、アイテムは customer_update としてマークされます。ただし、このツールを使用すると、Packages,xxx のコールにフラグが設定されるので今もなお便利です。

このパッケージコール削除ツールは、sa_mapping_ext_commands および sa_custom_operation で使用されるいくつかのパッケージコールを報告する場合があります。これらのパッケージコールは MID サーバーに属します。クラスがないため、コードは MID サーバーで実行されます。[エラー] セクションで次に列記するパッケージコールを見つけた場合は、[却下] (無視) としてマークします。ツールは、そのパッケージコールを再度報告しません。

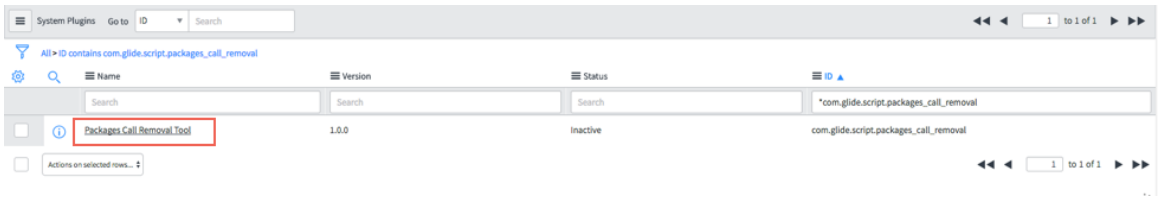
- Packages.com.snc.sw.util.JSONUtil.toJSONPlain(file_content);
- Packages.com.snc.sw.util.JSONUtil.toJSONPlain(file_name);
- Packages.com.snc.sw.commands.HttpCallHandler;
- Packages.com.snc.sw.dto.ProviderType.SSH

詳細情報

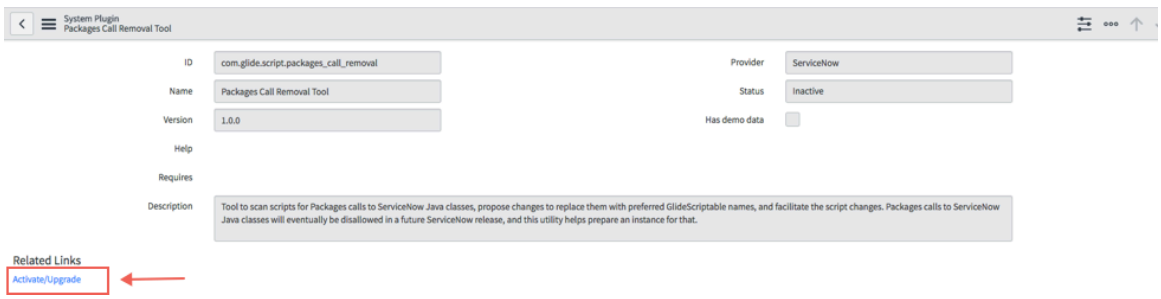
属性	説明
プラグイン名	com.glide.script.packages_call_removal
構成タイプ	[システム定義] > [プラグイン]
目的	承認されたデータアクセスのみを許可する Glide 許容 (GlideScriptable) 名の未承認パッケージ/メンバーコールを削除すること
推奨値	有効
機能への影響	この修正により、パッケージコールが `GlideScriptable` API に置き換えられ、パッケージコールを含むカスタマイズに影響を与える可能性があります。このツールは、実際にはパッケージコールを自動的に置き換えるわけではありません。代わりに、packages_call_item テーブルに保存される提案が提供されます。その後、アドミニストレーターは提案された変更を承認するか却下するかを決定できます。
セキュリティリスク	(中) サーバー上でのデータ取得またはオブジェクトアクセスという結果をもたらすクライアント側 API 呼び出しは、セキュリティの観点から危険であると見なされます。機密オブジェクトへのアクセスを許可および制限するために、検証が必要になります。

設定手順

1. 移動先 システム定義 > プラグイン

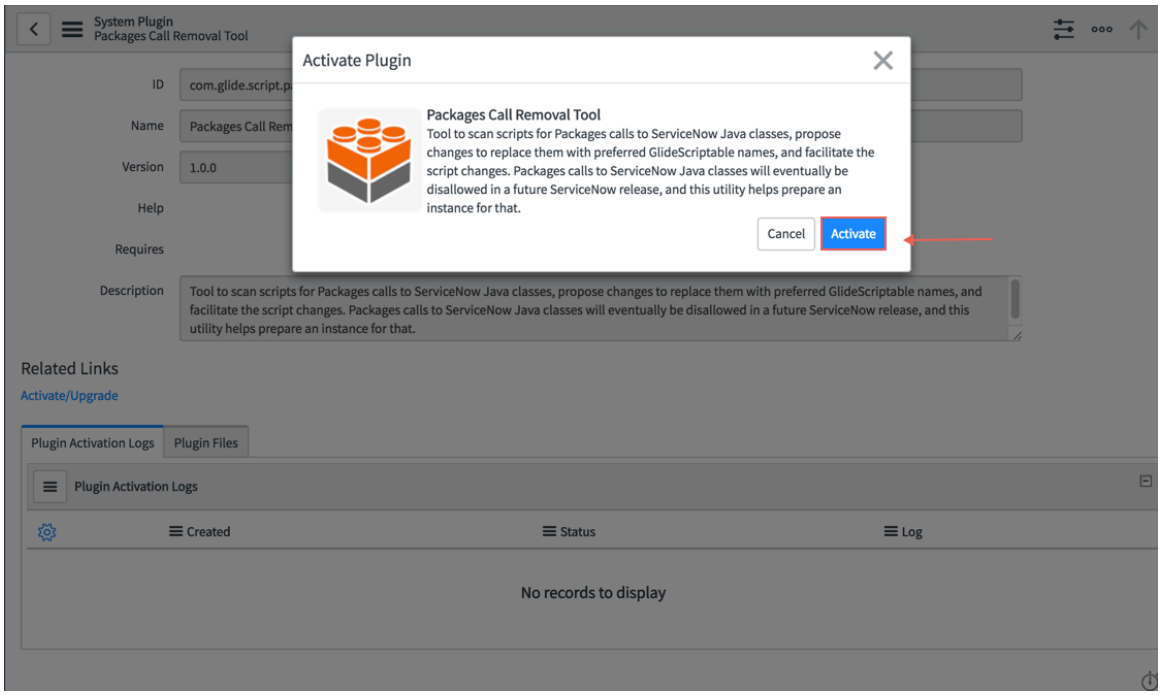


2. プラグイン ID = `com.glide.script.packages_call_removal` を検索します。



3. [アクティブ化/アップグレード **(Activate/Upgrade)]** をクリックしてプラグインをアクティブ化します。

自動翻訳



4. 包含リストパッケージコールと包含リストメンバーコールを確認するには、 許可された Java パッケージを制限する (セキュリティセンター 1.3 で更新)の「設定手順」セクションで概説されているアクションを完了します。

LDAP の初期識別名の設定を解除する (Security Center 1.3 で更新、2.0 で削除)

このプロパティを使用して、LDAP サーバーレコードの識別名を管理します。

このプロパティは、すぐに利用可能な (OOB) 修正スクリプトの実行時に挿入される LDAP サーバーレコードの識別名をコントロールします。推奨値の "" または空白に設定されている場合は、低い権限のユーザーが LDAP サーバーデータを列挙できます。

詳細情報

属性	説明
構成名	*glide.ldap.initial.dn*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	文字列
推奨値	空
デフォルト値	空
カテゴリ	検証、サニタイズ、およびエンコーディング
セキュリティリスク	• 重大度スコア：2.7 • CVSS スコア：低 • セキュリティリスクの詳細：プロパティ値を "" または空白に設定すると、権限の低いユーザーが LDAP サーバーデータにアクセスできる可能性があります。

属性	説明
依存関係と前提条件	なし

厳格なセッション **Cookie** のセキュリティを強制する (セキュリティセンター **1.3** で更新)

glide.ui.secure_cookies プロパティを使用して、適切にフォーマットされた Cookie を要求します。

プロパティが true に設定されていると、関連付けられた Cookie が想定される形式でない場合、インスタンスはセッションを拒否します。

⚠ **警告:** これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

詳細情報

属性	説明
プロパティ名	*glide.ui.secure_cookies*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	検証、サニタイズ、およびエンコーディング
目的	より安全なセッション認証を実現すること
推奨値	true
デフォルト値	true
セキュリティリスク評価	8.8
機能への影響	プロパティが true に設定されている場合、不適切な形式の Cookie は拒否されます。このような Cookie が却下された場合、ユーザーは再度ログインする必要があります。

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

GlideXMLUtil スクリプト可能項目のエンティティ拡張しきい値を最小化する (セキュリティセンター 1.3、1.5、および **2.0** で更新)

glide.xmlutil.max_entity_expansion プロパティを使用して、エンティティ拡張の上限を小さい数値に変更します。

このプロパティは、XML パーサー内のエンティティ展開の最大量を制御します。*glide.xmlutil.max_entity_expansion* が推奨値の 3000 以下に設定されていない場合、GlideXMLUtil 解析スクリプト可能はサービス拒否攻撃に対して脆弱になる可能性があります。

プロパティ *glide.xmlutil.max_entity_expansion* が 3000 以下に設定されていることを確認します。インスタンスが Washington 以降にある場合、sys_propertiesレコードが存在しない場合、デフォルトの暗黙的な値は 3000 です。インスタンスが Washington 以降にない場合、インスタンスアドミンは、名前が *glide.xmlutil.max_entity_expansion* 、値が 3000 の sys_propertiesレコードを作成することをお勧めします。

ℹ **注:** ServiceNow AI Platform によって設定されるデフォルトの最小値は 500 で、これは安全なしきい値と見なされています。

詳細情報

属性	説明
プロパティ名	*glide.xmlutil.max_entity_expansion*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	検証、サニタイズ、およびエンコーディング
目的	この修復コントロールは、XML エンティティ拡張/Billion Laugh 攻撃から防御するために有効にする必要があります。
推奨値	3,000
デフォルト値	100000
セキュリティリスク評価	5.3
機能への影響	カスタマイズで大規模なエンティティ拡張を使用している場合、 ServiceNow AI Platform によって以降の処理がブロックされる可能性があります。
セキュリティリスク	(中) 攻撃者はこの脆弱性を利用してデータを指数関数的に拡張し、すべてのシステムリソースを短時間で消費する可能性があります。

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

空の **ACL** の作成を防止 **(Security Center 2.0** の新機能)

無効な ACL の作成、更新、または保存の試行をブロックするには、*glide.security.empty_acl.popup_window.enabled* プロパティを安全な値 true に設定します。この設定により、ACL のロールまたはセキュリティ属性を構成するためのクライアント側モデルも提供されます。

*glide.security.empty_acl.popup_window.enabled*プロパティは、アクセス制御リスト (ACL) をフォームベースで編集するユーザー (特にsys_security_acl) が、無効なデータ条件、スクリプト、セキュリティ属性、またはロールリストを含む無効な ACL を作成、更新、または保存できるかどうかを決定します。それ以外の場合は、未構成 (空の ACL) のままになります。Xanadu リリースの時点で、空の ACL はアクセスを拒否します。Xanadu より前のServiceNowバージョンでは、空の ACL で無条件のアクセスが許可されます。

*glide.security.empty_acl.popup_window.enabled*プロパティがセキュアな値 true に設定されている場合、無効または空の ACL の作成、更新、または保存の試行をブロックし、ACL のロールまたはセキュリティ属性を構成するためのクライアント側モデルを提供します。プロパティが安全でない値 false に設定されている場合、そのような試行は許可され、クライアント側のモデルは表示されません。

注: このプロパティは大文字と小文字を区別します。たとえば、値 True (大文字の「T」) は false と評価されます。さらに、このプロパティは、High Security (com.glide.high_security) プラグインがインストールされていてアクティブな場合にのみ機能します。

詳細情報

属性	説明
構成名	*glide.security.empty_acl.popup_window.enabled*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	文字列
推奨値	true
デフォルト値	true
カテゴリ	検証、サニタイズ、およびエンコーディング
セキュリティリスク	• 重大度スコア：6.5 • CVSS スコア：中 • セキュリティリスクの詳細:このプロパティが true に設定されている場合、空の ACL 警告ポップアップにより、ユーザーはクライアント側で空の ACL を送信できなくなります。false に設定すると、ポップアップは表示されなくなります。
依存関係と前提条件	なし
機能への影響	このプロパティを使用すると、ユーザーは空の ACL 警告ポップアップのオンとオフを切り替えることができます。
参照	空の ACL の作成を防止 (Security Center 2.0 の新機能)

制限されたダウンロード可能な **MIME** タイプを定義する (セキュリティセンター 1.3、1.5、および **2.0** で更新)

glide.ui.attachment.force_download_all_mime_types プロパティを使用して MIME タイプをダウンロードし、ブラウザーでインラインレンダリングしません。

text/html、image/svg、image/svg+xml、application/xml などの危険な MIME タイプが *glide.ui.attachment.download_mime_types* に含まれている場合、危険なファイルがブラウザーでインラインでレンダリングされ、クロスサイトスクリプティング攻撃 (XSS) につながる可能性があります。このプロパティは、カンマ区切りの添付 MIME タイプのリストであり、ブラウザにインラインでは表示されません。たとえば、text/html を含めると、HTML ファイルはブラウザでインライン表示されず、添付ファイルとしてクライアントに強制的にダウンロードされます。このリストを適切に管理することで、クロスサイトスクリプティング攻撃を防ぐことができます。

glide.ui.attachment.download_mime_types システムプロパティに「text/html、image/svg、image/svg+xml、application/xml」などの危険な MIME タイプが含まれていない場合、危険なファイルがブラウザーにインラインで表示される可能性があります。これにより、クロスサイトスクリプティング (XSS) 攻撃が発生する可能性があります。このチェックは、*glide.ui.attachment.force_download_all_mime_types*が **false** に設定されている場合にのみ関連します。

このプロパティは、添付ファイルの MIME タイプをカンマで区切って列挙したもので、ブラウザーではインラインレンダリングされません。たとえば、 text/html を含めると、HTML ファイルはブラウザでインライン表示されず、添付ファイルとしてクライアントに強制的にダウンロードされます。

*glide.ui.attachment.force_download_all_mime_types*が **false** に設定されている場合は、*glide.ui.attachment.download_mime_types* システムプロパティに危険な MIME タイプ text/html、image/svg、image/svg+xml、application/xml が含まれていることを確認します。

詳細情報

属性	説明
構成名	*glide.ui.attachment.force_download_all_mime_types*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	文字列 (MIME タイプのカンマ区切りリスト)
推奨値	text/html,image/svg,image/svg+xml,application/xml
デフォルト値	text/html,image/svg,image/svg+xml,application/xml
フォールバック値	text/html,image/svg,image/svg+xml,application/xml
カテゴリ	検証、サニタイズ、およびエンコーディング
セキュリティリスク	• 重大度スコア：6.3 • CVSS スコア：中 • セキュリティリスク:このリストを適切に管理することで、クロスサイトスクリプティング攻撃を防ぐことができます。
依存関係と前提条件	このチェックは、*glide.ui.attachment.force_download_all_mime_types* が **false** に設定されている場合、またはシステムプロパティ [sys_properties] テーブルに存在しない場合にのみ関連します。

アップロードされる **MIME** タイプを制限する (セキュリティセンター **1.3** および **2.0** で更新)

glide.security.file.mime_type.validation プロパティを使用して、アップロードする MIME タイプのチェックを有効にします。添付ファイルの MIME タイプ検証を有効 (プロパティを **true** に設定) または無効 (**false** に設定) にすることができます。

必須条件

このプロパティを設定する前に *glide.attachment.extensions* プロパティを設定してください。*glide.attachment.extensions* で指定された拡張子のみがアップロード時に MIME タイプをチェックされます。詳細については、「ファイル拡張子を制限する」を参照してください。

Glide プロパティ *glide.ui.jelly.js_interpolation.protect_nested_expressions* が存在し、値が true に設定されていることを確認します。プロパティがsys_propertiesテーブルに含まれていない場合は、新しいレコードを追加してください。

詳細情報

属性	説明
プロパティ名	*glide.security.file.mime_type.validation*

属性	説明
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	検証、サニタイズ、およびエンコーディング
目的	ファイルのアップロード中に MIME タイプ/マジックバイトのチェックを適用すること。
推奨値	true
デフォルト値	true
セキュリティリスク評価	5.4
機能への影響	この修正により、アプリケーションの添付ファイルの MIME タイプの検証が有効になります。この検証は MIME タイプとデータ間の不一致をチェックするだけであるため、ファイルのアップロードに悪意がある場合を除き、機能への影響はありません。
セキュリティリスク	(中) ファイルの包含や悪意があるファイルのアップロードなどの脆弱性を低減するには、MIME タイプ検証を有効にする必要があります。
参照	添付ファイルの管理

強化のためのプロパティの設定の詳細については、「ハードニング設定」を参照してください。

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

XML 外部エンティティを制限する (セキュリティセンター 1.3 および 2.0 で更新)

XML 外部エンティティ (XXE) 攻撃を防ぐために、 *glide.xml.entity.whitelist* プロパティと *glide.xml.entity.whitelist.enabled* プロパティが推奨値に設定されていることを確認してください。

許可リストを使用して XXE 攻撃から保護し、サーバーが実行する可能性のある任意の HTTP 要求を攻撃者が含めるのを防ぎます。これにより、サーバーと他のエンティティとの信頼関係を利用した追加の攻撃につながる可能性があります。

glide.xml.entity.whitelist システムプロパティの値に http://java.sun.com/j2ee/dtds/ を追加し、**glide.xml.entity.whitelist.enabled** システムプロパティを **true** に設定します。

http://java.sun.com/j2ee/dtds/ 以外の値は、**glide.xml.entity.whitelist** プロパティに含めることができますが、初期設定のプラットフォーム状態には不要です。追加の値を確認して、安全かどうかを判断します。

⚠ 警告: これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

詳細情報

属性	説明
プロパティ名	*glide.xml.entity.whitelist,glide.xml.entity.whitelist.ena*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	検証、サニタイズ、およびエンコーディング

属性	説明
目的	この修復コントロールは、XXE 攻撃から防御するために有効にする必要があります。
推奨値	http://java.sun.com/j2ee/dtds/
デフォルト値	http://java.sun.com/j2ee/dtds/
セキュリティリスク評価	9.8
機能への影響	カスタマイズで *glide.xml.entity.whitelist* プロパティの包含リストではなく外部エンティティを使用している場合、NOW Platform が以降の処理をブロックする可能性があります。
セキュリティリスク	(重大) 攻撃者は DTD を使用して、サーバーが実行する可能性のある任意の HTTP 要求を含めることができます。この場合、サーバーと他のエンティティとの信頼関係を利用する他の攻撃につながる可能性があります。

allowlistDisable エンティティ拡張のある **XMLdoc2** エンティティ検証を必須とする (セキュリティセンター **1.3** で更新)

カスタマイズでエンティティの拡張が必要ない場合は、*glide.xmlutil.max_entity_expansion* プロパティを使用して外部エンティティの拡張を完全に無効化します。XML は解析を完了しますが、内部または外部エンティティは含まれません。

Glide プロパティ *glide.stax.whitelist_enabled* がシステムプロパティ [sys_properties] テーブルに存在しない場合、または推奨値の **true** に設定されていない場合、Glide プロパティ *glide.stax.allow_entity_resolution* の値が **true** に設定されていれば、すべての外部エンティティが許可されます。

カスタマイズでエンティティの拡張が必要ない場合は、*glide.stax.allow_entity_resolution* プロパティを使用して外部エンティティの拡張を完全に無効化します。XML は解析を完了しますが、内部または外部エンティティは含まれません。

- *glide.stax.allow_entity_resolution*を **true** に設定すると、すべての外部エンティティは、*glide.stax.whitelist_enabled* プロパティの設定に従って、対象エンティティの解決または拡張を試みます。
- *glide.stax.allow_entity_resolution*を **false** に設定すると、すべてのエンティティの解決と拡張がブロックされます。このプロパティの詳細については、「XMLDocument2 ストリーミングパーサー内のエンティティ拡張を無効にする (セキュリティセンター 1.5 で更新)」を参照してください。

glide.stax.whitelist_enabledが **true** に設定されている場合は、**glide.xml.entity.whitelist** プロパティでカンマ区切りの FQDN のリストを定義します。XML エンティティ処理プロパティを使用して到達できるのは、これらの URL のみです。詳細については、「XML 外部エンティティを制限する (セキュリティセンター 1.3 および 2.0 で更新)」を参照してください。攻撃者はこの脆弱性を利用して、外部エンティティ拡張 (XXE) 攻撃でデータを指数関数的に拡張し、すべてのシステムリソースを短時間で消費する可能性があります。

必須条件

このプロパティを設定する前に、次の手順を実行してください。

- *glide.xml.entity.whitelist.enabled* プロパティと *glide.stax.whitelist_enabled* プロパティを **true** に設定します。詳細については、「XML 外部エンティティを制限する (セキュリティセンター 1.3 および 2.0 で更新)」を参照してください。
- *glide.xml.entity.whitelist* プロパティでカンマ区切りの FQDN リストを定義してください。XML エンティティ処理プロパティを使用してアクセスできるのは、これらの URL のみになります。詳細については、「XML 外部エンティティを制限する (セキュリティセンター 1.3 および 2.0 で更新)」を参照してください。

⚠ **警告:** これはセーフハーバープロパティです。つまり、いったん変更したら変えることはできません。元に戻すことはできません。

詳細情報

属性	説明
プロパティ名	*glide.stax.whitelist_enabled*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	検証、サニタイズ、およびエンコーディング
目的	この修復コントロールは、XML エンティティ拡張/Billion Laugh 攻撃から防御するために有効にする必要があります。
推奨値	true
デフォルト値	false
セキュリティリスク評価	9.8
機能への影響	カスタマイズでエンティティ拡張を使用している場合、ServiceNow AI Platform が以降の処理をブロックする可能性があります。
セキュリティリスク	攻撃者はこの脆弱性を利用して、外部エンティティ拡張 (XXE)攻撃でデータを指数関数的に拡張し、すべてのシステムリソースを短時間で消費する可能性があります。
ワークアラウンド	カスタマイズでエンティティ拡張が必要な場合は、このプロパティを true に設定し、「XML 外部エンティティを制限する (セキュリティセンター 1.3 および 2.0 で更新)」に記載されている手順を実行してください。

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

OWASp リソースの詳細については、「OWASp」を参照してください。

すべての翻訳済み **HTML** フィールドをサニタイズ **(Security Center 2.0** の新機能)

glide.translated_html.sanitize_all_fields プロパティを安全な値に設定して、すべての *translated_html* 要素が HTML サニタイザーでサニタイズされるようにする方法について説明します。

glide.translated_html.sanitize_all_fields プロパティが true に設定されている場合、すべての*translated_html*要素が HTML サニタイザーでサニタイズされます。プロパティが false に設定されている場合、辞書属性 *html_sanitize* が true に設定されている要素のみがサニ

タイズされます。このサニタイズは、クロスサイトスクリプティング (XSS) 攻撃につながる可能性のある悪意のあるコンテンツを攻撃者が埋め込むのを防ぐのに役立ちます。

詳細情報

属性	説明
構成名	*glide.translated_html.sanitize_all_fields*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン
推奨値	true
デフォルト値	true
カテゴリ	検証、サニタイズ、およびエンコーディング
セキュリティリスク	• 重大度スコア：4.6 • CVSS スコア：中 • セキュリティリスクの詳細:このプロパティを安全な値である true に設定しないと、すべての HTML 要素がサニタイズされないため、攻撃者がフィールドに悪意のあるコンテンツを埋め込む可能性が高くなります。
依存関係と前提条件	なし
機能への影響	ウィジェットが公開に設定され、プロパティの値に含まれている場合に、顧客がテーブル情報にアクセスできるようにします。

Impact ワークスペースモジュールの説明フィールドの **HTML** をサニタイズ **(Security Center 7.0** の新機能)

sn_impact_common.blacklist_tags_HTML_injection プロパティで HTML インジェクション攻撃のソースである HTML タグを削除して、説明フィールドの HTML をサニタイズします。

Impact ワークスペースモジュールでは、次の説明フィールドで HTML を使用できます。

- sn_impact_common_capabilities_mapテーブルとsn_impact_common_par_version_phase_app_mappingテーブルのcustomer_notesフィールド。
- sn_impact_common_manual_capability_descriptionテーブルのmanual_descriptionフィールド。

このシステムプロパティに HTML タグ (スクリプトなど) のカンマ区切りリストが含まれている場合、それらのタグとそのコンテンツは、リストされたフィールドの HTML 部分から削除されます。これらのタグを削除すると、HTML インジェクション攻撃のソースである HTML タグが削除され、説明フィールドの HTML がサニタイズされます。このプロパティがシステムプロパティ [sys_properties] テーブルで設定されていない場合、値はデフォルトで拒否された HTML タグのデフォルトリストになります。プロパティが空の場合、すべての HTML タグが許可されます。

Impact ワークスペースモジュールの説明フィールドから削除される HTML タグのカンマ区切りリストを指定する *sn_impact_common.blacklist_tags_HTML_injection* を使用します。この削除は、HTML インジェクション攻撃を防ぐのに役立ちます。少なくとも、このリストにはデフォルトリストの内容が含まれている必要があります。システムプ

ロパティ [sys_properties] テーブルでプロパティが設定されていない場合、デフォルトは script、iframe、object、embed、form、onerror、onload、style、img、video、audio、source、button のリストになります。

詳細情報

属性	説明
プロパティ名	*sn_impact_common.blacklist_tags_HTML_injection*
構成タイプ	システムプロパティ (/sys_properties_list.do)
カテゴリ	検証、サニタイズ、およびエンコーディング
目的	HTML インジェクション攻撃のソースである HTML タグを削除して、説明フィールドの HTML をサニタイズします。
推奨値	少なくとも、script、iframe、object、embed、form、onerror、onload、style、img のデフォルト値
デフォルト値	script,iframe,object,embed,form,onerror,onload,style,img,video,audio,sourc
セキュリティリスク評価	4.4
機能への影響	HTML タグをデフォルトのリストに追加すると、説明フィールドに必要な HTML 機能が制限される場合があります。正確な影響は、お客様のインスタンスによって異なります。
セキュリティリスク	(中)
参照	高セキュリティ設定

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

SVG ファイルに安全コンテンツセキュリティポリシーを設定する (Security Center 1.3 の新機能)

com.glide.csp.self_script_src_svg プロパティは、翻訳メモリインデックス (IIX) ファイル拡張子を介して Scalable Vector Graphics (SVG) にアクセスするときに、HTTP Content-Security-Policy ヘッダーに **script-src none** ディレクティブを追加します。

com.glide.csp.self_script_src_svg プロパティは、クロスサイトスクリプティング (XSS) 攻撃を格納した悪意のある添付ファイルがインスタンスで実行されるのを防ぎます。このポリシーがないと、攻撃者がユーザーをだまして Web ブラウザで任意の JavaScript コードを実行させ、それがデータ流出やセッション乗っ取りなどのセキュリティの脆弱性につながる可能性があります。

詳細情報

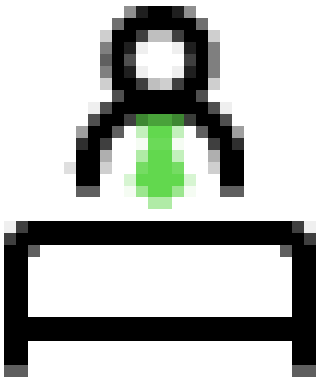
属性	説明
構成名	*com.glide.csp.self_script_src_svg*
構成タイプ	システムプロパティ (/sys_properties_list.do)
データタイプ	ブーリアン

属性	説明
推奨値	true
デフォルト値	true
カテゴリ	検証、サニタイズ、およびエンコーディング
セキュリティリスク	• 重大度スコア：7.1 • CVSS スコア：高 • セキュリティリスクの詳細：このプロパティを推奨値の true に設定しないと、ユーザーが攻撃者からの任意の JavaScript コードを実行する可能性があります。
依存関係と前提条件	なし
機能への影響度	このプロパティは、Scalable Vector Graphics (SVG) ファイルが外部スクリプトにアクセスするのを防ぎます。

ログエクスポートサービス (LES)

ログエクスポートサービス (LES) を使用すると、インスタンスシステムおよびアプリケーションログをエンタープライズセキュリティ分析ツールにシームレスにエクスポートできます。

開始するには

探索	構成	管理
		
ログエクスポートサービス (LES) の詳細	Kafka および MID サーバーコンシューマー用の LES の構成	LES の管理

使用方法	参照	
 LES を使用したログレポートのレビュー	 アプリに関するその他の情報について学ぶ	

トラブルシューティングとサポート

- に関する質問または回答 ログエクスポートサービス (LES) の ServiceNow コミュニティ
- 既知のエラーポータルで既知のエラー記事を検索する
- 保持および除外するテーブルを含む LES のクローン作成に関する推奨事項については、「 LES 構成を保持するために保持するテーブル 」を参照してください。

ログエクスポートサービスの詳細 (LES)

LES サービスは、セットアップと保守が容易な分析ツールとの非常にスケーラブルでほぼリアルタイムの統合を提供します。LES を初めて使用する場合は、この概要セクションを読んで、このツールで何ができるかを学んでください。

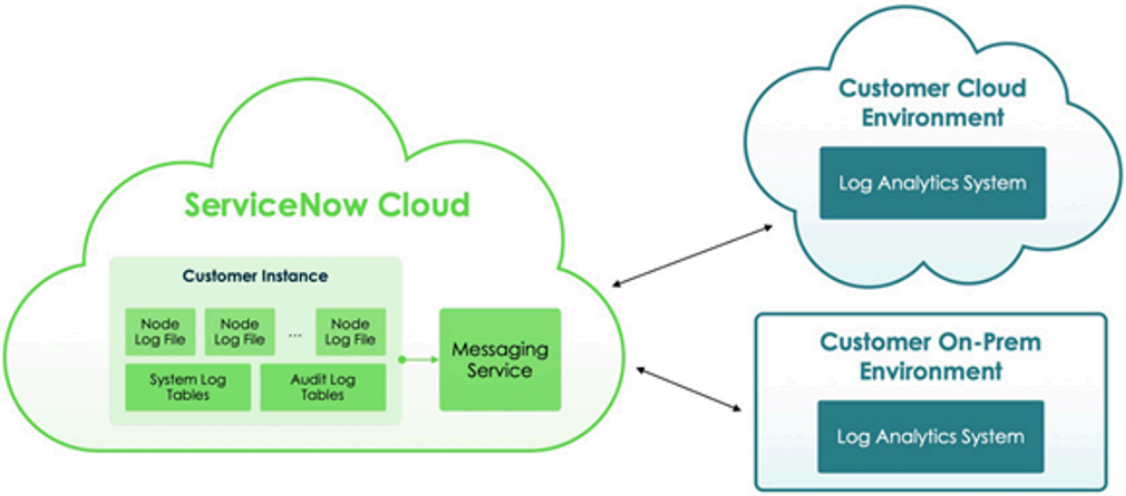
エンタイトルメントを確認して、 ログエクスポートサービスにアクセスできるかどうかを確認してください。

ログエクスポートサービスの概要

統合ツールを使用すると、分析ソリューションを活用して以下を実行できます。

- ServiceNow のセキュリティ上の脅威の検出とセキュリティインシデントの分析
- ServiceNow アプリのパフォーマンスのトラブルシューティングと最適化
- ServiceNow ユーザーエクスペリエンスの監視と最適化

LES は、Hermes メッセージングサービスと呼ばれる ServiceNow AI Platform 機能を活用します。これは、Apache Kafka 上に構築されたマルチテナント、マルチクラスター、データ転送、およびキューイングサービスであり、インスタンスが大量の Kafka イベントを生成および消費できるようにします。Apache Kafka は、組織内のビジネスシステム間でデータを交換するための単一の統合ポイントを提供するオープンソースのデータストリーミングプラットフォームです。



LES は、生成されたログイベントのコピーを Hermes メッセージングサービス に転送します。

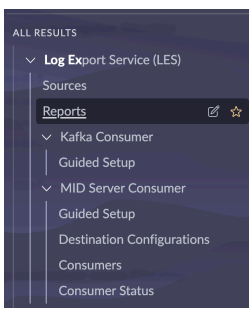
Hermes メッセージングサービス は、インスタンスで大量の Kafka イベントを生成および使用可能にする、Apache Kafka にビルドされたマルチテナント、マルチクラスター、データトランスポート、およびキューイングサービスです。Hermes メッセージングサービスは、Apache Kafka 向けストリームコネクト、ログエクスポートサービス (LES)、およびインスタンスデータレプリケーション (IDR) (IDR) の一部として利用可能な ServiceNow AI Platform の機能です。

クラウドまたはオンプレミスの外部ログアナリティクスシステムは、Hermes メッセージングサービス のログイベントを使用できます。LES には、ログを使用するための 3 つの接続オプションがあります。

- 専用 MID サーバー：オンプレミスまたはクラウドにインストールされた専用の MID サーバーは、Hermes メッセージングサービス に自動的に接続され、そこから継続的にログイベントを取得し、REST 接続を介してログアナリティクスツールにプッシュします。
- ログアナリティクスソリューション (Splunk など) から Kafka コネクタを活用する：選択したログアナリティクス製品の Kafka コネクタは、オンプレミスまたはクラウドにインストールされ、Hermes メッセージングサービス に自動的に接続され、そこから継続的にログイベントを取得し、ログアナリティクスツールにプッシュします。
- Kafka システムから直接：Kafka システムは Hermes メッセージングサービス に直接接続し、ネイティブの Kafka プロトコルコマンドと接続を使用して、そこからログイベントをプルします。

注: Kafka メッセージが構成されたメモリバッファーを超えると、構成した合計メモリバッファーより大きいことを示すエラーが Hermes から返されることがあります。

LES を構成および管理するには、ServiceNow Store から LES をインストールする必要があります。LES アプリケーションでは、サービスのインストールに役立つガイド付きセットアップ、サービス (ログソース、コンシューマー、および宛先) を構成するためのページ、およびログの作成と使用を把握するためのレポートが用意されます。



注: 新しいソース構成を作成することもできます。詳細については、「ログソース構成の作成」を参照してください。

ログエクスポートサービス ユーザー

ログエクスポートサービス には次のユーザーがいます。

ユーザー	説明
アプリケーションアドミニストレーター [sn_logstoanalytics.admin]	このロールは LES アプリケーションとともにインストールされ、アドミン以外のロールがアプリケーションを使用できるようにします。
システムアドミニストレーター [admin]	LES ストアアプリケーションのセットアップには、admin ロールが必要です。

ログエクスポートサービス のメリット

メリット	機能	ユーザー
ログにフィルターを設定するためのログソース構成を作成します	ログソース構成の作成	アプリケーション管理者
Kafka コンシューマー向けのエクスペリエンスガイド付きセットアップ	Kafka コンシューマー向けのガイド付きセットアップ	システムアドミニストレーター
MID サーバーコンシューマー向けのガイド付きセットアップエクスペリエンス	MID サーバーコンシューマー向けのガイド付きセットアップ	システムアドミニストレーター
ログレポートダッシュボードを調べて、各データログのサイズを分析します	ログレポートのレビュー	システムアドミニストレーターまたはアプリケーションアドミニストレーター

次に探索する内容

ログエクスポートサービスの使用の詳細については、以下を参照してください。

- ログエクスポートサービスの管理 (LES)
- ログエクスポートサービスの構成 (LES)
- ログエクスポートサービス (LES) の使用
- ログエクスポートサービス (LES) の参照

ログソース

ログエクスポートサービス (LES) は、一部のシステムログテーブル、監査テーブル、およびアプリケーションノードログファイルからログソースをエクスポートできます。

LES でエクスポートできるログソースは次のとおりです。

- システムログテーブル
  - syslog テーブル：インスタンスプロセス、レコード、およびサーバーマシンのメモリ使用率などの重要でないイベントの警告とエラーを表示
  - syslog_transaction テーブル：インスタンスのすべてのブラウザアクティビティを表示
  - sys_outbound_http_log テーブル：REST や SOAP などの送信 Web サービスのすべての要求と応答を表示
- 監査テーブル：sys_audit テーブルビューを使用して、監査対象として選択したテーブルでの変更を記録
- アプリケーションノードのログファイル：localhost ログファイルを使用して、アプリケーションノードのエラーを表示インスタンスには複数のノードがあり、各ノードには複数のログファイルがあります。

上記のログソースのスキーマと目的の詳細については、「システムログ」を参照してください。

ログエクスポートサービスの管理 (LES)

LES を使用して、ソースタイプごとにログソース構成とマルチトピックを作成します。

ログソース構成の作成

ログソース構成を作成して、転送するログのフィルターを規制および設定します。

始める前に
必要なロール：admin または sn_logstoanalytics.admin

手順

1. 移動先 すべて > ログエクスポートサービス.
   ソース構成のリストが表示されます。
2. 新しいソース構成を作成する場合は、[新規] を選択します。
   既存のソース構成を選択して変更することもできます。
   [ソース] フォームが表示されます。
3. フォームのフィールドに入力します。

ソースフォーム

フィールド	説明
ソースタイプ	ログソースのタイプ ○ ノードログ ○ テーブル 詳細については、「ログソース」を参照してください。
ログレベル	ログ記録出力を制御するために使用できる一連の標準ログ記録レベル。規則に従って、各レベルは同等以上の重大度のログを転送します。

フィールド	説明
	注: このフィールドは、次のいずれかの条件が満たされた場合にのみ表示されます。 ○ ソースタイプとして [ノードログ] を選択した場合 ○ ソースタイプとして [テーブル] を選択し、テーブルが syslog である場合
受け入れ	ログが Hermes に転送される形式を指定します。JSON またはプレーンテキストとして送信できます。
テーブル	テーブルタイプログのテーブルの選択。 注: このフィールドは、[ソースタイプ] として [テーブル] を選択した場合にのみ表示されます。
フィルタタイプ	ログを選択的に転送する条件。 注: このフィールドは、syslog または sys_audit のいずれかをテーブルとして選択した場合にのみ表示されます。

4. [ソーストピック] 関連リストの詳細を確認します。
各ログテーブルを確認し、独自のトピックを作成できます。

   注: この関連リストは、[フィルタータイプ] フィールドで [ログテーブル] を選択した場合にのみ表示されます。

   トピック名は自動入力されず、独自のトピックを選択または作成できます。

5. ログテーブルのトピックを選択します。

   ログテーブルのトピックは、既存のトピックを選択することも、新しいトピックを作成することもできます。

   a. [新規] を選択して、ログテーブルの新しいソーストピックを作成します。[ソーストピック] フォームが表示されます。
   b. [ログテーブル] フィールドで必要なテーブルを選択します。
   c. [トピック] フィールドでルックアップアイコンを選択します。

      注: リストから既存のトピックを選択できます。[Kafka トピック] リストで [新規] を選択して、新しいトピックを作成することもできます。新しい Kafka トピックの作成 LES ソーステーブルでのソースタイプとマルチトピックの作成 を参照してください。

   d. [ソーストピック] フォームで [送信] を選択します。

6. [ソーストピック] 関連リストで、最近作成されたログテーブルとそれに対応するトピックを表示します。
7. 新しいソース構成を作成するには、[送信] を選択します。

LES ソーステーブルでのソースタイプとマルチトピックの作成

ソースタイプごとに複数のトピックを作成して、ソースタイプごとにログを消費します。デバッグプロセス中に、他のログテーブルに影響を与えることなく、さまざまなログソースの特定のトピックの選択をカスタマイズするオプションを利用できるようになりました。

始める前に

注: ログソースを削除しても、sys_kafka_topicテーブル内の対応するトピックは削除されません。ログソースを再度作成すると、既存のトピックを再利用できるため、継続性が確保され、不要なトピックの再作成を回避できます。

必要なロール：admin または sn_logstoanalytics.admin

手順

1. 移動先 すべて > ログエクスポートサービス **(LES)** > ソース.
2. [新規] をクリックして、新しいソースを作成します。
   [ソース] フォームが表示されます。以前は、ソースタイプが選択されると [トピック] フィールドが自動的に入力されていました。Yokohama 以降、各ソースタイプが独自のトピックを作成できるようになりました。デフォルトでは、 ソースタイプ または テーブル 名を選択しても入力されません。これらは、Kafka トピックから直接作成することも、参照フィールドから作成することもできます。

   注: sys_auditテーブルには複数のログテーブルがあるため、[ソース] リストの [トピック] 列に特定のトピックは表示されません。以前は、すべてのソースタイプに同じトピックがありました。ソースごとに異なるトピックを選択できるようになりました。

3. ルックアップアイコンを選択して、Kafka トピックのリストに移動します。
   Kafka トピックのリストが表示されます。
4. オプション: 次の手順で、新しい Kafka トピックを作成します。

   注: このステップは、選択したソースタイプに新しいトピックを作成する場合にのみ適用されます。

   a. 選択したソースタイプの新しい Kafka トピックを作成するには、[新規] を選択します。Kafka トピックフォームが表示されます。
   b. フォームのフィールドに入力します。

   ソースフォーム

フィールド	説明
名前	作成しているトピックの名前
アプリケーション ID	sn_logstoanalyticsを入力
名前空間	デフォルトの名前空間を入力
パーティション	Hermes のトピックのパーティションフィールドは、トピックのデータが分割されるパーティションを参照します。スケーラビリティと並列処理において重要な役割を果たします

   c. [送信] を選択して、新しい Kafka トピックを作成します。
5. 特定のトピックに対して作成する ソースタイプ を選択します。

6. 選択した ソースタイプのリストから必要なトピックを選択します。
7. [送信] を選択して、特定の選択でソースタイプを作成します。
   ソースタイプが、選択したトピック名やその他の情報とともに [ソース] リストに表示されます。

ログエクスポートサービスの構成 (LES)

ガイド付きセットアップを使用して LES の初期構成を段階的に実行します。ガイド付きセットアップは、製品のロールアウトを計画し、本番稼働するための基本構成を実行するために役立ちます。

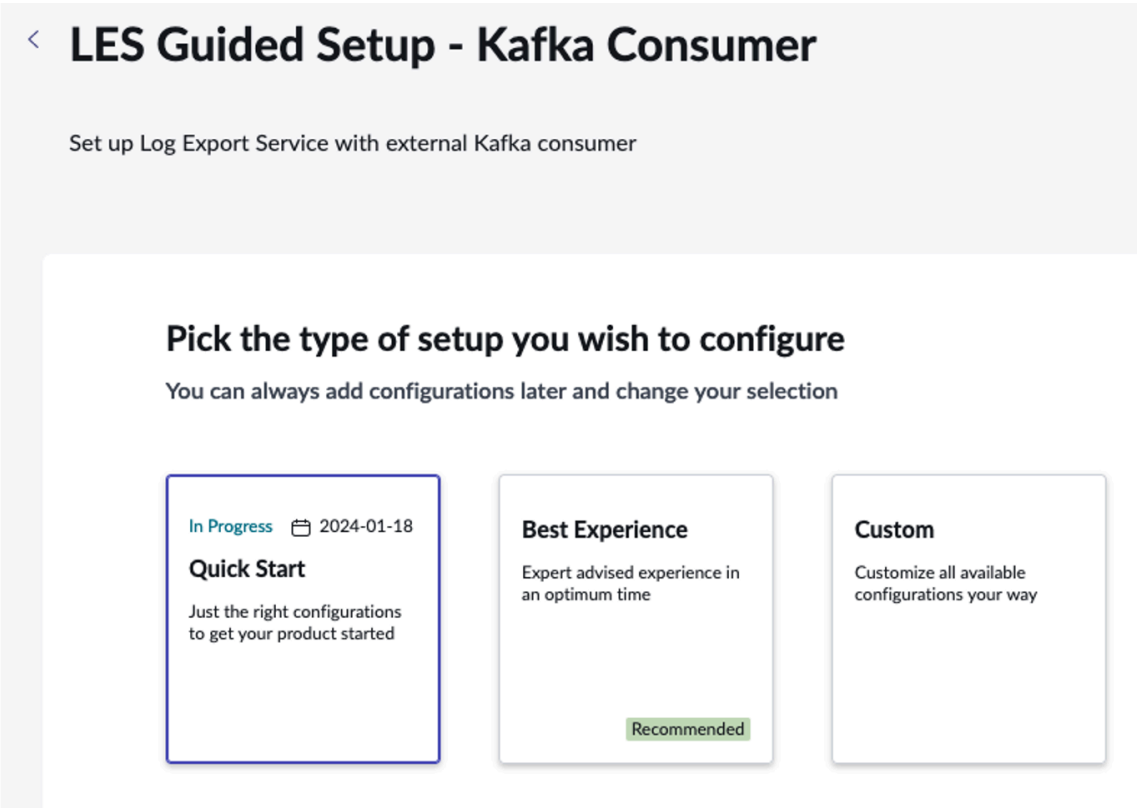
ガイド付きセットアップでは、構成アクティビティがカテゴリに整理されます。各カテゴリには、計画ガイダンス、セットアップ前の手順、役立つコンテンツへのリンクなどの情報が表示されます。カテゴリでは、構成を実行するインスタンス内のページへの一連のリンクも提供しています。ガイド付きセットアップのプロセスで完了した内容が追跡されるため、設定を中断しても、その時点から再開することができます。

Kafka コンシューマー

ガイド付きセットアップを使用して、Kafka コンシューマー向けの LES の初期構成を段階的に実行します。

ガイド付きセットアップのホームページ

ガイド付きセットアップのホームページには、ガイド付きセットアップの構成タイプの概要が含まれています。ガイド付きセットアップのタイプを選択し、[続行] を選択すると、ガイド付きセットアップ手順が開き、構成を開始できます。

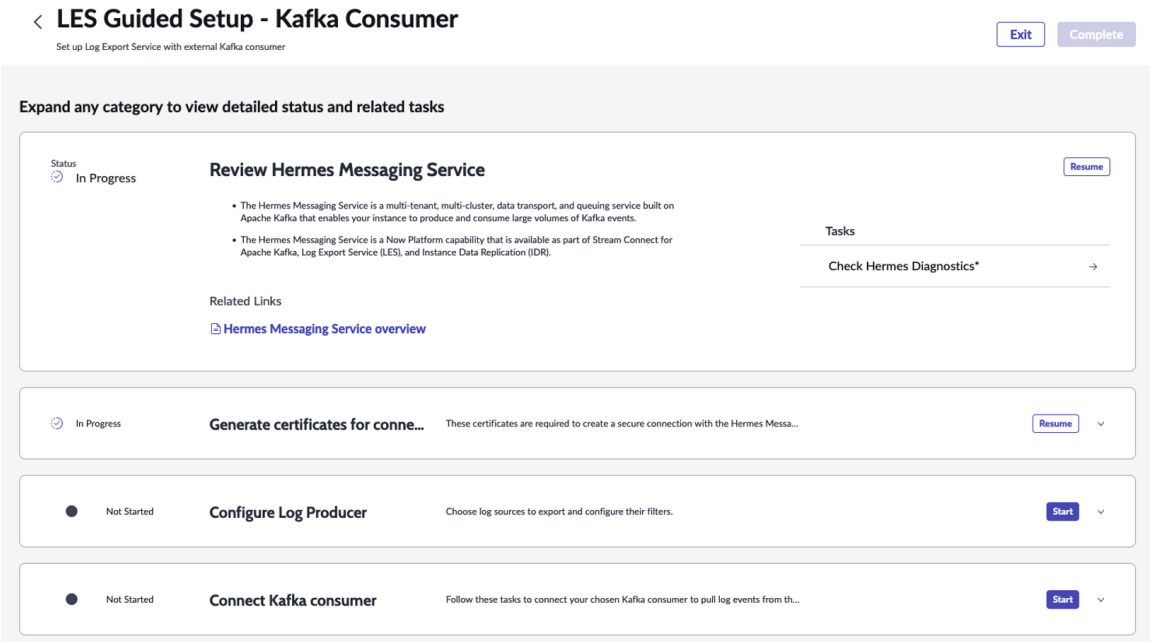


このページの 3 つのセットアップ (クイックスタート/最適なエクスペリエンス/カスタム) はすべて、同じタスクと機能を提供します。ServiceNow インスタンスと宛先ログアナリティクスツールの間の統合を各アドミニストレーターと調整する必要があります。ログアナリティクスアドミニストレーターは、ServiceNow インスタンスに安全に接続するようにツールを構成する必要があります。事前に Hermes メッセージングサービス への安全な接続の設定 ドキュメントをログアナリティ

クスアドミニストレーターと共有することをお勧めします。タスクを完了としてマークした場合でも、戻ってチェックを外して進行中の状態に戻すことができます。これを行うには、まずカテゴリの右上隅にある [編集] ボックスをクリックします。次に、チェックを外したいタスクの [編集] ボックスをクリックします。[完了としてマーク] ボックスはマークされなくなります。

ガイド付きセットアップのカテゴリページ

カテゴリページには、カテゴリと関連タスクの概要と説明が含まれています。下向きの矢印をクリックしてカテゴリに関する情報を表示するか、[Start (開始)] ボタンをクリックしてガイド付きセットアップ手順を開いて、構成を開始できます。



セットアップ手順に従って、各カテゴリのタスクを完了します。

Kafka コンシューマー向けのガイド付きセットアップ

Kafka コンシューマー向けの完全なガイド付きセットアップを行うには、次の手順を実行します。

始める前に

移動先 ログエクスポートサービス **(LES)** > **Kafka** コンシューマー > ガイド付きセットアップ. 構成するセットアップのタイプを選択し、[続行] を選択します。

> **注:** ログエクスポートサービスアプリケーションのインストール中、ServiceNow は、基盤となる Hermes メッセージングサービスインフラストラクチャをプロビジョニングします。このプロセスは、ログエクスポートサービスアプリケーションのインストールを要求してから最大で 2 時間程度かかることがあることに注意してください。

必要なロール：admin

手順

1. Hermes メッセージングサービス診断をレビューします。

   このステップで画面に表示される Hermes 診断ツールを使用して、Hermes メッセージングサービスが稼働していることを確認することをお勧めします。このページに「ページが見つかりません」というエラーが表示された場合、Hermes はインストールされていません。システムアドミンに連絡してください。

- セットアップ情報：次のブートストラップ情報は、Hermes メッセージングサービス への接続に使用されます。「プロデューサーブートストラップ」は Hermes にメッセージを送信するために使用される接続であり、「コンシューマーブートストラップ 1 および 2」は Hermes からメッセージを取得するために使用されます。
  - プロデューサーブートストラップ
  - コンシューマーブートストラップ 1
  - コンシューマーブートストラップ 2
- インスタンス PKI：インスタンス公開鍵インフラストラクチャ (PKI) コンポーネントにより、ServiceNow インスタンスは、X.509 信頼階層内で発行者として機能します。
- ブートストラップ接続:[テストを実行] を選択して、外部クライアントが定義されたインスタンスポート (プロデューサーとコンシューマー) に接続できることを確認します。
- インスタンス接続:[テストを実行] を選択して、インスタンスがメッセージを送受信できることを確認します。
- トピックを表示:リストされたトピックを選択して、最後の既知のメッセージのタイムスタンプを取得します。

> **注:** 今後、 Hermes 診断にアクセスして、潜在的な接続の問題をトラブルシューティングするには、ガイド付きセットアップのこのステップに戻るか、 すべて > *Hermes* メッセージングサービス > 診断.

2. Hermes メッセージングサービスへの安全な接続のための証明書を生成し、そこからログイベントをプルします。

外部システムに接続するときにこれらの証明書を使用します。

Hermes メッセージングサービスへの安全な接続をセットアップします。詳細については、「LES 用 Hermes Messaging Service への安全な接続を設定する」 を参照してください。これらの証明書は、Hermes からログをプルするクライアントでの認証と承認に必要となります。

> **注:** この手順には、admin ロールまたは Hermes_admin ロールが必要です。

3. ログプロデューサーの構成：エクスポートするログソースを選択し、フィルターを構成します。ログプロデューサーを構成するには、次のタスクを完了します。

- エクスポートするログソースの構成：エクスポートするログソースごとに 1 つのソースレコードを作成します。

  > **注:** この手順を完了するには、admin ロールまたは sn_logstoanalytics.admin ロールが必要です。

  a. 右上隅にある [新規] を選択します

  b. ソースタイプを選択します

  c. Select a Table

  d. ログ出力の制御に使用できるログレベルを選択します

  e. ログソースのエクスポート先のトピックを選択または作成します。新しいトピックを作成する場合は、次のフィールドに入力します。
  - 名前:作成しているトピックの名前
  - アプリケーション ID:sn_logstoanalyticsを入力します

- 名前空間:デフォルトの名前空間を入力します
- パーティション:Hermes のトピックのパーティションフィールドは、トピックのデータが分割されるパーティションを参照します。スケーラビリティと並列処理において重要な役割を果たします。

f. ログを選択的に転送するには、[フィルタータイプ] 条件を選択します。

> 注: フィルターは選択したソースの種類によって異なります

g. [Update (更新)] を選択します。
正常に作成されると、ログソースのエクスポート先になる Hermes トピックの名前が表示されます。トピック名を書き留めてください。後でログコンシューマーシステムを構成するときに必要になります。

[アクティブ] フィールドは、ログソースをエクスポートするかどうかを制御します。エラーが表示された場合は、「Hermes 診断の確認」タスクに戻り、Hermes ステータスを確認します。

- ログプロデューサーの検証:ログの生成元となるソースを作成したら、次を使用してトピック内のライブログレコードを表示できます。 **Hermes** メッセージングサービス > **Hermes** トピックインスペクター.

a. 外部トピックの選択

b. リストトピックを選択

c. 前のステップからトピックの行を選択 ([Sources (ソース)] にリストされているもの)

d. 必要に応じてメッセージの開始日を調整

e. [表示] を選択して、トピックにエクスポートされたログメッセージを表示します

4. Kafka コンシューマーの接続：以下のタスクに従って、選択した Kafka コンシューマーを接続して、Hermes からログイベントをプルします。

- Kafka コンシューマーの特定：ログアナリティクスアーキテクチャに基づいて 2 つのオプションがあります。
  - 独自の Kafka システムがあり、それをログアグリゲーション用に選択した場合は、ネイティブ Kafka プロトコルを介して Hermes Messaging Service に直接接続できます。
  - ログアナリティクスツールを Hermes Messaging Service に直接接続することを選択した場合は、ログアナリティクスシステムでサポートされている Kafka コネクタ (つまり、Splunk Connect for Kafka) を展開する必要があります。

  > 注: いずれの場合も、これらのシステムのアドミニストレーターと協力して、Hermes Messaging Service との接続を調整する必要があります。

- Kafka コンシューマーシステムへの Hermes 証明書のインポート：Kafka コンシューマーシステムにログインし、システムを構成して外部システムに接続するための適切なアドミンエンタイトルメントがあることを確認します。「Hermes Messaging Service への安全な接続のセットアップ」タスクで生成された証明書を Kafka コネクタまたは Kafka サーバーにインポートします。選択した Kafka コンシューマーのドキュメントの指示に従います。
- Kafka プロセスの構成：Hermes Messaging Service は、高い可用性を実現するように設計されています。Hermes からのメッセージを消費するには 2 つのプロセスが必要です。Hermes は、フェイルオーバー目的で、Kafka クラスターをペアで使用するため、プロセスが 2 つ必要になります。片方のクラスターがダウンすると、データはもう一方の Hermes Kafka クラスターに生成されます。

  Kafka コンシューマーシステムでは、両方の Hermes Kafka クラスターに接続するために 2 つの個別のコンシューマープロセスを作成する必要があります。両方のプロセスで同じ Hermes

Kafka トピックを指定しますが、次の 2 つの別々のブートストラップアドレスを構成する必要があります。

- <instance_name>.service-now.com:4100,<instance_name>.service-now.com:4101,<instance_name>.service-now.com:4102,<instance_name>.service-now.com:4103
- <instance_name>.service-now.com:4200,<instance_name>.service-now.com:4201,<instance_name>.service-now.com:4202,<instance_name>.service-now.com:4203

重要：

- 外部システムから Kafka トピックにアクセスする場合は、ログの転送先のトピックの先頭に「snc.<インスタンス名>」を付ける必要があります。
- 各コンシューマーを同じ Kafka コンシューマーグループ ID で構成します。
- キーストアファイルとトラストストアファイルを、コンシューマーがアクセスできる場所にインストールします。
- コンシューマーが必要とする場合は、次の Kafka JSON コンバーターのプロパティを指定してスキーマを無効にします：「key.converter.schemas.enable=false」、「value.converter.schemas.enable=false」。

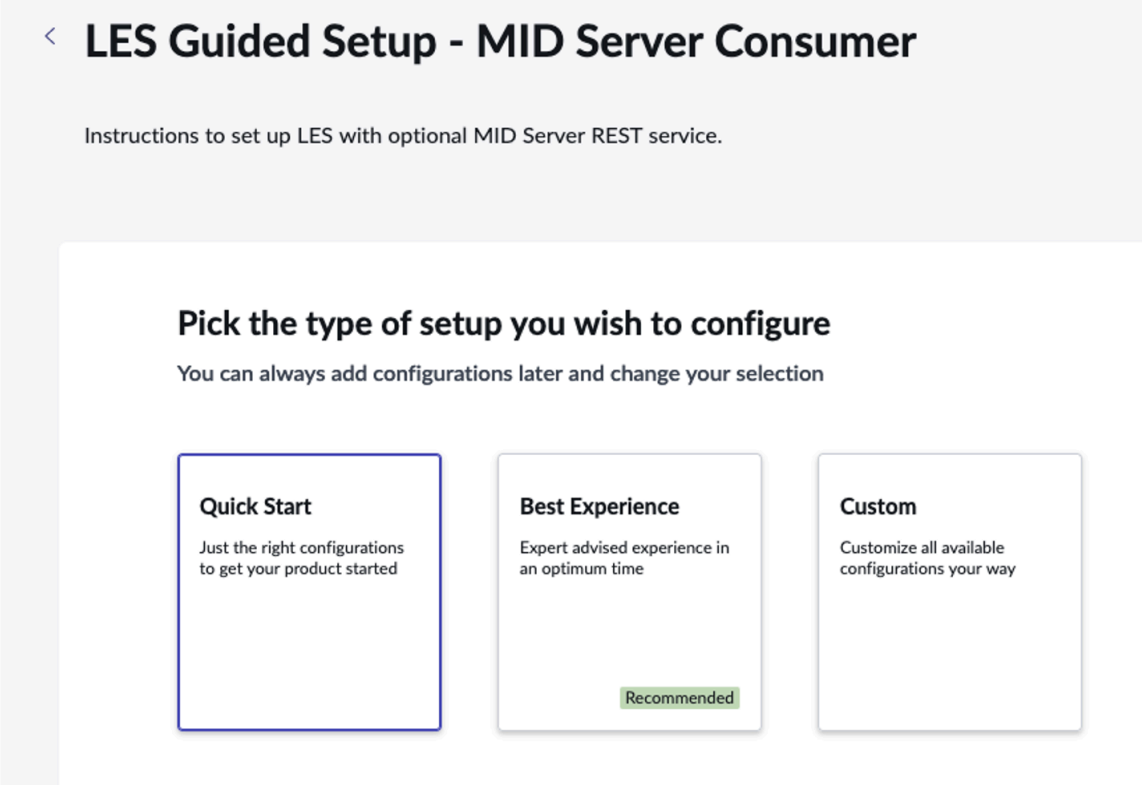
◦ Kafka コンシューマーが Herme からログをプルすることを確認する：選択した Kakfa コンシューマーで、 Hermes Messaging Service からログイベントをプルできることを確認します。

MID サーバーコンシューマー

ガイド付きセットアップを使用して、MID サーバーコンシューマー向けの LES の初期構成を段階的に実行します。

ガイド付きセットアップのホームページ

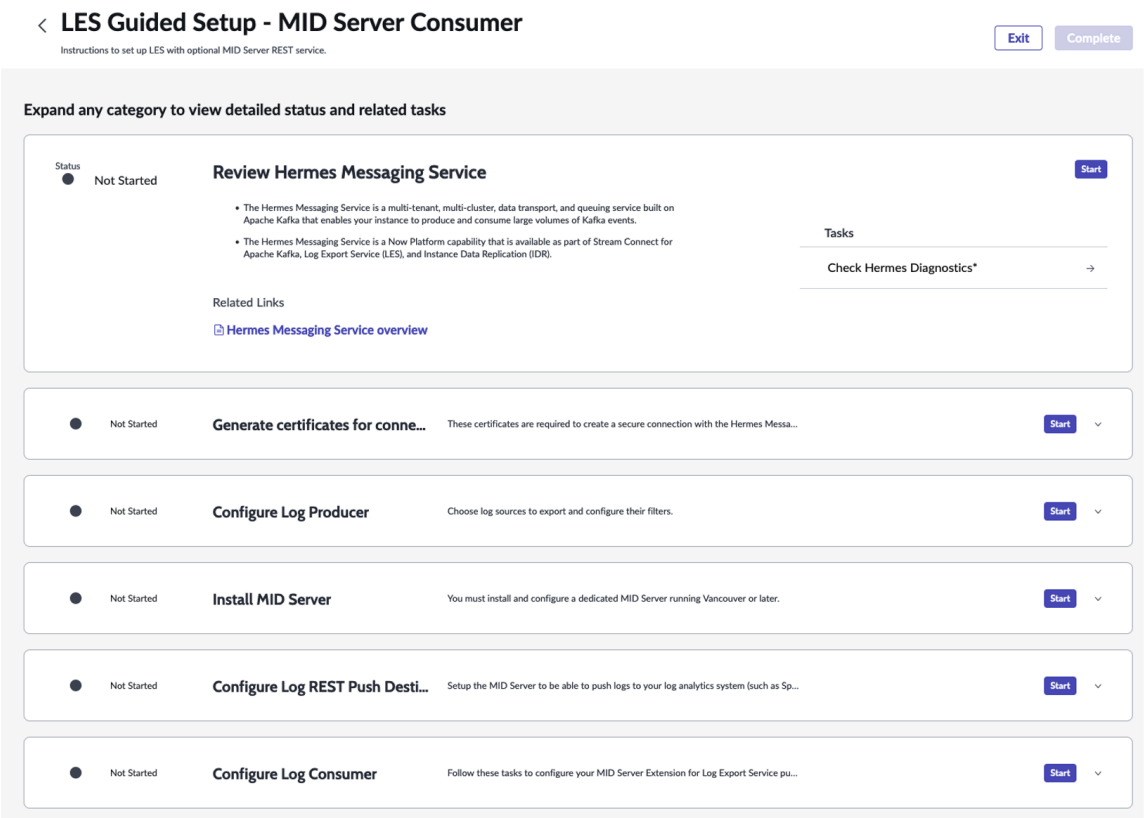
ガイド付きセットアップのホームページには、ガイド付きセットアップの構成タイプの概要が含まれています。ガイド付きセットアップのタイプを選択し、[続行] を選択すると、ガイド付きセットアップ手順が開き、構成を開始できます。



このページの 3 つのセットアップ (クイックスタート/最適なエクスペリエンス/カスタム) はすべて、同じタスクと機能を提供します。インスタンスからログアナリティクスシステムにログを継続的にストリーミングするには、専用の MID サーバーが必要です。MID サーバーは、Hermes Messaging Service への安全な接続を確立するための 1 回限りのセットアップが必要です。タスクを完了としてマークした場合でも、戻ってチェックを外して進行中の状態に戻すことができます。これを行うには、まずカテゴリの右上隅にある [編集] ボックスをクリックします。次に、チェックを外したいタスクの [編集] ボックスをクリックします。[完了としてマーク] ボックスはマークされなくなります。

ガイド付きセットアップのカテゴリページ

カテゴリページには、カテゴリと関連タスクの概要と説明が含まれています。ドロップダウン矢印をクリックしてカテゴリに関する情報を表示することも、[開始] をクリックしてガイド付きセットアップ手順を開いて構成を開始することもできます。



セットアップ手順に従って、各カテゴリのタスクを完了します。

MID サーバーコンシューマー向けのガイド付きセットアップ

MID サーバーコンシューマー向けの完全なガイド付きセットアップを行うには、次の手順を実行します。

始める前に

移動先 ログエクスポートサービス **(LES)** > **MID** サーバーコンシューマー > ガイド付きセットアップ. 構成するセットアップのタイプを選択し、[続行] を選択します。

> **注:** ログエクスポートサービスアプリケーションのインストール中、ServiceNow は、基盤となる Hermes メッセージングサービスインフラストラクチャをプロビジョニングします。このプロセスは、ログエクスポートサービスアプリケーションのインストールを要求してから最大で 2 時間程度かかることがあることに注意してください。

必要なロール：admin

手順

1. Hermes メッセージングサービス診断をレビューします。

   このステップで画面に表示される Hermes 診断ツールを使用して、Hermes メッセージングサービスが稼働していることを確認することをお勧めします。このページに「ページが見つかりません」というエラーが表示された場合、Hermes はインストールされていません。システムアドミンに連絡してください。

   - セットアップ情報：次のブートストラップ情報は、Hermes Messaging Service への接続に使用されます。「プロデューサーブートストラップ」は、Hermes にメッセージを送信するために使用される接続であり、「コンシューマーブートストラップ 1 および 2」は、Hermes からメッセージを取得するために使用されます。

- プロデューサーブートストラップ
- コンシューマーブートストラップ 1
- コンシューマーブートストラップ 2

◦ インスタンス PKI：インスタンス公開鍵インフラストラクチャ (PKI) コンポーネントにより、ServiceNow インスタンスは、X.509 信頼階層内で発行者として機能します。

◦ ブートストラップ接続:[テストを実行] を選択して、外部クライアントが定義されたインスタンスポート (プロデューサーとコンシューマー) に接続できることを確認します。

◦ インスタンス接続:[テストを実行] を選択して、インスタンスがメッセージを送受信できることを確認します。

◦ トピックを表示:リストされたトピックを選択して、最後の既知のメッセージのタイムスタンプを取得します。

注: 今後、 Hermes 診断にアクセスして、潜在的な接続の問題をトラブルシューティングするには、ガイド付きセットアップのこのステップに戻るか、 すべて > *Hermes* メッセージングサービス > 診断.

2. Hermes Messaging Service への安全な接続のための証明書を生成し、そこからログイベントをプルします。

Hermes Messaging Service (メッセージングサービス) への安全な接続をセットアップします。詳細については、 「LES 用 Hermes Messaging Service への安全な接続を設定する」 を参照してください。これらの証明書は、Hermes からログをプルするクライアントでの認証と承認に必要となります。

注: この手順には、admin ロールまたは Hermes_admin ロールが必要です。

3. ログプロデューサーの構成：エクスポートするログソースを選択し、フィルターを構成します。ログプロデューサーを構成するには、次のタスクを完了します。

◦ エクスポートするログソースの構成：エクスポートするログソースごとに 1 つのソースレコードを作成します。

注: この手順を完了するには、admin ロールまたは sn_logstoanalytics.admin ロールが必要です。

新しいソースを作成するには、次に移動します: ログエクスポートサービス > ソース

a. 右上隅にある [新規] を選択します

b. ソースタイプを選択します

c. Select a Table

d. ログ出力の制御に使用できるログレベルを選択します

e. ログソースのエクスポート先のトピックを選択または作成します。新しいトピックを作成する場合は、次のフィールドに入力します。

- 名前:作成しているトピックの名前
- アプリケーション ID:sn_logstoanalyticsを入力します
- 名前空間:デフォルトの名前空間を入力します
- パーティション:Hermes のトピックのパーティションフィールドは、トピックのデータが分割されるパーティションを参照します。スケーラビリティと並列処理において重要な役割を果たします。

f. ログを選択的に転送するには、[フィルタータイプ] 条件を選択します。

注: [フィルタータイプ] オプションは、選択したソースタイプによって異なります。

g. [Update (更新)] を選択します
正常に作成されると、ログソースのエクスポート先になる Hermes トピックの名前が表示されます。トピック名を書き留めてください。後でログコンシューマーシステムを構成するときに必要になります。

[アクティブ] フィールドは、ログソースをエクスポートするかどうかを制御します。エラーが表示された場合は、「Hermes 診断の確認」タスクに戻り、Hermes ステータスを確認します。

- ログプロデューサーの検証:ログの生成元となるソースを作成したら、次を使用してトピック内のライブログレコードを表示できます。 **Hermes** メッセージングサービス > **Hermes** トピックインスペクター.

  a. 外部トピックの選択

  b. リストトピックを選択

  c. 前のステップからトピックの行を選択 ([Sources (ソース)] にリストされているもの)

  d. 必要に応じてメッセージの開始日を調整

  e. [表示] を選択して、トピックにエクスポートされたログメッセージを表示します

4. MID サーバーのインストール：Vancouver 以降を実行する専用の MID サーバーをインストールして構成する必要があります。
MID サーバーをインストールするには、次のタスクを完了します。

- 専用の MID サーバーのインストール：ログエクスポートサービスが使用する MID サーバーは、この目的専用で、他のプロセスの実行を期待されないようにする必要があります。これは、エクスポートされたログメッセージを REST エンドポイントにタイムリーに配信するうえで重要です。新しい MID サーバーをインストールするには、MID サーバーガイド付きセットアップを使用する を使用するか、手動でインストールします。手動インストールの場合は、最初に MID サーバーのネットワーク接続の設定 ドキュメントに従い、その後で MID サーバーのインストール ドキュメントに従います。
- MID サーバーの検証：MID サーバーがインストールされた後は、手動で MID サーバーを検証して、自動化タスクを実行できるようにする必要があります。LES 専用の MID サーバーを検証するには、「MID サーバーの検証 」を参照してください。

5. ログ REST プッシュ先の構成：ログアナリティクスシステム (Splunk など) にログをプッシュできるように MID サーバーをセットアップします。
ログ REST プッシュ先を構成するには、次のタスクを完了します。

- MID プロパティの追加：Hermes に接続できるように、MID サーバープロパティを追加する必要があります。 **[MID** サーバー] > [プロパティ] に移動し、以下にリストされている各プロパティの適切な値を設定します。

名前	値
mid.les.kafka.ssl.truststore.password	<パスワード>
mid.les.kafka.ssl.keystore.password	<パスワード>
mid.les.kafka.ssl.key.password	<パスワード>
mid.les.kafka.ssl.truststore.location	<パス>/<トラストストア>.p12
mid.les.kafka.ssl.keystore.location	<パス>/<キーストア>.p12

名前	値
mid.les.kafka.ssl.truststore.type	PKCS12
mid.les.kafka.ssl.keystore.type	PKCS12
mid.les.kafka.client.id	<インスタンス名>
mid.les.kafka.group.id	snc.<インスタンス名>.group1
mid.les.kafka.bootstrap.servers	<instance_name>.servicenow.com:4100,<instance_name>.s
mid.les.kafka.set2.bootstrap.servers	<instance_name>.servicenow.com:4200,<instance_name>.s

上記の値の一部を取得する方法については、次の注意事項に従ってください。

- <パスワード> はキーストアとトラストストアに設定したパスワードです。
- <パス> は、ダウンロードしたキーストアファイルとトラストストアファイルを保持するディレクトリへのファイルパスです。証明書は MID を実行しているサーバー上にある必要があります。
- <インスタンス名> は ServiceNow インスタンスの名前です不明な場合は、sys_properties テーブルで見つけることができます。
- Hermes の診断ページから mid.les.kafka.bootstrap.servers と mid.les.kafka.set2.bootstrap.servers の両方の値を取得できます。移動先 **Hermes** メッセージングサービス > 診断 をクリックし、[コンシューマーブートストラップ 1] と [コンシューマーブートストラップ 2] の下の文字列をそれぞれコピーします。

◦ 宛先の構成：新しい宛先設定レコードを作成します。これは、この拡張機能がログを転送する REST エンドポイントを定義します。

> **注:** この手順を完了するには、admin ロールまたは sn_logstoanalytics.admin ロールが必要です。

a. 移動先 ログエクスポートサービス **(LES)** > 宛先の構成

b. 新しい設定レコードを作成します。

c. エクスポートされたログソースの目的のエンドポイントの URL を指定します。

d. エンドポイントに接続するための新しい認証情報を検索または作成します。エンドポイントの認証情報を作成するときは、LES では、Basic Auth、OAuth、API キーの認証情報タイプのみが有効であることに注意してください。

e. 変換スクリプトを検索するか新規に作成します。事前に作成されたスクリプト、Splunk 用の **SplunkTransform** が付属しています。

6. ログコンシューマーの構成：次のタスクに従って、ログエクスポートサービスを目的とした MID サーバー拡張機能を構成します。

◦ LES コンシューマーコンテキストの構成：この手順では、ログエクスポートサービス用にインストールした専用の MID サーバー上で実行されるように LES コンシューマーレコードを更新します。移動先 **MID** サーバー > 拡張 > **LES** コンシューマーコンテキスト をクリックし、次のフィールドを設定して LES コンシューマーレコードを更新します。

- [LES コンシューマー] を選択して、MID サーバーコンテキストレコードを開きます
- [Execute on (実行)] フィールドで特定の MID Server (MID サーバー) を選択します
- [MID Server (MID サーバー)] フィールドに、前の手順で検証した MID の名前を入力します

[更新] を選択して保存します。

> **注:** 事前にビルドされたコンテキストが付属しています。2 つ目のコンテキストは作成しないでください。これにより、意図しない結果が生じる可能性があります。

- コンシューマーの構成：ログエクスポートサービス MID サーバー拡張機能の一部であるプロセスを表す新しいコンシューマーレコードを作成します。移動先 ログエクスポートサービス **(LES)** > コンシューマー をクリックし、ログメッセージを取得する Hermes トピックとログメッセージのリレー先を指定する新しい設定レコードを作成します。[コンシューマー] モジュールを選択すると、コンシューマー名と宛先構成に関する情報が表示されます。

  a. 新しいコンシューマーレコードを作成します

  b. ドロップダウンからソーストピックを選択します

  c. 宛先構成を選択します

  d. コンシューマーを起動します

- MID サーバーの統合の確認:移動先 ログエクスポートサービス **(LES)** > コンシューマーステータス をクリックし、定義されたレコードの [ステータス] フィールドと [ステータスの詳細] フィールドを表示します。これらのフィールドの情報は、REST エンドポイントへのメッセージの中継中に発生した可能性のあるエラーを含め、MID サーバーで実行されているプロセスの現在の状況を報告します。これはステータスビューページのみです。移動先 ログエクスポートサービス **(LES)** > コンシューマー 新しいコンシューマーレコードを作成する場合。#コンシューマーステータスがプロセスが開始されたことを示している場合は、エンドポイントを調べて、エンドポイントに中継されたログを表示できるはずです。さらに、MID サーバーのログを表示して、発生した可能性のあるエラーに関する追加の詳細があるかどうかを確認できます。必要に応じて、MID サーバーのデバッグログを有効にして追加情報を取得することもできます。

  > **注:** コンシューマーレコードの 1 つに変更を加えると、その変更は [コンシューマーステータス] ビューページに表示されます。[コンシューマーステータス] リストでコンシューマーレコード名を選択すると、選択したレコードのコンシューマーフォームが開きます。その後、選択したレコードの [名前] と [宛先構成 (Destination Configuration)] を更新できます。

一意の **MID** サーバーを使用したマルチコンシューマーサポート

新しいマルチコンシューマーシステムでログ消費を正確に管理できるようになり、特定のログストリームごとに専用のコンシューマーと MID サーバーを使用できるようになりました。

システムでマルチコンシューマーログの消費がサポートされるようになりました。つまり、各ログソースを独自の専用トピックで個別に消費できます。

以前は、すべてのログが同じトピックから消費されていました。ただし、新しいマルチトピックの概念では、さまざまなトピックに対して複数のコンシューマーを作成できます。コンシューマーを作成するには、以下を選択します。

- ドロップダウンメニューからの特定のトピック
- それぞれの宛先の構成
- それぞれのコンシューマーコンテキスト

> **注:** チューリッヒリリース以降、 コンシューマーコンテキスト は [コンシューマー] フォームに追加された新しいフィールドです。各コンシューマーコンテキストは、1 つの一意の MID サーバーに関連付けられています。

複数のコンシューマーを作成するには、コンシューマーごとに個別のトピックを選択する必要があります。この場合、個別のコンシューマーコンテキストレコードとそれに対応する一意の MID サーバーを作成する必要があります。

複数の MID サーバーによる並列処理

このマルチコンシューマーアーキテクチャの主な利点は、複数のコンシューマーを並行して実行できることです。この並列処理により、ログ消費の全体的なスループットと効率が大幅に向上し、システムは以前よりも大量の多様なログデータをより効果的に処理できるようになります。

LES の Hermes Messaging Service への安全な接続を設定する

ServiceNow® インスタンス署名証明書を生成して Kafka トピックを保護します。

始める前に
Hermes メッセージングサービス を設定するには、ネットワーク管理者および Kafka 管理者との調整が必要です。ネットワークアドミニストレーターと協力して必要なセキュリティ証明書を取得し、必要なポートを開きます。Kafka アドミニストレーターと協力して、Kafka 環境が正しく構成され、アプリケーションが標準の Kafka プロトコルを使用して Hermes メッセージングサービス に接続できることを確認します。

次のセットアップが行われていることを確認します。

- Hermes メッセージングサービス がアクティブ化されている。「Hermes メッセージングサービス のアクティブ化」を参照してください。
- キー管理フレームワークプラグイン (com.glide.kmf.global) がアクティブ化されている。
- 証明書 [sys_kmf_certificate] テーブルに、ServiceNow インスタンスのルート CA 証明書が含まれている。
- インスタンスがカスタム URL で構成されていない。カスタム URL は、インスタンス PKI 証明書ジェネレーターではサポートされていません。

必要なロール：hermes_admin、sn_kmf.cryptographic_manager または admin

KMF ロールのアサインの詳細については、「キー管理フレームワークとともにインストールされるロール」を参照してください。

手順

1. 移動先 すべて > 証明書ジェネレーター > インスタンス **PKI** 証明書ジェネレーター.
2. オプション: 名前空間またはトピックレベルでアクセス制御リスト (ACL) を設定して、トピックへのアクセスを制御します。

オプション	説明
名前空間に ACL を適用する	**a. [ACL** を構成 **(Configure ACLs)]** を選択します。 **b.** [トピック ACL] ダイアログボックスで、[名前空間] を選択します。 **c.** 構成する名前空間を入力します。 **d.** [読み取り専用] または [読み取り/書き込み] を選択して、アクセス許可レベルを設定します。 **e.** [追加] を選択します。

自動翻訳

オプション	説明
定義されたトピックに ACL を適用する	**a.** **[ACL を構成 (Configure ACLs)]** を選択します。 **b.** [トピック ACL] ダイアログボックスで、[定義されたトピック] を選択します。 **c.** 設定する既存のトピックを入力します。 **d.** [読み取り専用] または [読み取り/書き込み] を選択して、アクセス許可レベルを設定します。 **e.** [追加] を選択します。

証明書の対象者には、名前空間内のトピックまたは選択した既存のトピックへの読み取りアクセス権または読み取り/書き込みアクセス権が付与されます。

3. Hermes メッセージングサービス のセキュリティを設定します。
   a. [インスタンス PKI 証明書ジェネレーター] ページに戻ります。
   b. [証明書パスワード] フィールドにキーストアパスワードを入力します。
   c. [生成] を選択します。
   システムは、証明書 [sys_kmf_certificate] テーブルにインスタンス署名付き証明書を生成し、キーストアを作成して、トラストストアを作成します。

   IPKI 証明書ジェネレーターで制限付きの申請者アクセスが許可されていない場合は、クロススコープアクセスエラーが表示されます。制限付き発信者アクセスの許可については、 カスタマーサービス & サポート にお問い合わせください。この問題を解決するには、 カスタマーサービス & サポート 制限付き発信者アクセス特権 [sys_restricted_caller_access] テーブルで source_scope=76f9d51369115083f4ea77aab1677cc0 を参照できます。
4. [キーストアをダウンロード] を選択して、キーストアのコピーを保存します。
5. [トラストストアをダウンロード] を選択して、トラストストアのコピーを保存します。
6. キーストアファイルとトラストストアファイルを、Hermes メッセージングサービス に接続する各プロデューサーとコンシューマークライアントにコピーします。

結果
これで、Hermes メッセージングサービス への安全な接続を作成できます。

注: Hermes に接続するには、インスタンス PKI 証明書ジェネレーターを使用して生成したキーストアを使用する必要があります。ServiceNow のドキュメントに従って作成されていないカスタム生成のキーストアは、サポートされていません。

次のタスク
。

ログエクスポートサービス (LES) の使用

LES を使用して、ログレポートダッシュボードを確認します。

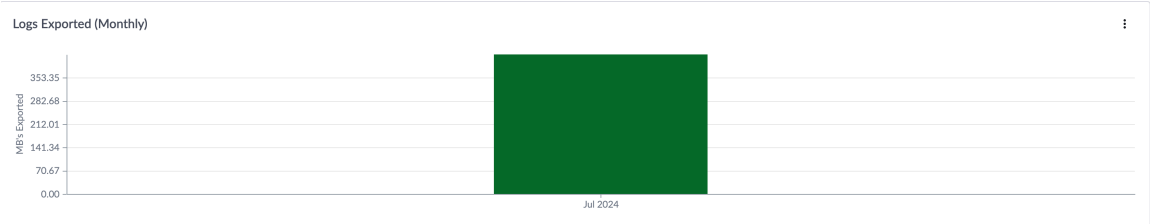
ログレポートのレビュー

ログレポートダッシュボードをレビューして、各データログソースのサイズを分析します。

始める前に
必要なロール：admin または sn_logstoanalytics.admin

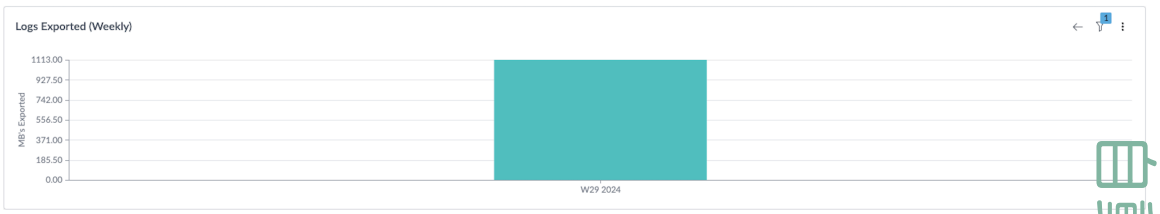
手順

1. 移動先 すべて > ログエクスポートサービス > レポート.
2. 次のウィジェットでログデータを確認します。
   - エクスポートされたログ (月次) (Logs Exported (Monthly)) : 初期ビューには、1 か月あたりのエクスポートされたメガバイト数が表示されます。グラフをドリルダウンして、各週または日次のデータを表示できます。過去 395 日分のデータを表示できます。



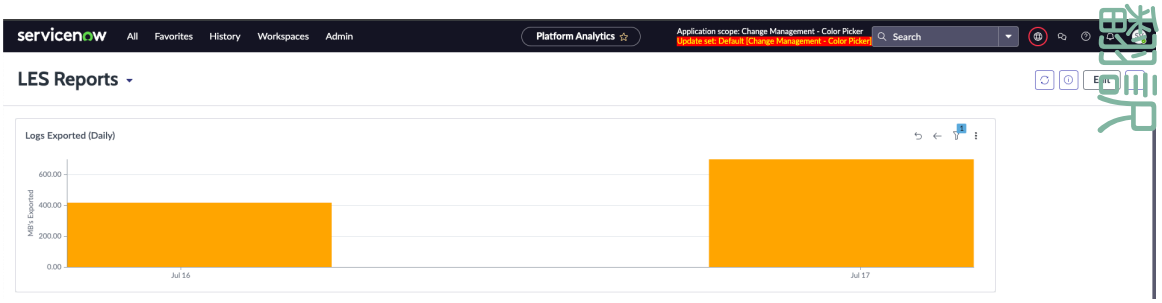
注: ウィジェットの右上隅にある矢印をクリックして、データをドリルダウンします。親データに戻ることもできます。

週



次：

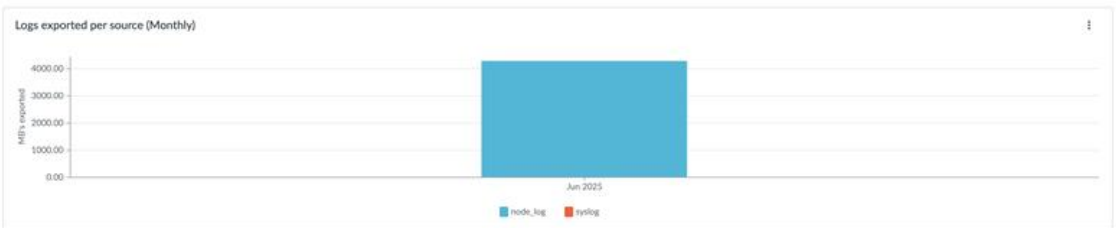
日



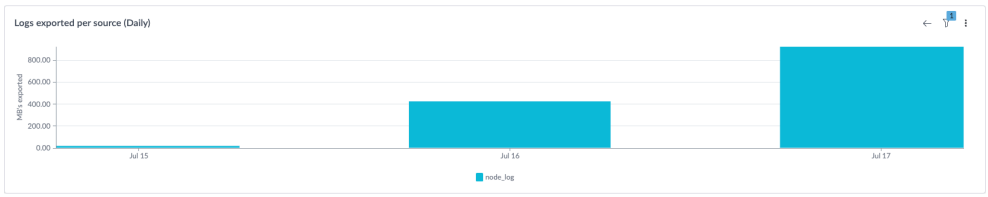
次：

このレポートのデータは、4 時間前に取得されたコレクションを反映しています。

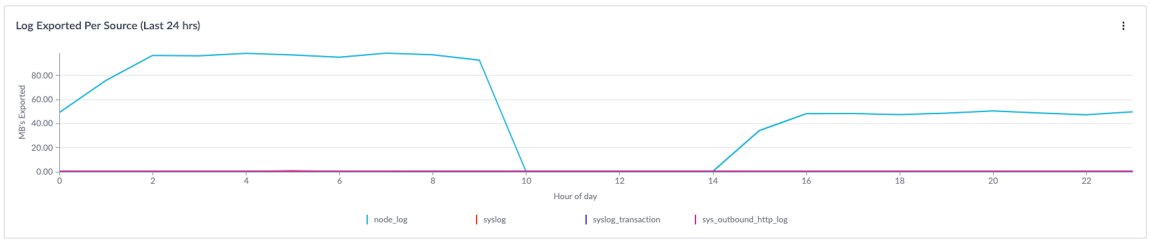
   - ソース別のエクスポートされたログ (月次) (Logs exported per source (Monthly)):このウィジェットには、月ごとにエクスポートされたログの種類が表示されます。ドリルダウンして、週次および日次のログデータを表示することもできます。過去 395 日分のデータを表示できます。



注: データを表示するソースを選択または選択解除できます。ログを表示できるソースは、node_log、syslog、syslog_transaction、および sys_outbound_http_log です。

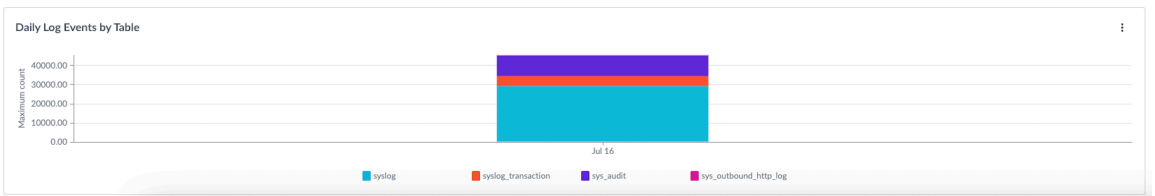


- ソース別のエクスポートされたログ (過去 24 時間) (Log Exported Per Source (Last 24 hrs)) : 過去 24 時間のデータが表示されます。グラフは 24 時間スケールで左から右に遡り、最も古いデータ (24 時間前) がレポートの一番右に、最新のデータが一番左に配置されます。データを表示するソースを選択または選択解除できます。ログを表示できるソースは、node_log、syslog、syslog_transaction、および sys_outbound_http_log です。



注: このデータを確認する時刻に応じて、ウィジェットは動的に更新され、正確に過去 24 時間が反映されます。

- テーブル別の日次ログイベント (Daily Log Events by Table) : 各ログテーブルによって生成されたログイベントの日次データが表示されます。 データを表示するテーブルを選択または選択解除できます。



注: ログイベントを表示できるテーブルは、syslog、syslog_transaction、sys_audit、および sys_outbound_http_log です。

ログエクスポートサービス (LES) の参照

参照セクションで LES に関するその他の情報をすべて見つけてください。

ログエクスポートサービス ロール

ログエクスポートサービス はこれらのロールに併せてインストールされます。

アプリケーションアドミニストレーター [sn_logstoanalytics.admin]

ユーザーごとのサブスクリプション管理の詳細については、「サブスクリプション管理 でのユーザーごとのサブスクリプションの管理」を参照し、アカウント担当者にお問い合わせください。

このロールは LES アプリケーションとともにインストールされ、アドミン以外のロールがアプリケーションを使用できるようにします。

ロールを含む

ロール内に含まれるロールのリスト。

なし。

グループ

このロールがデフォルトでアサインされているグループのリスト。

なし。

昇格

ロールが昇格されたロールかどうか。昇格されたロールはユーザーまたはグループにアサインされず、昇格で使用する必要があります。詳細については、「特権ロールへの昇格」を参照してください。

なし。

特別な考慮事項

なし。

システムアドミニストレーター **[admin]**

ユーザーごとのサブスクリプション管理の詳細については、「サブスクリプション管理 でのユーザーごとのサブスクリプションの管理」を参照し、アカウント担当者にお問い合わせください。

LES ストアアプリケーションのセットアップには、admin ロールが必要です。

ロールを含む

ロール内に含まれるロールのリスト。

- sn_templated_snip.template_snippet_admin
- sn_employee.admin
- taxonomy_admin
- sn_ace.ace_user
- sn_hr_sp.esc_admin

グループ

このロールがデフォルトでアサインされているグループのリスト。

なし。

昇格

ロールが昇格されたロールかどうか。昇格されたロールはユーザーまたはグループにアサインされず、昇格で使用する必要があります。詳細については、「特権ロールへの昇格」を参照してください。

いいえ。

特別な考慮事項

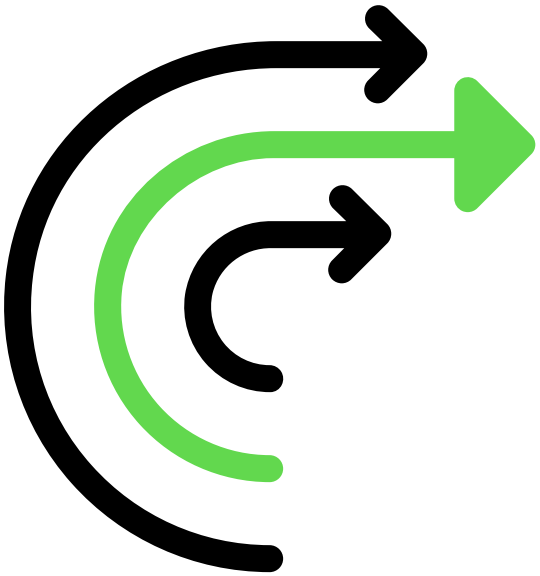
なし。

ログ

ログモジュールは、インスタンス内で発生するトランザクションやイベントのトラブルシューティングとデバッグに使用できるさまざまなログを提供します。

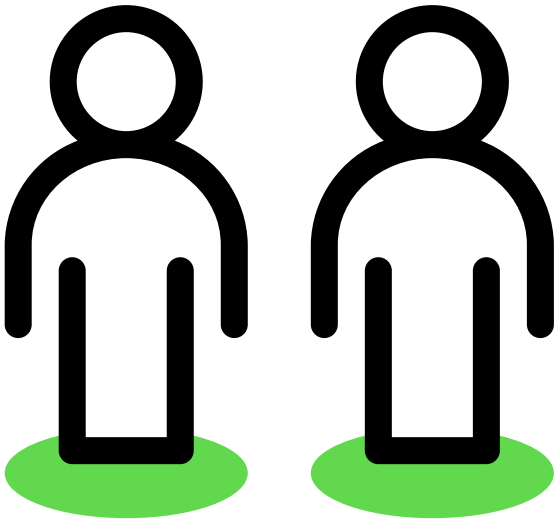
開始

システムログ



システムログモジュールは、インスタンス内で発生するトランザクションやイベントのトラブルシューティングとデバッグに使用できるさまざまなログを提供します。

ログ記録、監査、およびエラー



ログ記録と監査戦略を適用して、不審なアクティビティを適切なタイミングで特定して対処できるようにします。

システムログ

システムログモジュールは、インスタンス内で発生するトランザクションやイベントのトラブルシューティングとデバッグに使用できるさまざまなログを提供します。

システムログモジュールから次のログにアクセスします。

ログ	説明
トランザクション	インスタンスのすべてのアプリケーションアクティビティ。
メールとプッシュ	システム内のすべてのインスタンスから送信されたすべてのメール通知とプッシュメッセージ。
イベントログ	システム内で発生するすべてのシステムイベント。
インポート	プラットフォーム内のデータインポートアクティビティ。
テーブル変更	システム内のすべてのテーブルに加えられた変更。

ログ	説明
送信 Web サービスのログ記録	REST 要求や SOAP 要求などのすべての送信 Web サービスの要求。
署名画像	人事署名パッドの電子署名。
System	インスタンスプロセス、レコード、およびサーバーマシンのメモリ使用率などの重要でないイベントの警告とエラー。

ログファイルブラウザーを使用して、ログを検索してダウンロードします。ログ履歴でアーカイブされたログを検索することもできます。

その他のログ

インスタンスは、システムログモジュール内のログに加えて他のログを提供します。たとえば、システム診断モジュール はアップグレード履歴と遅いクエリログを提供します。これらを使用すると、クエリがプラットフォームのパフォーマンスにどのように影響しているかを把握できます。顧客アップデートテーブル は、システムで行われたすべての変更を記録します。

システムログ

インスタンスプロセス、レコード、およびサーバーマシンのメモリ使用率などの重要でないイベントの警告とエラーを表示します。

次の情報がシステムログで追跡されます。

- ワークフロー
- 設定
- チャットセッション
- システム内の各ページの各ビューのトランザクション (ネットワーク、サーバー、ブラウザのロード時間など)
- 受信メールと送信メール
- システムでトリガーされたイベント
- インポートと統合
- システムの警告、エラー、およびスクリプトログ
- プラグインのアクティベーション、更新セット、またはシステムのアップグレードに関するアップグレード情報

ログエントリは当日にのみ表示されます。他のログファイルを表示するには、ログファイルブラウザを使用します。このログには、すべてのオカレンスに関して次の情報が表示されます。

システムログ

フィールド	説明
作成日時	インスタンスを実行しているマシンのロケールのログ記録アクティビティの日時。
レベル	メッセージのタイプレベルは、デバッグ、エラー、警告、および情報です。

システムログ (続く)

フィールド	説明
	• 警告は、処理されて復旧されたエラーです。 • エラーは修正する必要があります。
メッセージ	オカレンスの性質に関するシステム生成メッセージ。
ソース	オカレンスの影響を受けるプロセスまたはエリアの名前。たとえば、オカレンスのソースはメールまたはメモリです。
ソースパッケージ	オカレンスに関連付けられているアプリケーションパッケージの名前。名前をクリックすると、このパッケージのストアアプリケーション [sys_store_app] レコードが表示されます。

ワークフローログ

- 次を含む、実行された各アクティビティ
  - 開始日時
  - 終了日時
  - ステータス (完了、キャンセル、タイムアウト、エラーなど)
  - 結果
  - 障害の説明 (エラーが発生した場合)
- 次を含む移行履歴：
  - 移行時間
  - アクティビティの移行元
  - アクティビティの移行先
  - トリガーされた移行
- ワークフローに追加されたログステートメントを含むワークフローログ

構成情報

- 実行したアクション (挿入、更新、削除など)
- 変更のカテゴリ
- 変更とともに記録されたコメント
- 変更の名前
- 変更の XML の差異
- 変更が関連付けられている更新セット
- 変更の日時
- 変更を加えたユーザー
- 変更が行われたテーブル

- 変更されるオブジェクトの名前
- 変更されるオブジェクトのタイプ
- フォームまたはリストに加えられた変更の表示

トランザクションログ

トランザクションログには、インスタンスのすべてのブラウザアクティビティが記録されます。システム障害のデバッグを支援するために、トランザクションログをアプリケーションスコープでフィルタリングし、表示されるトランザクションを特定のスコープで発生したトランザクションのみに制限することができます。

に移動し、トランザクションログにアクセスします すべて > システムログ > トランザクション. トランザクション ログは、すべてのアクティビティについて次の情報を提供します。

フィールド	説明
作成日時	インスタンスを実行しているマシンのロケールのアプリケーションアクションの日時。
タイプ	記録されたトランザクションのタイプ。
作成者	このアクティビティを作成したユーザー。
作成元アプリケーション	トランザクションが開始されたアプリケーションスコープ。トランザクションがグローバルスコープで開始された場合、[グローバル] が表示されます。
応答時間	アプリケーション要求の往復応答時間 (ミリ秒)。
ネットワーク時間	アプリケーション要求が行われた後のネットワーク応答の待ち時間 (ミリ秒)。
出力長	インスタンスからアプリケーションに送信される出力文字列のサイズ (バイト)。
SQL カウント	このアクティビティに対して実行された SQL サーバーのコマンド数。
ビジネスルール数	このアクティビティに対して実行されたビジネスルール数。
ビジネスルール時間	このアクティビティのビジネスルールの実行経過時間。
URL	クライアントアプリケーションによって接続されたアプリケーションまたはモジュール。
Sys ID	要求を行うクライアントインスタンスのシステム生成識別子。この ID は、複数のインスタンス (ノード) がデータベースと通信するクラスター環境で使用されます。
IP アドレス	要求を行うクライアントの IP アドレス。
GZip 完了	圧縮された Web ページがアプリケーションによって要求されたかどうかを示します。
プロトコル	このインスタンスのアプリケーションで使用される HTTP プロトコル。

クライアントトランザクションタイミング

クライアントトランザクションタイミングプラグインは、クライアントとサーバー間で行われるトランザクションの期間に関する追加情報を提供することで、システムログを拡張します。

時間がどこで消費されているか、トランザクション中にどのように時間が費やされたかを表示することで、ソースに至るまでパフォーマンスの問題を追跡できます。

このプラグインでは、応答時間インジケーター 応答時間インジケーター を有効にする必要があり、次のブラウザーから情報を収集します。

- Firefox
- Internet Explorer
- Chrome

クライアントトランザクション情報

このプラグインをインストールすると、クライアントトランザクションモジュールがシステムログアプリケーションに追加されます。これは、最終日内にログに記録されたクライアントとサーバーとの間でのすべてのトランザクションのリストを提供します。

クライアントトランザクション情報

フィールド	説明
作成日時	トランザクションが記録された時刻
応答時間	サーバーがトランザクションの実行に費やしたミリ秒数
ビジネスルール時間	トランザクションによってトリガーされたビジネスルールによって費やされたミリ秒数
SQL 時間	SQL データベースが費やしたミリ秒数
クライアント応答時間	(Load_completion_time) - (start_time)。サーバー時間を含みます。
クライアントネットワーク時間	クライアントがネットワークの接続に費やしたミリ秒数
ブラウザー時間	ブラウザーがトランザクション中に費やしたミリ秒数
クライアントスクリプト時間	クライアントスクリプトの実行に費やされたミリ秒数
UI ポリシー時間	UI ポリシーの実行に費やされたミリ秒数
タイプ	トランザクションタイプ： • フォーム • リスト • その他
テーブル	表示されるテーブルの名前。たとえば、incident、change_request など。
表示	このフォーム/リストの表示

クライアント詳細情報

すべてのフォームのレンダリング (リストのレンダリングではない) について、クライアントタイミングのより詳細なブレークダウンも追跡されます。詳細を見るには、特定のクライアントトランザクションレコードをドリルダウンし、画面の下部にある関連リストを確認します。

クライアント詳細情報

フィールド	説明
順番	この操作が行われたロード時の順番
タイプ	操作のタイプ
名前	この特定の操作の説明的な名前
期間	この操作が完了するまでにかかったミリ秒数

プッシュログ

プッシュログを参照して、システムから送信するためにキューに入れられているプッシュ通知のステータスを追跡します。

プッシュログを表示するには、次に移動します: システムログ > プッシュ通知. プッシュログを表示するには、ユーザーは push_admin または admin ロールを持っている必要があります。

プッシュログフィールド

フィールド	説明
請求	プッシュ通知を送信するスケジュール済みジョブによって生成される識別番号。この数は、複数のプッシュスケジュール済みジョブ間で一貫性を確保するために [要求] フィールドに適用されます。
ペイロード	プッシュ通知のコンテンツ。
キュー数	システムがプッシュ通知を送信しようとした回数。プッシュ通知のステータスは、そのキュー数に関連しています。 • キュー数が予想よりも長く 0 になっている場合は、プッシュ通知を送信しようとしているスケジュール済みジョブはありません。 • キュー数が 0 より大きく、[タイプ] が [成功] の場合、これは最終的に送信する前にシステムがプッシュ通知を送信しようとした回数であると推測できます。 • キュー数が 10 に達すると、プッシュ通知の送信が停止されます。[タイプ]が [失敗] に変わります。
要求 ID	プッシュ通知の一意の識別番号。メールのメッセージ ID と同様に、要求 ID はプッシュ通知の相関トークンとして使用されます。
タイプ	プッシュ通知が送信されたかどうかを示すステータス。[タイプ] 列には、次の値を指定できます。 • **failure**：メッセージを送信できませんでした。 • **pending**：メッセージは処理のためにキューに入れられます。 • **success**：メッセージは正常に送信されましたが、モバイル端末で受信されたとは限りません。

関連トピック

スケジュール済みジョブ

システムメールログおよびメールボックス

システムメールログは、インスタンスが作成またはを受信するすべてのメールを記録します。システムメールボックスは、このログのフィルター済みのビューです。

メール [sys_email] レコードでは、インスタンスが作成または受信する各通知メールが記録されます。これらのレコードのログに移動できます。 システムログ > メール.

システムメールボックスは、メール [sys_email] テーブルのフィルター済みのビューです。インスタンスは、[タイプ] および [ステータス] フィールドの値に応じてシステムメールボックスに、メールレコードをアサインします。詳細については、「システムメールボックス」を参照してください。

以下のフィールドは、システムログのレイアウトと任意のシステムメールボックスに含めることができます。

メール ログ

フィールド	説明
メールボックス	このメールレコードを表示するシステムメールボックス。インスタンスは、[タイプ] および [ステータス] フィールドの値に応じて、このフィールドの値を設定します。
状況	メールの現在の状態 (エラー、無視、処理済み、または準備完了)。
受信タイプ	受信メールのタイプ (なし、転送、新規、または返信)。
タイプ	メールのステータス。選択肢は次のとおりです。 • received：サーバーでこのメールが受信されました。 • received - ignored：サーバーでこのメールが受信されましたが、受信メールアクションの目的のためにインスタンスによって無視されました。通常、これらのメールはスパムメールまたは自動返信のいずれかです。詳細については、エラー文字列フィールドを参照してください。 • send - failed：サーバーがメールを送信しようとしましたが失敗しました。詳細については、エラー文字列フィールドを参照してください。 • send - ignored：サーバーはこのメールの送信をスキップしました。通常、これは生成されたが受信者のメール アドレスがないメール、または重複したメール用です。詳細については、エラー文字列フィールドを参照してください。 • send - ready：メールは送信準備が整っていますが、メールサーバーによって送信されていません。通常、メールがこの状態になるのは短時間だけです。 • 送信 - 翻訳 - 準備完了：メールの翻訳中にメールが生成され、送信されます。通常、メールがこの状態になるのは短時間だけです。 • sent：エラーや問題なくインスタンスによってメールが送信されました。
ターゲット	特定のレコードの挿入、更新、または削除によってメールが生成された場合、レコードを参照するドキュメント ID。
ユーザー	メール通知が送信されたインスタンスの、ユーザーレコードからのユーザーの名前。 注: これは文字列フィールドです。

メール ログ (続く)

フィールド	説明
通知タイプ	通知のタイプ。選択肢は次のとおりです。 • なし • SMS • SMTP
UID	サーバー上で保存されているメールの一意の ID。
作成日時	インスタンスを実行しているマシンのロケールのメールアクティビティの日時。
削除済み	受信メールの場合は、メールがメールサーバーから削除されたかどうかを示します。
重み付け	同じテーブル上の他の通知との相対的な送信優先度を決定する、メールの重み。
重要性	緊急など、重要度が変更されたメールが送信されたことを表示。
元のイベントおよびメール通知	通知によって生成されたメールの場合、メールを作成したイベントと通知を格納している埋め込みリスト。
件名	メールの件名。通知の場合、件名テキストは システム通知 > メール > 通知.
エラー文字列	メールが送信されなかった理由を特定する、メール サーバーからキャプチャされたエラー文字列。これは、メールが送信に失敗した場合にのみ記録されます。
受信者	受信者のメールアドレス。
本文	生の HTML マークアップで表示されるメールの本文。関連リンクの HTML 本文のプレビューを使用して、本文テキストをレンダリングされた HTML として表示します。
コンテンツタイプ	メールのコンテンツ タイプ。
ヘッダー	メールに埋め込まれたヘッダー。

イベントログ

イベントログには、ServiceNow AI Platform 内で発生するすべてのシステムイベントが記録されます。

このログは、発生するすべてのイベントについて次の情報を提供します。

イベントログ

フィールド	説明
作成日時	インスタンスを実行しているマシンのロケールのイベントの日時。

イベントログ (続く)

フィールド	説明
名前	イベント登録にリストされているイベントの名前。
URI	イベントを生成した HTTP クエリ。
Parm1	イベントと受信者に依存するイベント固有の値。
Parm2	イベントと受信者に依存するイベント固有の値。
テーブル	このイベントで機能するデータベーステーブル。
処理済み	イベントの処理が開始された日時。この時刻は、インスタンスを実行しているマシンのロケールを反映しています。
処理時間	このイベントの処理に要した時間 (ミリ秒)。
キュー	プロセッサーキュー名。

インポートログ

インポートログには、プラットフォーム内のデータインポートアクティビティに関する情報が詳細形式で表示されます。

特定のログを生成したインポートセットの詳細については、次を参照してください。 インポートセット > 変換履歴.

このログは、すべてのインポートに次の情報を提供します。

インポートログ

フィールド	説明
作成日時	インスタンスを実行しているマシンのロケールのインポート日時。
レベル	表示されるメッセージのタイプ。インポートファイルの場合、レベルは情報です。
メッセージ	インポートのステータスに関するシステム生成メッセージ。
ソース	統合などのインポートの外部ソースの名前。

システム診断モジュール

システム診断アプリケーションは、プラットフォームに関連するログを提供します。

次のログを利用できます。

アップグレード履歴

インスタンスへのすべてのアップグレードを追跡します。

遅いクエリー

クエリーがプラットフォームのパフォーマンスにどのように影響するかについてのインサイトを提供します。「遅いクエリログの使用」を参照してください。

顧客アップデートテーブル

システムで行われた変更は、顧客アップデート [sys_update_xml] テーブルに時系列に記録されます。以下に示すように、いくつかの例外があります。

このテーブルに移動するには、ナビゲーションフィルターに「sys_update_xml.list」と入力します。Update Sets の詳細については、「System Update Sets」を参照してください。

更新ごとに次の情報が保存されます。

顧客アップデートテーブル

フィールド	説明
名前	更新されたレコードを識別する名前。
作成日時	顧客アップデートレコードが作成された日時。
作成者	変更を実行したユーザー。
タイプ	更新のタイプ。
更新日時	顧客アップデートレコードが更新された日時。
更新者	更新を実行したユーザー。
更新	レコードが更新された回数。
ターゲット名	変更された要素の名前。
表示	フォームレイアウトの変更である場合、変更されたフォームのビュー。
ペイロード	変更後のレコードの XML コンテンツ。
リモート更新セット	変更がリモート更新セットによって実行された場合は、その更新セットへの参照。
ローカル更新セット	変更が関連付けられている更新セット。

注: 顧客アップデートレコード (sys_update_xml) に表示されないアプリケーションの変更の例：

- updateSynch = false のメタデータのタイプ
- 表示名の変更など、テーブルやフィールドのカスケード変更
- 他のアプリケーションからの未解決のメタデータ参照 (要素に「表示値」がない)
- 結合ファイルの sys_id の変更
- sys_documentation を生成する可能性のあるフロー/フローアクションへの変更
- テーブルの作成時に生成される ua_table_license_config レコード
- 自然言語処理など、バックグラウンドで実行されているジョブ
- sys_update_xml が手動で変更または削除されるケース

詳細については、「変更を収容」を参照してください。

ログ履歴

古いログは、テーブルローテーションとテーブル拡張を使用してアーカイブされます。

デフォルトでは、次のスケジュールを使用して共通ログがアーカイブされます。

共通ログアーカイブスケジュール

テーブル	アーカイブスケジュール	ローテーション	タイプ
イベント [ecc_event]	毎日	7	ローテーション
キュー [ecc_queue]	毎日	7	ローテーション
イベント [sysevent]	毎日	7	ローテーション
ログ [syslog]	毎週	8	ローテーション
トランザクションログ [syslog_transaction]	毎週	8	ローテーション
メール [sys_email]	30 日ごと	8	拡張

ログファイルブラウザを使用する

インスタンスは、 ユーティリティログファイルブラウザとログファイルのダウンロードを提供します。

使用方法 システムログ > ユーティリティ > ノードログファイルブラウザ 任意のシステムログエントリを表示します。次のフィルターを使用して、ログファイルを検索できます。

ログファイルブラウザ

フィールド	説明
開始時間	インスタンスを実行しているマシンのロケールに対して検索する範囲の開始日時。
セッション ID	ログエントリを生成したセッションを識別する、システムによって生成された 16 進数の文字列。
終了時間	インスタンスを実行しているマシンのロケールに対して検索する範囲の終了日時。
メッセージ	オカレンスに関するシステム生成の説明。
レベル	表示されるメッセージのタイプ。レベルは、デバッグ、エラー、警告、および情報です。警告は、処理されて復旧されたエラーです。エラーは修正する必要があります。
スレッド名	ログファイルを作成したスレッドのシステム生成識別子。
最大行数	特定のフィルターに対して返されるレコードの最大数。

インスタンスは 2 日ごとにシステムログの圧縮アーカイブを作成し、21 日後にログアーカイブを消去します。

重要: ノードが廃止されると、ノードのログファイルはすぐに消去されます。つまり、さらに 21 日間アーカイブされることはありません。

ログファイルのアーカイブをダウンロードして、 システムログ > ユーティリティ > ノードログファイルのダウンロード. リストからログアーカイブを選択し、[関連リンク] の [ログをダウンロード] をクリックしてアーカイブを開くか保存します。

注: ログファイルは、現在ログインしているノードでのみ使用できます。現在ログインしているノードを確認するには、 システム診断 > 統計.

[トランザクション] フォームと [アクティブなトランザクション] フォームの新しい **[Syslog** レコードを表示] ボタンを使用して、トランザクションの実行中に生成されたシステムログエントリを表示します。トランザクションには、任意の数の syslog エントリを含めることができます。すべてのトランザクションに複数の syslog エントリがあると、トランザクションをそれぞれの syslog エントリに関連付けることが困難になります。**[Syslog** レコードを表示] UI アクションは、syslog テーブルをクエリするための URL を作成することで、アクティブなトランザクションと完了したトランザクションをそれぞれの syslog エントリに相互に関連付けるのに役立ちます。特定のトランザクションの正しい syslog エントリを特定すると、セキュリティ上の問題のデバッグと対処に役立ちます。

ログ記録のセキュリティの強化

ノードのログ行の [属性] フィールドを調べて、ログメッセージを生成したスクリプトまたはコンポーネントを特定します。トランザクションの開始行には、実行された要求のタイプを識別する新しいフィールドが含まれます。

新しい機能拡張を使用して、以下を実行します。

- 各ログ行のソースオリジネータをトレース
- オリジネータ情報が利用できない場合は、Java クラス名と属性を出力します。
- 各トランザクション行の先頭には、トランザクション ID とトランザクションタイプが含まれています。

各ログ行のトランザクション ID を使用して、各ログ行で指定された情報を把握します。トランザクションタイプを特定すると、各ログ行のオリジネータ情報を取得します。各ログ行のトランザクションタイプとオリジネータ 情報の両方が一緒に、各ノードログ行の必要なソース情報を提供します。

注: 属性の SYS_UI_MACRO および SERVICE_PORTAL_WIDGET スクリプトタイプは報告されません。

トランザクションタイプ

トランザクションタイプのリストを以下に示します。

- リスト
- フォーム
- XMLHttp
- レポート
- SOAP
- エクスポート
- スケジューラ
- TextSearch
- その他

- REST
- JSON
- AMB
- アーカイブ
- バッチ REST
- Instance Scan

システムプロパティ

この機能に必要なシステムプロパティは次のとおりです。

- Glide.log.append.attribution：このプロパティはデフォルトで有効になっています。各ノードラインの属性情報のオン/オフを切り替えます。
- Glide.db.log.append.classname.attribution：このプロパティはデフォルトで有効になっています。Java クラス名の属性のログ記録をオン/オフにします。

ログの改ざんの回避

システムログテーブルの保護ルールを設定して、アプリケーションログレコードの変更と削除の範囲を制限します。このルールを使用することで、これらのテーブルの変更や変更の試行のログ記録を決定できます。

security_admin であれば、Protected Tables プラグイン (com.glide.Protected_tables) をアクティブ化すると、プラットフォームで次のシステムログテーブルの更新、挿入、および削除操作を制限できます。

- syslog
- syslog_transaction
- sys_outbound_http_log
- sysevent
- sys_audit
- sys_push_notification
- syslog_app_scope
- protected_table_configuration (設定は変更不可)

> **注:** com.glide.protected_tables プラグインで保護されるのは上記のシステムログテーブルのみです。レコードを更新、挿入、または削除しようとすると、protected_table_log テーブルにメッセージが記録されます。

詳細については、「ログ保護プラグインのインストールと設定」を参照してください。

システムログテーブルごとに、次のいずれかのログ保護レベルを指定できます。

- 試行をブロックしてログ記録：変更をブロックして試行をログに記録する
- 試行のブロックのみ：変更をブロックして試行をログに記録しない
- 試行のログ記録のみ：変更はブロックしないが試行はログに記録する
- 試行をブロックおよびログに記録しない：変更をブロックせず、試行もログに記録しない

レコードが最初に作成された後に変更しようとすると、プラットフォームが各システムログテーブルに指定されたログ保護レベルを使用してその試行をブロックまたはログに記録します。

注: security_admin であれば、各システムログテーブルのデフォルトのログ保護レベルを上書きすることで、インスタンスのカスタマイズ内容に適合させることができます。

システムログテーブルの変更が試行された場合、その試行は protected_table_log テーブルにログに記録されます。

注: テーブルに保護レベルが指定されていない場合、変更の試行は protected_table_log テーブルに一切記録されません。

アドミンパネルのテーブルでプラグイン操作を無効にするには、com.glide.security.protected_table.enabled プロパティを false に設定します。詳細については、「ログ保護のプロパティの作成」を参照してください。

ログ保護プラグインの設定

各テーブルと操作の保護ルールを設定して、ログ保護プラグインの設定を完了します。

始める前に
必要なロール：security_admin

手順

1. 移動先 すべて > 保護テーブル > ログの保護.
   [管理パネル] ページが表示されます。

   注: Utah リリース以降、Protected Tables プラグインはデフォルトでインストールされていますが、無効になっています。

2. プラグインの設定を続行するには、ロールを security_admin に昇格させます。

   a. [システムアドミニストレーター] を選択します。

   b. [ロールを昇格] を選択します。
      ロールの昇格モーダルが表示されます。

   c. security_admin オプションを選択してロールを昇格し、[更新] を選択します。

3. 各テーブルと操作の保護ルールを設定します。保護ルールは、更新、挿入、削除に適用されます。一部のテーブルでは保護レベルを変更できません。syslog および syslog_app_scope テーブルには、更新および削除保護のための固定値があります。protected_table_configuration テーブルには、3 つの操作すべての固定の保護値があります。

   注: sysevent テーブルでは、挿入保護をブロックに設定できません。

   以前のリリースで syslog に対して [子テーブルに適用] をオンにした場合、Utah リリースへのアップグレード時に、syslog と同じ保護ルールで子テーブルがログ保護に追加されます。これは syslog でのみ発生し、他のテーブルでは発生しません。

4. [ログの保護を有効にする **(Enable Log Protection)**] トグルを選択して、機能を有効にします。

   注: このプラグインは、sys_properties テーブルの com.glide.security.protected_table.enabled プロパティを変更するだけで無効にできます。

ログ保護のプロパティの作成

ログの改ざんのリスクを回避するために、ログ保護のプロパティを作成します。

始める前に
必要なロール：admin

手順

1. *com.glide.security.protected_table.enabled* プロパティが [システムプロパティ] リストに存在しない場合は、[新規] を選択します。
   新しいシステムプロパティのフォームが表示されます。
2. フォームで、詳細を入力します。

フィールド	説明
名前	プロパティの名前 com.glide.security.protected_table.enabled
アプリケーション	このプロパティがあるアプリケーション。
説明	プロパティの説明
選択肢	
タイプ	値のタイプ：true または false
値	プロパティの実際の値
キャッシュを無視	キャッシュの内容を無視するオプション
プライベート	プロパティを非公開にするオプション ○ 読み込みロール ○ 書き込みロール

3. [送信] を選択してプロパティを作成します。

ログ記録、監査、およびエラー (インスタンスセキュリティ強化)

ログ記録と監査戦略を適用して、不審なアクティビティを適切なタイミングで特定して対処できるようにします。

インスタンスのログ記録の詳細については、「システムログ」を参照してください。を使用して、ログインや失敗したログインなどのシステムイベントをモニタリングするスケジュールがあることを確認します システムログ > イベント.

SQL エラーメッセージの無効化 (インスタンスセキュリティ強化)

glide.db.loguser プロパティを使用して、SQL エラーメッセージがブラウザーでレンダリングされないようにします。

詳細情報

属性	説明
プロパティ名	glide.db.loguser
構成タイプ	システムプロパティ (/sys_properties_list.do)
インスタンスセキュリティセンターで構成可能	いいえ

属性	説明
目的	ブラウザー内での SQL エラーメッセージの表示を無効にすること
タイプ	true | false
推奨値	false
機能への影響度	(低) この修正により、SQL エラーメッセージをレンダリングできなくなります。機能への影響はありません。
セキュリティリスク	(中) 攻撃者に役立つ可能性のある機密の SQL 情報は、Web ページのエラーメッセージに含まれて表示されることはありません。

システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

シークレット管理

ServiceNow シークレット管理を使用し、ビジネスニーズに合わせてパスワードへのアクセスを詳細に管理します。

重要:

シークレット管理は提供終了プロセスを開始しており、Yokohama リリースの時点で販売終了と更新終了のマイルストーンに達しています。双方向暗号化を使用したパスワードの保存については、「 Key Management Framework (KMF) による Password2 暗号化」を参照してください。

シークレット管理には 2 つのバージョンがあります。シークレット管理 Core は追加コストなしで ServiceNow プラットフォームに含まれます。シークレット管理 Enterprise は製品のプレミアムサブスクリプションバージョンです。これらのバージョンの違いの詳細については、「 シークレット管理の詳細」を参照してください。

自動翻訳

シークレット管理の主な機能とビジネス価値について学びます。	シークレット管理ダッシュボードを確認します。
構成  コア構成を計画します。	

シークレット管理の詳細

ServiceNow シークレット管理を使用し、ビジネスニーズに合わせてパスワードへのアクセスを詳細に管理します。

重要: アドミニストレーターには、シークレット管理に関連するモジュールとレコードを表示するためのロールが必要です。シークレット管理ロールの情報については、「シークレット管理ロール」を参照してください。

シークレット管理の **Core** バージョンと **Enterprise** バージョンからの選択

ビジネスニーズに応じて、[シークレット管理 Core] および [シークレット管理 Enterprise] から選択します。

シークレット管理コアプラグイン (com.glide.sm.core) はデフォルトで利用可能です。使用するためにインスタンスにインストールする必要はありません。シークレット管理 Enterprise プラグインは、 ServiceNow Vault v1 PROD18537 ライセンスでのみ使用できます。シークレット管理 Enterprise プラグインのサポートについては、カスタマーサポートにお問い合わせください。

シークレット管理 **Core**	シークレット管理 **Enterprise**
シークレット管理 Core は、追加コストなしでインスタンスにインストールできます。このプラグインは、ServiceNow アプリケーションエンジニアリングチームによって作成された ServiceNow プラットフォームで提供される非	シークレット管理 Enterprise には、アドミンがシークレットグループを作成して管理するために役立つ追加の機能が含まれています。Enterprise は、Core に一覧表示される機能に加えて次の機能を提供します。

シークレット管理 **Core**	シークレット管理 **Enterprise**

カスタムテーブルの基準を持つシークレットグループを使用する機能を提供します。	• 詳細なアクセス制御を使用し、次のいずれかの基準に基づいてシークレットグループを作成します。
	  ○ スコープ
	  ○ テーブル
	  ○ 列
	  ○ レコード
	• ServiceNow がアクセスできない独自のキーを使用して暗号化された、クライアントのアクセス可能なシークレットを作成します。
	• シークレット管理ダッシュボードを使用して、インスタンスに構成されているシークレットグループを確認し、潜在的なセキュリティの問題について確認します。
	注: シークレット管理 Enterprise は、ServiceNow 担当者が本番インスタンスで有効にする必要がある有料プラグインです。

シークレットグループを使用したシークレットの整理

シークレット管理を使用してシークレットをグループに整理し、グループレベルでそれらのシークレットにアクセスポリシーを適用します。

基本シークレットグループ

これらのグループは、スコープ内のすべてのシークレットに適用されます。これらのシークレットは、一般的な暗号化モジュールとモジュールアクセスポリシーによって復号化されます。

基準のあるシークレットグループ

基準のあるシークレットグループは、基本シークレットグループと同じように機能しますが、含まれるものが基準によってさらに絞り込まれています。次のような基準があります。

- アプリケーションスコープ
- パッケージ
- テーブル
- シークレット列
- レコードのフィルタリング

いずれかのタイプのシークレットグループを、インスタンスアクセス可能にするか、クライアントアクセス可能にすることができます。

インスタンス側のシークレットグループ

インスタンス側のシークレットグループには、インスタンスで復号化できるシークレットが含まれています。

クライアント側のシークレットグループ

クライアント側のシークレットグループは、公開鍵と秘密鍵のペアを使用して、クライアントのみがシークレットを復号化できるようにします。クライアントアクセス可能なシークレットグループを作成する場合は、公開鍵をインスタンスにアップロードし、秘密鍵を MID サーバーに保持します。インスタンスは公開鍵を使用してシークレットを暗号化しますが、秘密鍵を使用してのみ復号化できます。

注: これらのグループタイプの詳細については、「クライアント側 シークレット管理 について」を参照してください。

シークレットグループを使用したより詳細な制御

password2 は ServiceNow プラットフォームで利用できますが、シークレット管理は次の追加機能を提供します。

詳細なアクセス制御	Password2 password2 を使用すると、アドミニストレーターはアプリケーションスコープへのアクセスを制御できますが、スコープ内の要素へのアクセスを制限することはできません。 シークレット管理 シークレット管理を使用すると、アドミニストレーターは定義した基準に基づいてアクセスを制限できます。基準タイプは、パッケージ、テーブル、列などの基準に基づくことができます。
セキュアストレージ	クライアント側シークレットグループの場合、シークレット管理は新しい暗号化スキームを使用します。この暗号化スキームでは、ServiceNow は暗号化キーを保存しません。このため、データのセキュリティは ServiceNow セキュリティに依存しません。

グループへのモジュールアクセスポリシーの適用

シークレットをシークレットグループにグループ化した後、グループレベルでシークレットにアクセスする方法を決定するポリシーを適用できます。モジュールアクセスポリシーは、暗号化キーの有効期間などのインスタンスレベルでの制御を定義するために、暗号化モジュールに適用するアクセス制御メカニズムです。モジュールアクセスポリシーの詳細については、「モジュールアクセスポリシーの概要」を参照してください。

シークレット管理とともにインストールされるテーブル

シークレット管理は、次のテーブルを追加または変更します。

新しくなったテーブル	
[sn_sm_secret_group]	シークレットグループを格納します
[sn_sm_secret_group_criteria]	基準シークレットグループを格納します
[sn_sm_secret]	ラップされたシークレットを格納します
[sn_sm_identity_group]	ID のグループを公開鍵にマッピングするための ID グループを定義します
[sys_kmf_wrapped_module_key]	ラップされた対称暗号化キーを格納します
変更されたテーブル	

[sys_kmf_crypto_module]	追加された暗号化モジュールタイプ(ID 暗号化モジュールまたはシークレットグループ暗号化モジュール)
[sys_kmf_module_key]	• 概念的な秘密暗号化キー (キーマテリアルなし) を格納します • ID 公開鍵を格納します
[sys_kmf_crypto_caller_policy]	新しいモジュールアクセスポリシータイプを追加します

シークレット管理のユースケースの例

安全な ITOM ディスカバリー

このインフォグラフィックは、組織が ServiceNow IT Operations Management (ITOM) ディスカバリーを展開する方法を示す簡単なリファレンスアーキテクチャを示しています。インフォグラフィックに示されているように、複数の Windows および Linux サーバーが管理、計測、ディスカバリー (MID) サーバーに接続し、いくつかの MID サーバーエージェントがディスカバリープロセスで構成管理データベース (CMDB) を更新できるようにします。すべての MID サーバートランザクションには安全な認証が必要であるため、認証情報の管理はセキュリティの観点から極めて重要です。

統合ハブとのワークフロー接続を安全に高速化

ServiceNow の統合ハブを使用して、自動化されたアプリケーションプログラミングインターフェイス (API) を使用してさまざまなシステムに接続します。統合ハブが API を使用してシステムに接続するたびに、接続を確立するための認証情報が必要になります。多数のアプリケーションと接続用 API を管理するには、シークレット管理ソリューションが必要です。

シークレット管理は、組織のサイバーセキュリティを確保するための重要な部分です。暗号化キー、API トークン、パスワードなどのデジタル認証情報の作成、保存、送信、管理に関連するすべてのプロセスとツールが含まれています。シークレットを安全かつ効果的に管理するために、シークレットのライフサイクルのすべてのフェーズの標準ルールと手順を確立するコアシークレット管理ポリシーを構築できます。

クライアント側 シークレット管理 について

シークレット管理 を使用してシークレットとグループへのアクセスを管理する方法について説明します。

用語

クライアント側シークレット管理は、プロキシを使用せず、さらに復号化されたデータへ ServiceNow のアクセスを付与せずにシークレットを管理する方法を提供するように設計されています。このプロセスを理解するために、まず以下の暗号化の用語を理解してください。

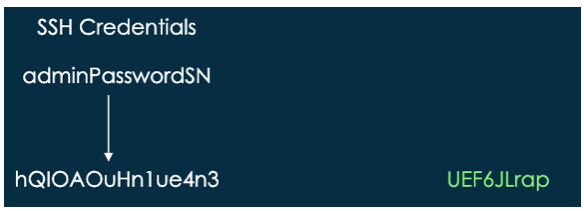
用語	説明
対称暗号化	対称暗号化では、データの暗号化と復号化の両方に 1 つの同じキーを使用します。データが対称キーで暗号化されている場合は、このキーだけで復号化できます。
対称キー	対称キーはシークレットを暗号化し、クリアテキストのパスワードを判読不可能な暗号テキストに変換します。
非対称暗号化	非対称暗号化では、2 つのキーを使用し、1 つは暗号化に、もう 1 つは復号化に使用します。

用語	説明
公開鍵	公開鍵は非対称キーペアの片方のキーです。このキーは、インスタンスに格納され、 キーを使用して対称キーを暗号化します。この暗号化された対称キーは、秘密鍵とペアになっている場合にのみ復号化できます。
秘密鍵	秘密鍵は非対称キーペアの片方のキーです。このキーは、MID サーバーのキーストアに格納されます。ServiceNow はこのキーにアクセスできません。 公開鍵と組み合わせて、非対称キーペアがシークレットを復号化するために使用されます。

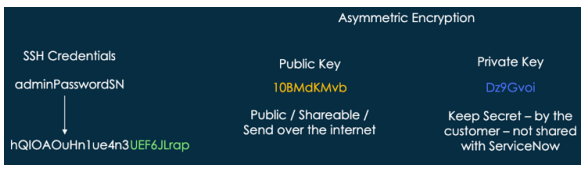
クライアント側の暗号化プロセス

対称キーは、認証情報 (この場合は admin パスワード) を暗号化し、読み取り可能なクリアテキストから暗号化された暗号テキストへと変換します。

対称キー (緑色で表示) を認証情報に適用して、認証情報を暗号化または復号化できます。

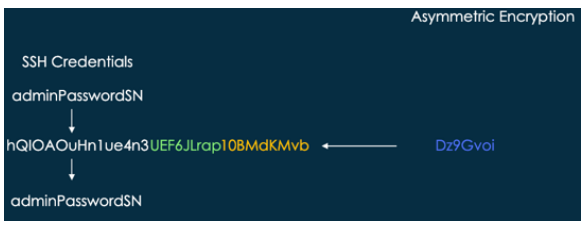


この時点で、非対称暗号化で公開鍵 (黄色) と秘密鍵 (青色) の使用が開始されます。



公開鍵は、対称キーとともに認証情報を暗号化します。対称キーは現在保護されているため、認証情報の復号化に使用できません。公開鍵はこの暗号化を実行できますが、単独で復号化に使用することはできません。

公開鍵で暗号化した後は、認証情報を復号化するために秘密鍵が必要です。顧客のみがこのキーを持っているため、暗号化された認証情報にアクセスできるのは顧客だけです。



クライアントのアクセス可能なシークレットの構成

クライアントのアクセス可能なシークレットを使用するようにインスタンスを構成する方法について説明します。

この実装例を使用して、プロキシを使用せずにシークレット管理を構成したり、復号化されたデータへの ServiceNow アクセス権を付与したりします。

クライアント側のシークレット管理を使用してパスワードとグループへのアクセスを管理する方法の詳細については、「クライアント側 シークレット管理 について」を参照してください。

これらの手順では、ローカルネットワークに MID サーバーが構成されていることを前提としています。プロセスの詳細については、「MID サーバー」を参照してください。

プロセスの概要

1. 暗号化キーと証明書の作成

   ローカル環境でターミナルコマンドを使用して暗号化キーと証明書を作成します。

2. ServiceNow の信頼できるキーストアに証明書を追加する

   キーと証明書を ServiceNow の信頼できるキーストアにアップロードします。

3. 基準のあるシークレットグループの作成

   シークレットのグループを作成します。シークレットグループを使用してシークレットをグループに整理します。これらのグループを使用して、グループレベルでこれらのシークレットにアクセスポリシーを適用できます。次に、シークレットグループを ID グループに関連付け、MID サーバーをその ID グループに追加します。

4. 公開鍵/秘密鍵ペアの MID サーバーへのアップロード

   公開鍵/秘密鍵ペアを MID サーバーにアップロードします。このキーペアにより、MID サーバーはインスタンスからの認証要求を処理できます。

5. 認証情報の作成と認証情報の暗号化のテスト

   サードパーティシステムに対して認証するための認証情報を作成し、ServiceNow でその認証情報にアクセスできないことをテストします。

6. 統合を管理するためのフローデザイナーの構成

   インスタンスで フローデザイナー を使用して、ローカルネットワークとインスタンス間の統合を管理します。

7. エンドツーエンドのクライアント側で暗号化されたシークレットの統合をテストする

   統合をテストし、実行の詳細を確認して、構成が機能していることを確認します。

暗号化キーと証明書の作成

ローカル環境でターミナルコマンドを使用して暗号化キーと証明書を作成します。

始める前に

必要なロール：なし

手順

1. ローカル環境で、ターミナル (Mac または Linux の場合) またはコマンドライン (Windows の場合) を開きます。
2. ターミナルを使用し、`cd` を使用して暗号化キーを保存するフォルダーに移動します。
3. ターミナルを使用して、次のように入力します。

   ```
   openssl req -newkey rsa:4096 -nodes -keyout sm_private_key.pem -x509 -days 365 -out
    sm_public_cert.pem
   ```

   注: この例では、OpenSSL を使用してキーと証明書を生成します。要件に基づいて、他の同等のツールに置き換えることもできます。

   このコマンドは、秘密鍵と公開証明書 (一致する公開鍵を含む) を生成します。「Country Name」で始まる、必要な情報の一連のプロンプトが表示されます。
4. 要求された情報をプロンプトに入力します。

次のプロンプトが表示されます。

- 国名
- 都道府県名
- 市区町村名 (市区町村など)
- 組織名 (会社など)
- 組織単位名 (部門など)
- 共通名 (完全修飾ホスト名など)
- メールアドレス

セキュリティチームと協力して、正しい証明書情報が入力されていることを確認します。

```
Country Name (2 letter code) []:US
State or Province Name (full name) []:CO
Locality Name (eg, city) []:Boulder
Organization Name (eg, company) []:ServiceNow
Organizational Unit Name (eg, section) []:Product Management
Common Name (eg, fully qualified host name) []:fake@servicenow.com
Email Address []:fake@servicenow.com
```

5. ステップ 2 で選択したフォルダーをチェックして、秘密鍵と公開証明書が作成されたことを確認します。
   ステップ 3 の例と同じファイル名を使用した場合は、次のファイルが表示されます。
   - sm_private_key.pem
   - sm_public_cert.pem
6. 同じフォルダーで、次のコマンドを使用します。

   重要: 使用する特定のコマンドは、オペレーティングシステムによって異なります。

Linux の場合：	`cat sm_private_key.pem sm_public_cert.pem > sm_keypair_bundle.pem`
Windows の場合：	`sm_private_key.pem sm_public_cert.pem > sm_keypair_bundle.pem`

   このコマンドにより秘密鍵と公開証明書が単一のファイルにバンドルされ、後の手順で MID サーバーにロードされます。

7. フォルダをもう一度確認して、秘密鍵 (sm_keypair_bundle.pem) と公開証明書を含む新しいファイルが作成されたことを確認します。

ServiceNow の信頼できるキーストアに証明書を追加する

キーと証明書を ServiceNow の信頼できるキーストアにアップロードします。

始める前に
必要なロール：admin

この例で作成した公開証明書は、「自己署名」証明書と見なされます (つまり、信頼できるルート機関からのものではありません)。証明書を使用するには、証明書を ServiceNow の信頼できるキース

トアに追加する必要があります。認証局からの証明書を使用する場合は、この手順を完了する必要はありません。

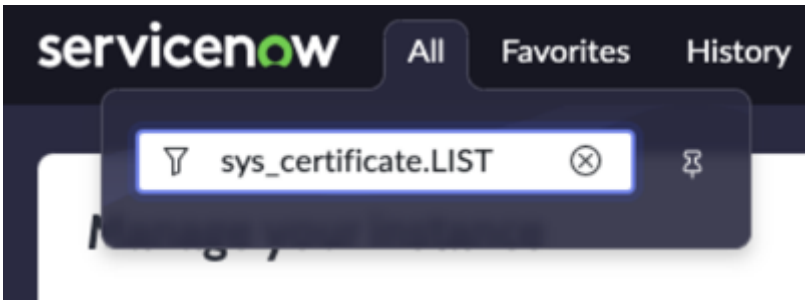
手順

1. ローカル環境で、ターミナル (Mac または Linux の場合) またはコマンドライン (Windows の場合) を開きます。
2. ターミナルを使用し、`cd` を使用して暗号化キーを作成したフォルダーに移動します。
3. ターミナルで、次のコマンドを入力します。

   ```
   cat sm_public_cert.pem
   ```

   公開証明書の内容を表示して、信頼できるキーストアにコピーする必要があります。この `cat` コマンドは証明書を表示します。

4. 証明書情報をクリップボードにコピーします。
   `-----BEGIN CERTIFICATE-----` 行で始まり `-----END CERTIFICATE-----` 行で終わるようにします。
5. ServiceNow インスタンスで、ナビゲーションフィルターに「`sys_certificate.LIST`」と入力し、**[X.509** 証明書] リストに移動します。



6. [新規] を選択して、**[X.509** 証明書] レコードを作成します。
7. フォームで、次のフィールドに入力します。

X.509 証明書のフィールド

フィールド	値
名前	証明書の名前。この名前は任意の名前にすることができます。
フォーマット	**PEM** を選択します。 注: Privacy Enhanced Mail (PEM) ファイルは、キーと証明書に使用される公開鍵インフラストラクチャ (PKI) ファイルの一種です。前のステップで作成したレコードはこのファイルタイプです。
タイプ	[信頼できるストア証明書 **(Trusted Store Cert)]** を選択します。
簡単な説明	証明書の説明。この証明書の使用目的を示す値を入力します。
PEM 証明書	ステップ 4 でコピーした証明書情報を貼り付けます。

8. [送信] をクリックしてレコードを保存します。

基準のあるシークレットグループの作成

シークレットのグループを作成します。シークレットグループでシークレットをグループに整理し、グループレベルでそれらのシークレットにアクセスポリシーを適用できるようにします。次に、シークレットグループを ID グループに関連付け、MID サーバーをその ID グループに追加します。

始める前に
必要なロール：admin、KMF_admin、sn_secrets.secret_manager、および sn_kmf.cryptographic_manager

手順

1. 移動先 すべて > シークレット管理 > 基準のあるシークレットグループ.
2. [新規] を選択して、[基準のあるシークレットグループ] を作成します。
3. フォームで、次のフィールドに入力します。

基準のあるシークレットグループフィールド

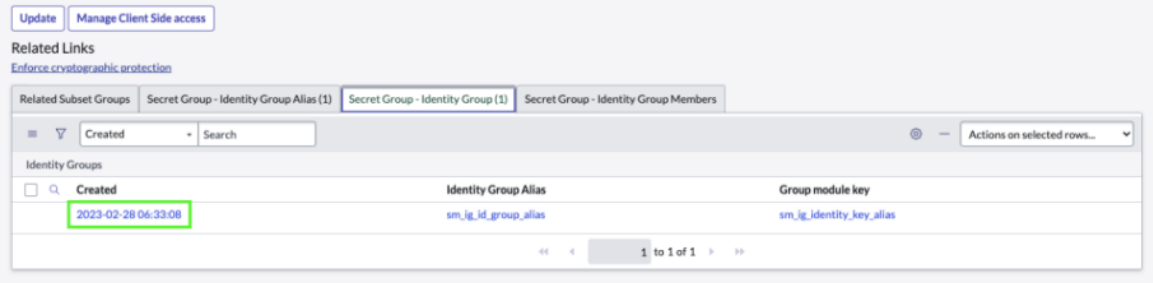
フィールド	値
グループ名	シークレットグループの名前。この名前は任意の名前にすることができます。
シークレットタイプ	[クライアントアクセス可能] を選択します。
自動生成モジュール	ボックスをオンにします。
簡単な説明	シークレットグループの説明。このグループの使用目的を示す値を入力します。
基準タイプ	[ターゲットテーブル] を選択します。
ターゲットテーブル	**SSH** 認証情報 **[ssh_credentials]** を選択します。

4. フォームヘッダーを長押し (または右クリック) し、コンテキストメニューから [保存] を選択してレコードを保存します。
5. [アクティブ] チェックボックスがオフになっていることを確認します。
6. [クライアント側のアクセスを管理] ボタンを選択して、ID グループを作成します。
   [ID グループエイリアス] ウィンドウが表示されます。
7. [新規] ボタンを選択します。
8. **[ID** グループエイリアス] フィールドの横にある参照アイコン ( ) を選択します。
9. [グループエイリアス名] フィールドに値を入力します。
   この名前は任意の値にすることができます。
10. ステップ 9 で一意の名前を選択した場合は、[送信] を選択します。
11. **[ID** キーをアップロード **(Upload identity key)]** ボタンを選択します。
    [ID 公開鍵証明書をインポート] ウィンドウが表示されます。
12. **[ID** キーエイリアス] フィールドに値を入力します。

> **重要:** この値は任意の値にすることができますが、後のステップで MID サーバーに挿入する値と完全に一致する必要があります。

13. [インポート] を選択します。
    [添付ファイル] ウィンドウが表示されます。
14. [ファイルを選択] を選択します。
15. 前のステップで作成した公開証明書を選択します。
    この証明書は sm_public_cert.pem ファイルである必要があります。
16. [X] アイコンを選択してウィンドウを閉じます。
17. **[OK]** を選択して **[ID** 公開鍵証明書をインポート] ウィンドウを閉じます。
    インポートの成功を確認する青色の [キーと証明書がインスタンスに正常にインポートされました] バナーが表示されます。
18. [送信] を選択します。
    [ID グループ **(Identity groups)]** リストが表示されます。
19. リストの ID グループレコードの左側にあるチェックボックスをオンにします。

20. [シークレットグループの関連付け **(Associate secret group)**] ボタンを選択します。
[基準のあるシークレットグループ] レコードに戻ります。[シークレットグループ **- ID** グループ エイリアス **(Secret Group – Identity Group Alias)**] および [シークレットグループ **- ID** グループ **(Secret Group – Identity Group)**] 関連リストが表示されます。これらの関連リストには、前のステップで作成したレコードが表示されます。
21. [シークレットグループ **- ID** グループ **(Secret Group – Identity Group)**] 関連リストで、そのリストのレコードの [作成日時 **(Created)**] フィールドを選択します。



[ID グループ **(Identity groups)]** レコードが表示されます。

22. **[ID** グループメンバー **(Identity group members)]** 関連リストで、[新規] ボタンを選択します。
[ID グループメンバー **(Identity group members)]** レコードが表示されます。
23. [メンバーテーブル] フィールドで、**[MID** サーバー **[ecc_agent]]** を選択します。
24. **[ID** グループメンバー **(Identity group members)]** レコードフィールドの横にある参照アイコン ( ) を選択し、MID サーバーを選択します。

> **注:** [すべてのレコードを含める **(Include all records)**] チェックボックスをオンにすると、インスタンスに接続されているすべての MID サーバーが ID グループに追加されます。

25. **[OK]** を選択して、[ドキュメントを選択] ウィンドウを閉じます。
26. **[Submit** (送信)] を選択します。
27. に戻る すべて > シークレット管理 > 基準のあるシークレットグループをクリックし、ステップ 2 で作成したレコードを開きます。
28. [アクティブ] フィールドを有効にします。
29. [更新] をクリックしてレコードを保存します。

公開鍵/秘密鍵ペアの **MID** サーバーへのアップロード

公開鍵/秘密鍵ペアを MID サーバーにアップロードします。このキーペアにより、MID サーバーはインスタンスからの認証要求を処理できます。

始める前に
必要なロール：なし

ServiceNow は秘密鍵にアクセスできないため、公開鍵とペアリングして対称キーを復号化してから認証情報を復号化することはできません。MID サーバーがこの暗号化された認証情報を使用しようとしても、秘密鍵にアクセスせずに認証用の認証情報を復号化することはできません。

これらのステップでは、秘密鍵を MID サーバーにアップロードして公開/秘密キーチェーンを完了します。このアップロードでは、ServiceNow にアクセス権を付与せずに MID サーバーにアクセス権を付与します。

MID サーバーに秘密鍵へのアクセス権を付与するには、PowerShell でアドミニストレーターとして実行するコマンドをビルドする必要があります。この例では、コマンドは Azure Windows 仮想マシン用です。

重要:

これらのステップを実行するシステムで MID サーバーとキーペアファイルの両方にアクセスできることを確認してください。

手順

1. ローカル環境で、暗号化キーと証明書の作成 のステップでキーペアを作成したフォルダーを見つけます。
2. 完全パスを見つけて manage-certificates.bat ファイルにコピーします。

   注: このファイルは MID サーバーフォルダーにあります。MID サーバーフォルダーを保存した場所に応じて、パスは次の例のようになります。

   ```
   C:
   \Users\<your_user_account>\Documents\SM_Implementation\mid.utah-07-08-2022__patch4b01-31-2023_02-07-2023_1702.windows.x86-64\sm_ig_MIDS\bin\scripts\manage-certificates.bat
   ```

3. テキストファイルを作成し、パスをファイルに貼り付けます。
4. テキストファイルで、パスの後に次を追加します。
   -a your_identity_key_alias

   your_identity_key_alias を、公開証明書のアップロード時に作成した ID キーエイリアスの名前に置き換えます。
5. キーペアファイルの完全パスを見つけてコピーします。

   注: これらのステップの名前を使用した場合、そのファイルの名前は sm_keypair_bundle.pem になります。

6. テキストファイルで、このパスを行の最後に追加し、このパスと前の情報の間にスペースを追加します。
   テキストファイル内のテキストは、次の例のようになります。

   ```
   C:
   \Users\<your_user_account>\Documents\SM_Implementation\mid.utah-07-08-2022__patch4b01-31-2023_02-07-2023_1702.windows.x86-64\sm_ig_MIDS\bin\scripts\manage-certificates.bat -a your_identity_key_alias
    C:\Users\<your_user_account>\Desktop\sm_keypair_bundle.pem
   ```

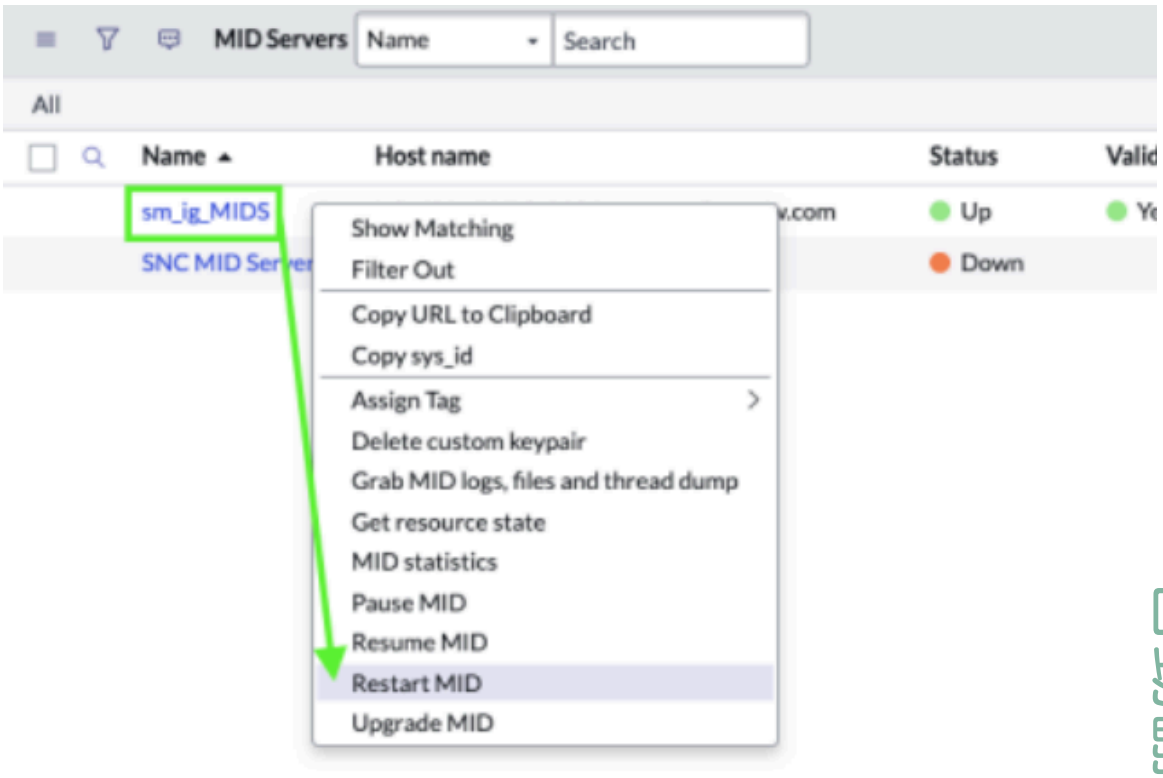
   注: この例では、パスを短くするために sm_keypair_bundle.pem ファイルがデスクトップにあります。

7. テキストファイルのテキスト全体をクリップボードにコピーします。
8. システムで Powershell を見つけ、[アドミニストレーターとして実行 **(Run as Administrator)]** オプションを選択します。
9. テキストファイルのテキストを Powershell に貼り付け、Enter キーを押します。
   成功すると、次のメッセージが表示されます。

   ```
   Installed certificate with alias: <your_identity_key_alias> into the MID keystore.
   ```

ヒント: このメッセージが表示されない場合は、コマンドにエラー、スペース、または不要な引用符が含まれていないことを確認してください。完全パスが正しく入力されていることを確認します。

10. MID サーバーレコードに移動し、レコードを右クリックして **[MID を再起動]** を選択して、MID サーバーを再起動します。



MID サーバーを再起動すると、アップロードされたキーペアが MID キーストアに同期され、操作で使用できるようになります。続行する前に、MID サーバーが再起動し、ステータスが [稼働中 **(Up)]** で検証済みの値が [はい] になるまで待機します。

認証情報の作成と認証情報の暗号化のテスト

サードパーティシステムに対して認証するための認証情報を作成します。

始める前に

必要なロール：admin、KMF_admin、sn_secrets.secret_manager、および sn_kmf.cryptographic_manager

重要: MID サーバーのログインユーザーには、admin、KMF_admin、sn_secrets.secret_manager、および sn_kmf.cryptographic_manager のロールも必要です。

手順

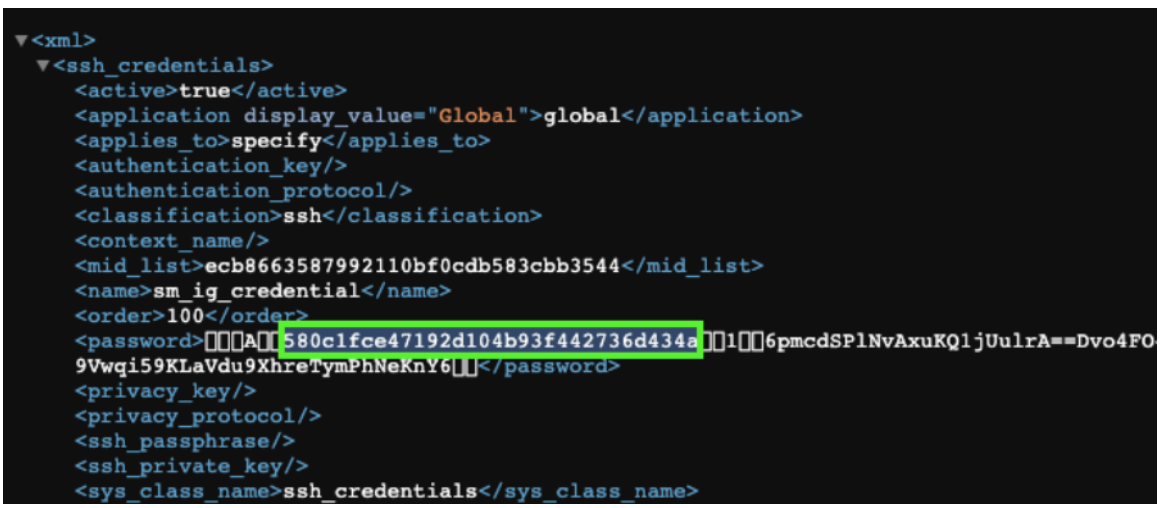
1. ナビゲーションフィルターに「ssh_credentials.list」と入力して、**[SSH** 認証情報] リストに移動します。
2. [新規] を選択して、**[SSH** 認証情報] レコードを作成します。

3. フォームで、次のフィールドに入力します。

SSH 認証情報フォーム

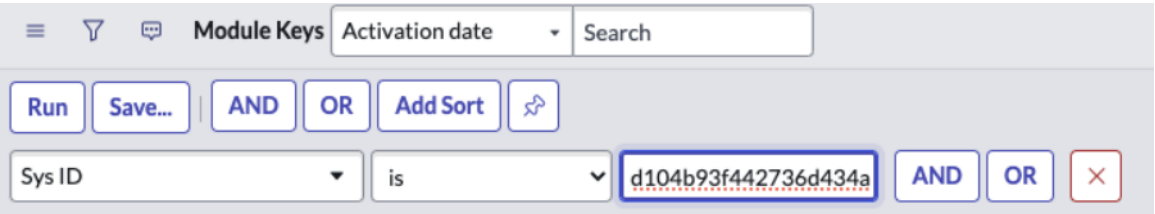
フィールド	値
名前	認証情報レコードの名前を入力します。この名前は任意の値にすることができます。
適用先	[特定の **MID** サーバー] を選択します。
MID サーバー	MID サーバーを選択します。
ユーザー名	ユーザー名を入力します。
パスワード	ユーザーに関連するパスワードを [ユーザー名] フィールドに入力します。

4. [認証情報エイリアス] フィールドの横にあるロックアイコン ( ) を選択します。
5. 参照アイコン ( ) を選択して、[接続および資格情報エイリアス] リストを開きます。
6. [新規] を選択して、[接続および資格情報エイリアス] レコードを作成します。
7. [名前] フィールドに名前を入力します。
8. [タイプ **(Type)**] フィールドで [認証情報 **(Credential)**] を選択します。
9. [送信] を選択します。
   [SSH 認証情報] レコードに戻ります。次のステップでは、認証情報が暗号化されていることをテストします。
10. フォームヘッダーを選択して長押し (または右クリック) し、**[XML** 表示] を選択します。
11. XML 内で <パスワード> XML タグを見つけます。
12. この <パスワード> タグ内の sys_id をクリップボードにコピーします。
    sys_id は、この認証情報の暗号化に使用されている対称キーを表す 32 文字のコードです。SSH 認証情報テーブルに入力した暗号化パスワードは、この同じ行で 2 つのボックスセットの右側にあります。



13. ナビゲーションフィルターに「sys_kmf_module_key.list」と入力して、[モジュールキー] リストに移動します。

14. **[Sys ID]** フィールドがステップ 12 でコピーした sys_id と一致するレコードのリストをフィルタリングし、[実行] を選択します。



検索すると、単一のモジュールキーレコードが返されます。このレコードは、対称キーを正常に作成して使用していることを示しています。

15. ナビゲーションフィルターに「sys_kmf_wrapped_module_key.list」と入力して、[ラップされたモジュールキー] リストに移動します。

16. [暗号化されたモジュール] フィールドが前のステップで作成した暗号化モジュールと一致するレコードのリストをフィルタリングし、[実行] を選択します。
検索すると、単一の [ラップされたモジュールキー] レコードが返されます。このリストから、次のことを確認できます。

    - [ラップされたキーマテリアル] 列には、(SSH 認証情報の暗号化に使用されている) 暗号化モジュール内の対称キーが、ID グループにアップロードした公開鍵によって暗号化されていることが表示されます。
    - [ラップされたキーの **Sys id]** フィールドは、[ラッピングキー **sys id]** (ID グループにアップロードされた公開鍵) によって暗号化されているキー (暗号化モジュール対称キー) であることを示しています。

    上記のフィールドがデフォルトでリストにない場合は、[リストをカスタマイズ] アイコン (⚙) を選択してリストに追加できます。

    > **注:** インスタンスは 10 分ごとにクリーンアップジョブを実行します。これにより、孤立キーが削除され、認証情報の更新後に関連付けられていないキーが急増するのを防ぐことができます。

統合を管理するためのフローデザイナーの構成

インスタンスで フローデザイナー を使用して、ローカルネットワークとインスタンス間の統合を管理します。

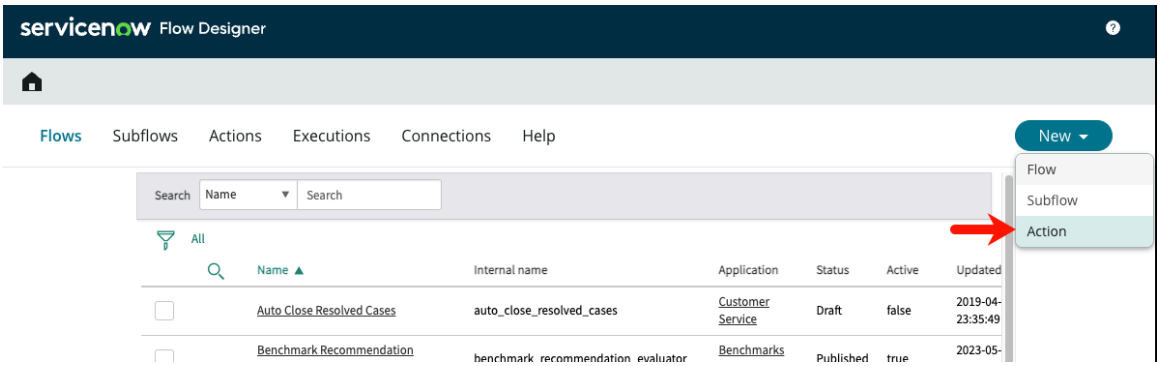
始める前に

必要なロール：admin、KMF_admin、sn_secrets.secret_manager、および sn_kmf.cryptographic_manager

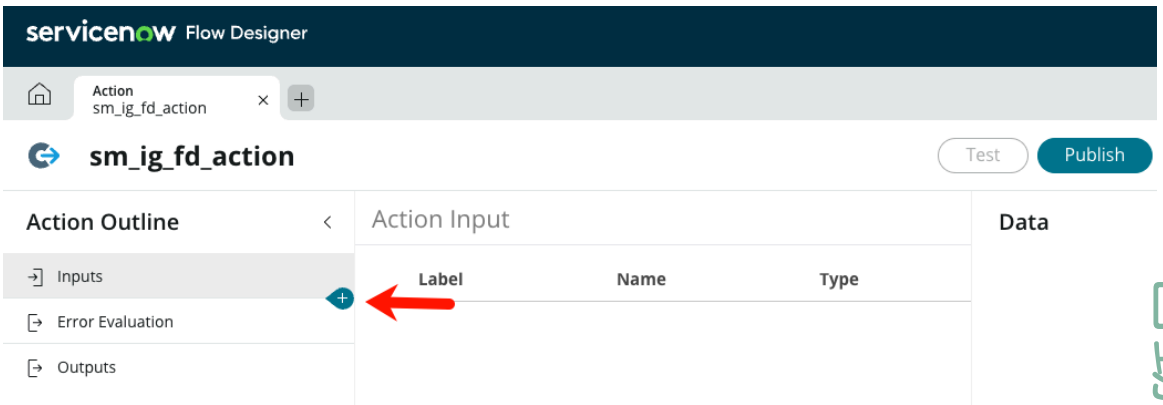
これらの手順では、フローデザイナー ワークフローを作成して、ローカルシステムにテキストファイルを作成します。

手順

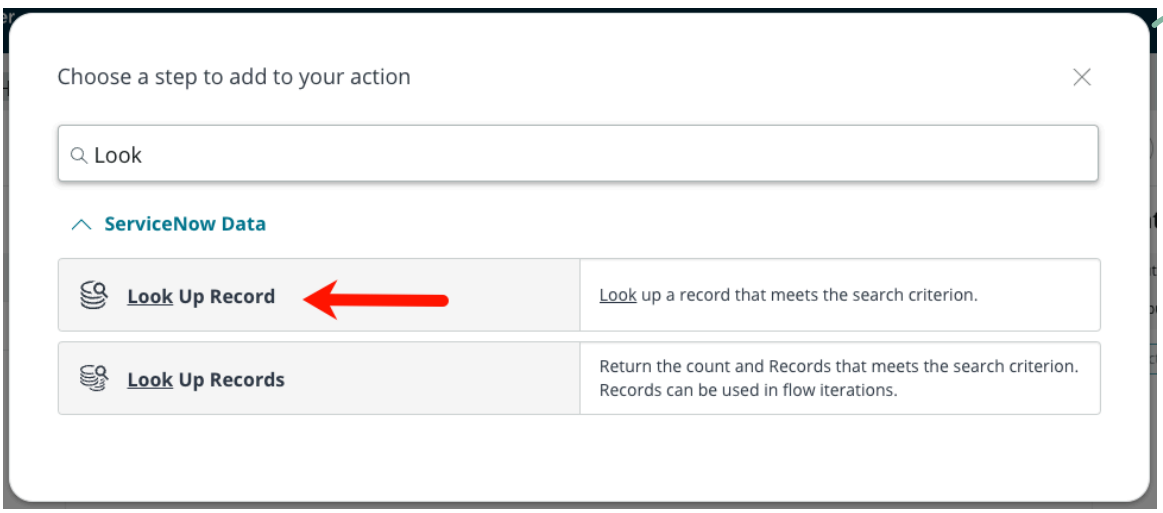
1. インスタンスで、 すべて > プロセス自動化 > フローデザイナー.
2. [新規] を選択し、[アクション] を選択して ServiceNow にアクションを作成します。



3. [アクション名] フィールドに名前を入力し、[送信] を選択します。
4. [アクションアウトライン] で [入力] と [エラーの評価] の間のプラス記号を選択して、ステップを作成します。



5. [アクションに追加するステップを選択] ウィンドウで、[レコードの検索] を選択します。



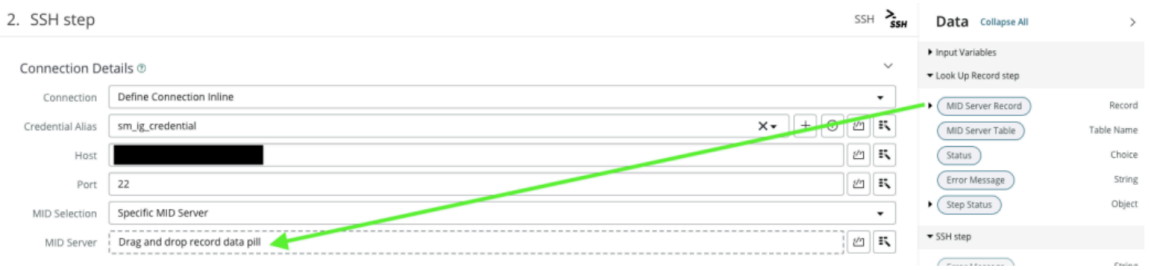
6. [レコードの検索 **(Look Up Record)]** ステップセクションで、[テーブル] フィールドで **[MID** サーバー **[ecc_agent]]** を選択します。
7. [レコードの検索 **(Look Up Record)]** ステップの下にあるプラス記号を選択して、別のステップを作成します。
8. [アクションに追加するステップを選択] ウィンドウで、**[SSH]** を選択します。

注: **[SSH]** オプションが表示されない場合は、必要なプラグインを有効にする必要があります。

9. **[SSH** ステップ **(SSH Step)]** セクションで、次の情報を入力します。

フィールド	値
接続	[インラインで接続を定義] を選択します。
認証情報エイリアス	前のステップで作成した SSH 認証情報の認証情報エイリアスを選択します。
ホスト	SSH 経由で接続するホストの IP アドレスを入力します。
ポート	「22」と入力します。
MID 選択	[特定の **MID** サーバー] を選択します。

10. **[MID** サーバー] フィールドに入力するには、[データ] セクションから **[MID** サーバーレコード **(MID Server Record)]** ピルをフィールドにドラッグします。



警告: ピルをフィールドにドラッグするときは、ピルの横にある黒い矢印ではなくピルを選択します。

11. **[SSH** 構成 **(SSH Configuration)]** セクションで、[コマンド] フィールドに次の値を入力します。

```
/bin/date > sm_ig_text_file.txt
```

このコマンドは、MID サーバーから復号化されたシークレットを使用して、ローカルシステムにテキストファイルを作成します。MID サーバーは、復号化されたシークレットへのアクセス権を ServiceNow インスタンスに付与せずに、ServiceNow インスタンスへのアクセス権を (フローデザイナーを介して) 付与します。

ヒント: `/bin/date` コマンドは、作成されたテキストファイルに現在の日付/時刻を挿入します。このコマンドは、テキストファイルが作成された日時と現在の日時に基づいて、統合がリアルタイムで行われていることを示します。

12. [保存] ボタンを選択してワークフローを保存します。

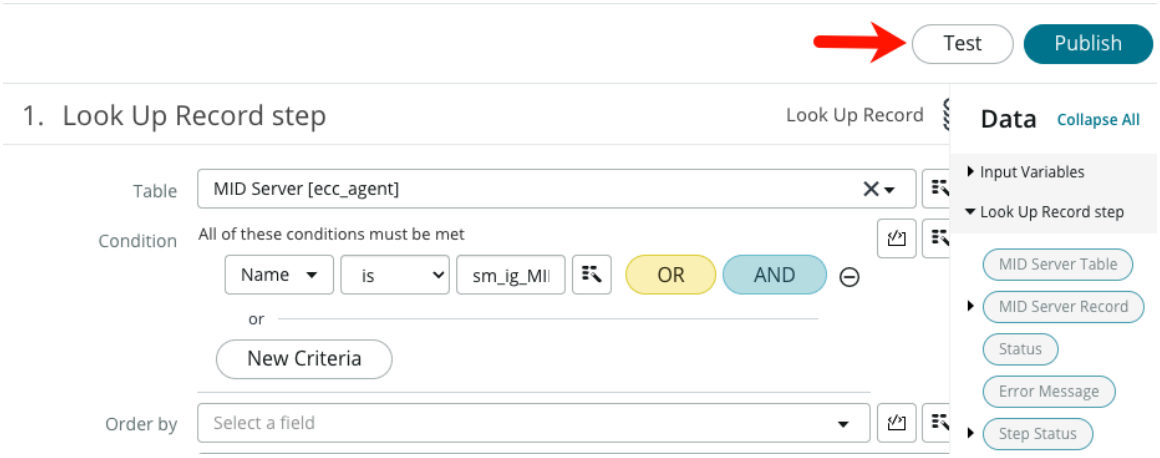
エンドツーエンドのクライアント側で暗号化されたシークレットの統合をテストする

統合をテストし、実行の詳細を確認して、構成が機能していることを確認します。

始める前に
必要なロール：admin、KMF_admin、sn_secrets.secret_manager、および sn_kmf.cryptographic_manager

手順

1. フローデザイナー で、画面の右上隅にある [テスト] ボタンを選択します。



2. [テストアクション] ウィンドウで、[テストを実行] を選択します。
3. [テストの実行が終了しました。アクションの実行の詳細を表示します **(Your test has finishe drunning. View the action execution details)]** を選択します。
4. 画面の右上に [テスト実行 – 完了] が表示されるまで、画面を更新します。
5. 画面の左下にある [ステップ] 矢印を選択します。
6. [ステップ出力データ] の見出しに次の成功メッセージが表示されるまで下にスクロールします。

```
{"Step Status";{"code";0,"message";"Success"}}
```

7. フローデザイナー でこのメッセージが表示されたら、テキストファイルがローカルシステムに作成されていることを確認します。

シークレット管理で暗号化された **Windows Management Instrumentation** 認証情報のテスト

Windows Management Instrumentation (WMI) 認証情報がシークレット管理で暗号化されていることを確認し、統合ハブワークフローを使用してエンドツーエンドのテストを実行します。

始める前に
必要なロール：admin、KMF_admin、sn_secrets.secret_manager、および sn_kmf.cryptographic_manager

- 統合ハブエンタープライズがインスタンス上でアクティブになっている必要があります。詳細については、「統合ハブ プラグインを要求する」を参照してください。
- インスタンスにシークレットグループを構成する必要があります。「クライアントのアクセス可能なシークレットの構成」にリストされているステップ 1 から 3 を参照してください。
- シークレット管理で暗号化された構成済みの認証情報が必要です。「認証情報の作成と認証情報の暗号化のテスト」にリストされているステップ 1 から 9 を参照してください。

手順

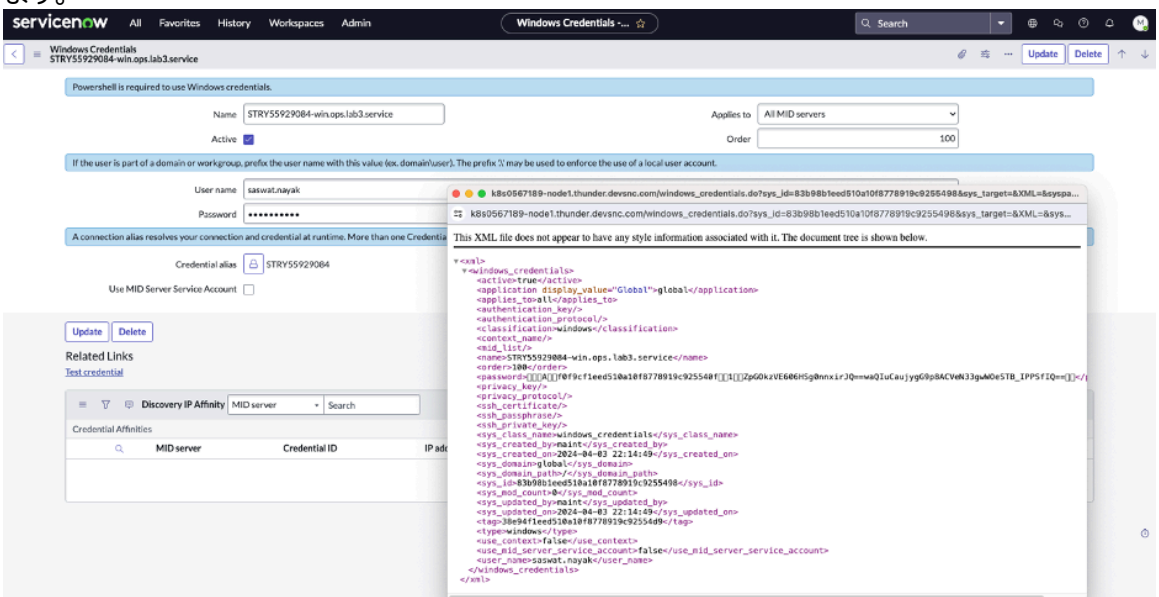
1. 目的の MID サーバーがシークレットグループに関連付けられていることを確認します。
   クライテリアのあるシークレットグループ [sn_sm_criteria_secret_group] レコードを開き、[シークレットグループ：**ID** グループメンバー] リストで MID サーバーを探します。MID サーバーがグループに関連付けられていない場合は、「基準のあるシークレットグループの作成」を参照してください。
2. 認証情報がシークレット管理で暗号化されていることを確認します。

   a. 移動先 すべて > 統合ハブ > 接続 **&** 認証情報 > 認証情報.

   b. 認証情報レコードを開きます。
   [認証情報エイリアス] フィールドに認証情報エイリアスが表示されていることを確認します。まだ作成していない場合は、「接続情報および認証情報エイリアスの作成」を参照してください。

   c. 認証情報レコードのヘッダーを選択して長押し (または右クリック) し、リストから **[XML** 表示] を選択します。

   d. レコードの XML ビューでパスワードを検索し、値の先頭が □□□A□□ であることを確認します。



3. ワークフロースタジオでテストワークフローを作成します。

   a. 移動先 すべて > プロセス自動化 > ワークフロースタジオ.

   b. [新規] を選択し、リストから [アクション] を選択します。

   c. [アクション名] フィールドに名前を入力します。
   [アプリケーション] フィールドは [グローバル] に設定されたままにします。

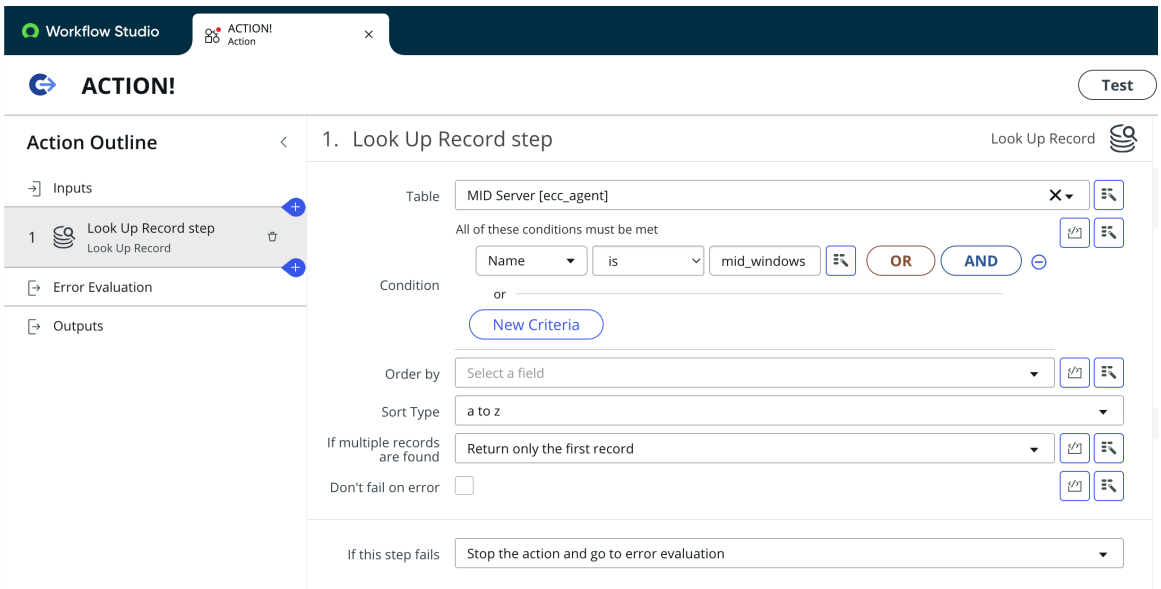
   d. [アクションを構築] ボタンを選択します。

4. ワークフロースタジオでレコードルックアップステップを構成します。

a. [アクションアウトライン] で、プラスボタンを選択し、新しいステップを追加します。
[レコードをルックアップ] ステップタイプを探して選択します。

b. [テーブル] フィールドで、**[MID** サーバー **[ecc_agent]]** を選択します。

c. 条件には、[名前] [次の値に等しい **(=) (is)]** を選択し、その右で MID サーバーの名前を選択します。



重要: 選択した MID サーバーが、シークレットグループに関連付けられているものと同じであることを確認します。

5. ワークフロースタジオでレコードルックアップステップを構成します。

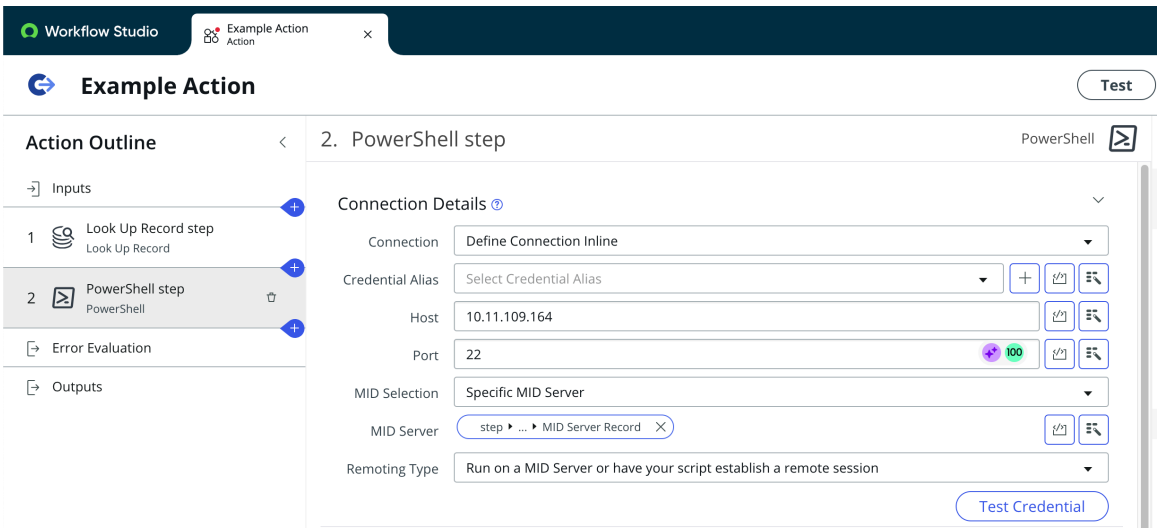
a. [アクションアウトライン] で、プラスボタンを選択して、[レコードをルックアップ] ステップの下に新しいステップを追加します。
[PowerShell] ステップタイプを探して選択します。

b. **[PowerShell** ステップ] フォームで、必要に応じてフィールドに記入します。

PowerShell ステップのフィールド

フィールド	値
接続	[インラインで接続を定義] を選択します。
認証情報エイリアス	認証情報エイリアスを選択します。
ホスト	ターゲット Windows サーバーの IP アドレスを入力します。
ポート	まだデフォルトになっていない場合は、 22 を入力します。
MID 選択	[特定の **MID** サーバー] を選択します。

フィールド	値
MID サーバー	[データ] パネルから **[MID サーバーレコード (MID Server Record)]** ピルをドラッグします。 このデータピルは、画面右端の [データ] パネルの [「レコードのルックアップ」ステップ] の下に表示されます。
リモート処理タイプ	**[MID サーバーで実行するか、スクリプトでリモートセッションを確立します]** を選択します。



6. テストアクションのスクリプトを構成します。

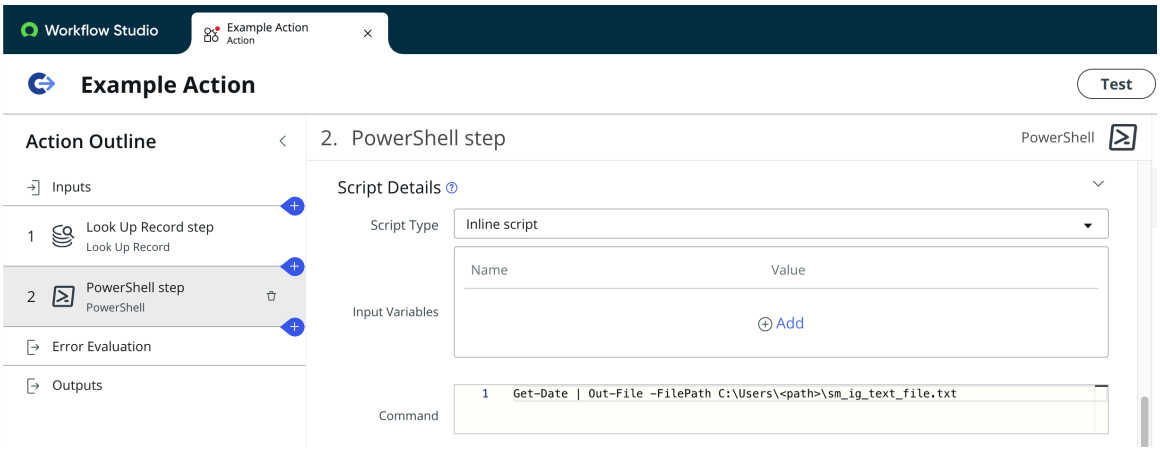
a. [スクリプトタイプ] フィールドで、[インラインスクリプト] を選択します。

b. 次のスクリプトを使用して、パスの例をテストホストのパスに置き換えます。

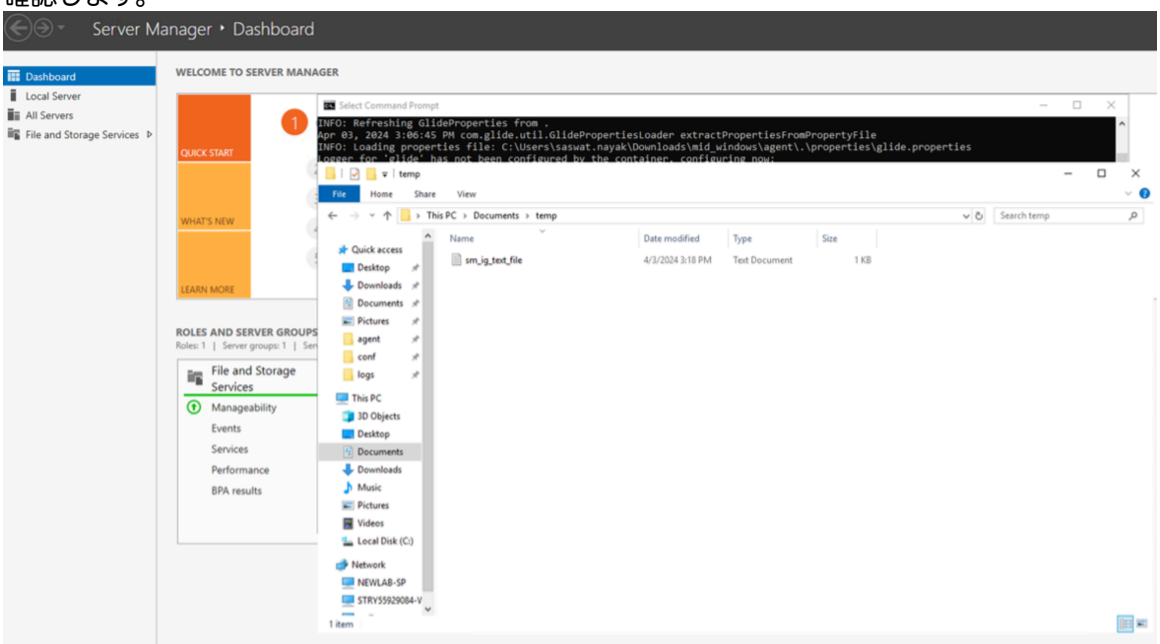
```
Get-Date | Out-File -FilePath C:\Users\<path>\sm_ig_text_file.txt
```

注: また、リモートホストの構成に適した独自のスクリプトを実行することもできます。SM で暗号化された値を使用して接続が確立されたことを確認することが重要です。

c. [保存] ボタンを選択してアクションを保存します。



7. [テスト] ボタンを選択してアクションをテストします。
8. 出力ログでエラーメッセージを確認します。
9. スクリプトで指定したファイルパスのホストサーバー上にテストファイルが作成されていることを確認します。



クローン作成とシークレット管理

クローン後にシークレットグループとクライアントシークレットグループを再構成する方法について説明します。

インスタンスのクローンを作成した後、シークレットグループとクライアントシークレットグループが想定どおりに動作するように再構成する必要がある場合があります。

シークレットグループの作成元	クローン後の動作
ターゲットインスタンスの元のインスタンスシークレットグループ	足りない暗号化モジュールを手動でインポートすれば、想定どおりに動作します。「XML でのデータのエクスポートとインポート」を参照してください。
ターゲットインスタンスの元のクライアントシークレットグループ	足りない暗号化モジュールを手動でインポートすれば、想定どおりに動作します。「XML での

シークレットグループの作成元	クローン後の動作
	データのエクスポートとインポート」を参照してください。
ソースインスタンスからクローンされたインスタンスシークレットグループ	手動でセットアップしない限り機能しません。
ソースインスタンスからクローンされたクライアントシークレットグループ	sg/ID グループ/エイリアス/mid を手動で構成すれば機能します。

シークレット管理ダッシュボード

シークレット管理ダッシュボードを使用して、インスタンスに設定されているシークレットグループを確認し、セキュリティの問題について確認します。

シークレットグループの概要

[シークレットグループの概要] タブには、設定済みのシークレットグループに関する情報が表示されます。このタブを使用して、インスタンスに設定されたシークレットグループに関する情報を表示します。

タイプ別のシークレットグループ

インスタンスにインストールされているシークレットグループをシークレットタイプ (クライアント側のインスタンス側) 別に示す円グラフを表示します。

基準タイプ別のシークレットグループ

インスタンスに設定されている基準を持つシークレットグループを、使用されている基準のタイプ別に示す円グラフを表示します。

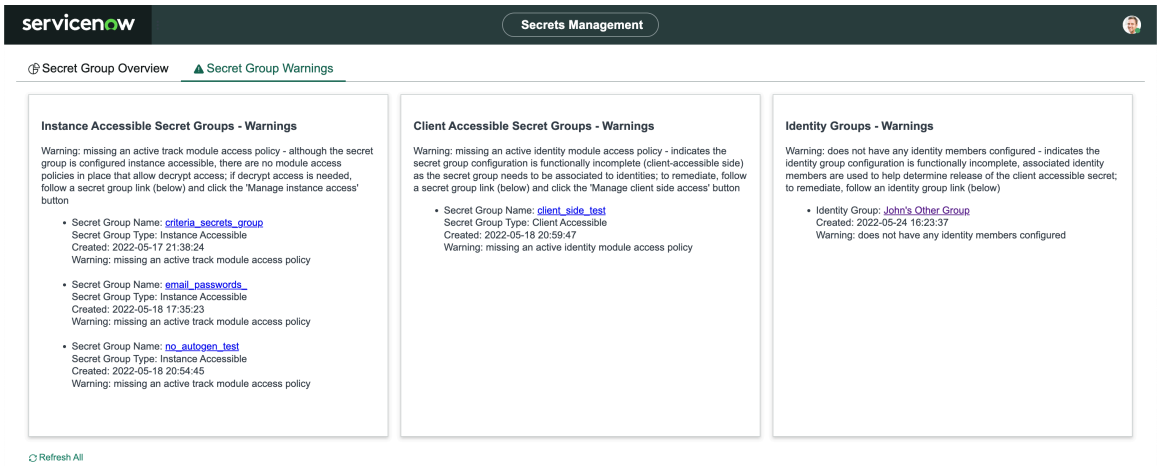
非アクティブなシークレットグループの数

インスタンスに設定されている非アクティブなシークレットグループの数を表示します。

専用シークレットの数

基本シークレットグループ内のシークレットの数を表示します。

シークレットグループの警告



[シークレットグループの警告] タブには、シークレットグループと ID グループに関連する警告が表示されます。

インスタンスアクセス可能なシークレットグループ：警告

アクティブなアクセスポリシーが設定されていないシークレットグループがある場合、このカードは警告を表示します。シークレットグループ名を選択すると、そのレコードが表示されます。

クライアントアクセス可能なシークレットグループ - 警告

アクティブな ID モジュールアクセスポリシー (MAP) を持たないクライアントのアクセス可能なシークレットグループがある場合、このカードは警告を表示します。シークレットグループ名を選択すると、そのレコードが表示されます。

ID グループ：警告

グループメンバーが構成されていない ID グループがある場合、このカードは警告を表示します。ID グループ名を選択すると、レコードが表示されます。

注: シークレット管理ダッシュボードは、シークレット管理 Enterprise の一部です。シークレット管理 Enterprise は、ServiceNow 担当者が本番インスタンスで有効にする必要がある有料プラグインです。

シークレット管理ロール

シークレット管理は次のロールを追加します。

シークレットアドミン **[sn_secrets.admin]**

アドミン以外のシークレットロールを他のユーザーに割り当てます。シークレットアドミニストレーターには、シークレットマネージャーおよびビューアーと同じ権限があります。

ロールを含む

ロール内に含まれるロールのリスト。

なし。

グループ

このロールがデフォルトでアサインされているグループのリスト。

なし。

特別な考慮事項

重要: より分化したロールが利用可能な場合は、アドミンロールを付与しないでください。

- ユーザーが sn_secret.admin ロールをアサインされるには、admin ロールと security_admin ロールの両方を持っている必要があります。
- より多くのターゲットロールが利用可能な場合は、admin ロールを付与しないでください。

シークレットマネージャー **[sn_secrets.secret_manager]**

次のいずれかの機能を実行できる必要があるユーザーにこのロールを付与します。

- シークレットおよびシークレットグループレコードを表示する
- アクセス履歴やその他の使用アクティビティ情報を表示する
- シークレットグループとフィルターを作成する
- シークレットプロバイダーを作成する
- シークレットグループ間でのシークレットを移動する
- シークレットグループとシークレットプロバイダーの設定を変更する

ロールを含む

ロール内に含まれるロールのリスト。

なし。

グループ

このロールがデフォルトでアサインされているグループのリスト。

なし。

特別な考慮事項

重要: シークレットマネージャーは、シークレットをプレーンテキストで表示することはできません。

シークレットビューアー **[sn_secrets.viewer]**

シークレットおよびシークレットグループレコードを表示できる必要があるユーザーにこのロールを付与します。

ロールを含む

ロール内に含まれるロールのリスト。

なし。

グループ

このロールがデフォルトでアサインされているグループのリスト。

なし。

特別な考慮事項

なし。

シークレットグループ暗号化モジュールの作成

暗号化と復号化を実行するためのシークレットグループ暗号化モジュールを作成します。

始める前に

必要なロール：

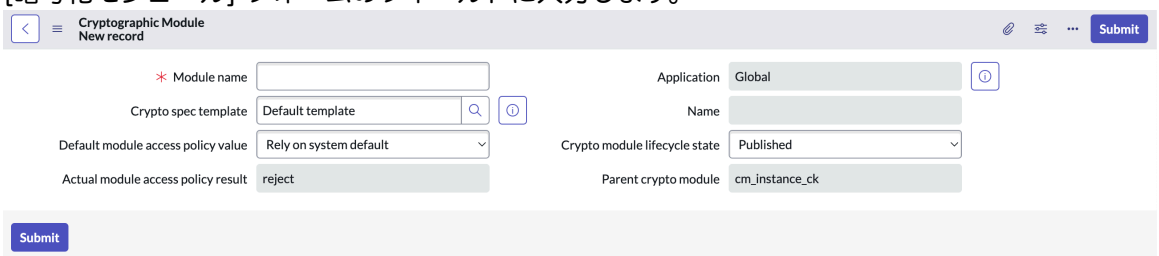
- admin
- sn_secrets.admin
- sn_secrets.secret_manager
- sn_kmf.cryptographic_manager
- sn_kmf.admin

手順

1. 移動先 すべて > シークレット管理 > シークレットグループ暗号化モジュールを作成.
2. 作成するグループ暗号化モジュールのタイプを選択します。

暗号化モジュールタイプ	説明
インスタンスのアクセス可能なシークレット暗号化モジュールを作成	インスタンスで復号化できるインスタンスのアクセス可能なシークレットを作成します。
クライアントのアクセス可能なシークレット暗号化モジュールを作成	ServiceNow がアクセスできない独自のキーを使用して暗号化された、クライアントのアクセス可能なシークレットを作成します。

3. [暗号化モジュール] フォームのフィールドに入力します。



フィールド	説明
モジュール名	モジュールのわかりやすい名前
アプリケーション	使用中のモジュールを含むアプリケーションスコープこのフィールドには現在のモジュールが自動的に入力されます。
暗号化仕様テンプレート	デフォルトでは、[デフォルトテンプレート] が選択されます。
名前	モジュールの名前。この名前はアプリケーションとモジュール名に基づいて自動的に生成されます。

フィールド	説明
デフォルトのモジュールアクセスポリシー値	○ システムのデフォルトに依存 ○ 却下 ○ 追跡
暗号化モジュールライフサイクル状況	モジュールのステータスが [ドラフト] であるか [公開済み] であるか。
実際のモジュールアクセスポリシーの結果	このフィールドは情報提供を目的としており、読み取り専用です。
親暗号化モジュール	前の手順で選択した暗号化モジュールタイプによって決まる親暗号化モジュール。このフィールドは読み取り専用です。

4. [送信] を選択します。

基本シークレットグループの作成

基準に関係なく、任意のシークレットをグループ化する基本シークレットグループを作成します。

始める前に

必要なロール：admin

基本シークレットグループには、テーブル、スコープ、またはアプリケーションに関係なく、追加したシークレットを含めることができます。グループを作成した後、グループにシークレットを手動で追加します。それらの共通の属性を共有するすべてのシークレット専用のグループを作成するには、基準を持つシークレットグループを「基準のあるシークレットグループの作成」の手順を使用して作成します。

手順

1. 移動先 すべて > シークレット管理 > シークレットグループ.
2. **[New** (新規)] を選択します。
3. [どのタイプのシークレットグループを作成しますか？] で [基本シークレットグループ] を選択します。
4. [シークレットグループ] フォームのフィールドに記入します。

シークレットグループのフィールド

フィールド	説明
グループ名	グループの名前 注: シークレットグループ名に使用できるのは、小文字、数字、アンダースコア (_) のみです。
シークレットタイプ	グループが [インスタンスアクセス可能 **(Instance accessible)]** または [クライアントアクセス可能 **(Client accessible)]** かどうか。
自動生成モジュール	このシークレットグループの新しい暗号化モジュールを生成します。このモジュールは、

フィールド	説明
	データを暗号化および復号化します。このフィールドはデフォルトで有効になっています。
アプリケーション	このレコードのスコープ対象のアプリケーション。この読み取り専用のフィールドには現在のスコープが自動的に入力されます。
簡単な説明	グループの説明
暗号化モジュール	このグループで使用する暗号化モジュールを選択します。このモジュールは、データを暗号化および復号化します。このフィールドが表示されるのは、[自動生成モジュール] が選択されていない場合のみです。モジュールアクセスポリシーの詳細については、「モジュールアクセスポリシーの概要」を参照してください。 注: [インスタンスのアクセスを管理] ボタンを使用して、シークレットグループに関連付するモジュールアクセスポリシーを確認できます。

5. [送信] をクリックします。

   注: 作成したときには、シークレットグループはデフォルトで非アクティブになります。グループをアクティブにするには、グループレコードに戻り、[アクティブ] を選択します。

基準のあるシークレットグループの作成

テーブル、スコープ、アプリケーションなど、共通の基準を共有するときに Password2 フィールドに入力されたシークレットを自動的に整理するための基準を使用して、シークレットグループを作成します。

始める前に

必要なロール：admin、KMF_admin、sn_secrets.secret_manager、および sn_kmf.cryptographic_manager

このタイプのシークレットグループ内のシークレットは、すべて共通の基準を共有している必要があります。この制限のないグループの場合は、基本シークレットグループを作成することを検討してください。基本シークレットグループの作成については、「基本シークレットグループの作成」を参照してください。

手順

1. 移動先 すべて > シークレット管理 > シークレットグループ.
2. **[New** (新規)] を選択します。
3. [作成するシークレットグループのタイプ] プロンプトで [基準のあるシークレットグループ] を選択します。
4. [シークレットグループ] フォームのフィールドに記入します。

シークレットグループのフィールド

フィールド	説明
グループ名	グループの名前 **注:** シークレットグループ名に使用できるのは、小文字、数字、アンダースコア (_) のみです。
シークレットタイプ	グループが [インスタンスアクセス可能 **(Instance accessible)]** または [クライアントアクセス可能 **(Client accessible)]** かどうか。
自動生成モジュール	このシークレットグループの新しい暗号化モジュールを生成します。このモジュールは、データを暗号化および復号化します。このフィールドはデフォルトで有効になっています。
アプリケーション	このレコードのスコープ対象のアプリケーション。この読み取り専用のフィールドには現在のスコープが自動的に入力されます。
簡単な説明	グループの説明
基準タイプ	このグループのシークレットが共有する基準 ○ スコープ ○ パッケージ ○ ターゲットテーブル ○ シークレット列 ○ レコードのフィルタリング
暗号化モジュール	このグループで使用する暗号化モジュールを選択します。このモジュールは、データを暗号化および復号化します。このフィールドが表示されるのは、[自動生成モジュール] が選択されていない場合のみです。モジュールアクセスポリシーの詳細については、「モジュールアクセスポリシーの概要」を参照してください。 **注:** [インスタンスのアクセスを管理] ボタンを使用して、シークレットグループに関連付するモジュールアクセスポリシーを確認できます。

注: 構成によっては、[暗号化モジュール] によって使用される値が自動的に選択される場合があります。

[基準タイプ] フィールドが [パッケージ] に設定され、[自動生成モジュール] フィールドが選択されている場合：	[暗号化モジュール] フィールドは空で読み取り専用になります。既存の Password2 サブモジュールが使用されます。Password2 サブモジュールが見つからない場合は、インスタンスレベルの Glide エンクリプターモジュールが使用されます。
[基準タイプ] フィールドが [パッケージ] に設定され、[自動生成モジュール] フィールドの選択が解除されている場合：([自動生成モジュール] フィールドの選択を解除できるのはエンタープライズユーザーのみです)	[暗号化モジュール] フィールドが編集可能になり、アドミンは使用する暗号化モジュールを選択できます。

5. フォームヘッダーを選択して長押し (または右クリック) し、[保存] を選択します。

注: 作成したときには、シークレットグループはデフォルトで非アクティブになります。

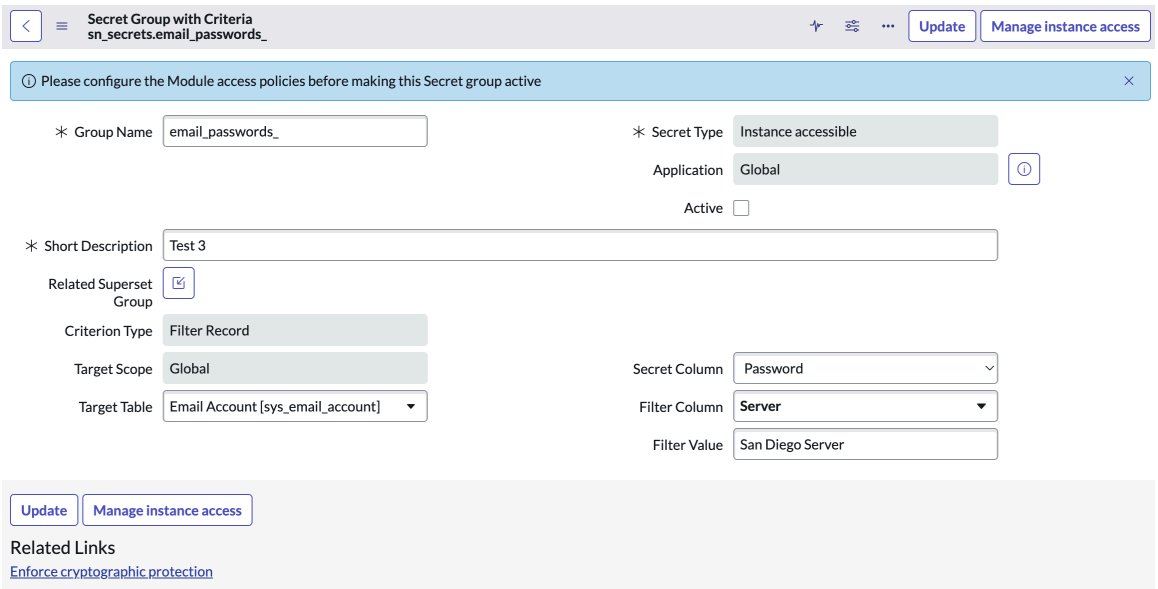
6. レコードを保存した後に、グループの設定に基づいて追加のフィールドが表示される場合があります。

追加のシークレットグループフィールド

フィールド	説明
ターゲットスコープ	このグループにアサインされるシークレットによって共有されるスコープ。このフィールドは、[基準タイプ] フィールドで [スコープ] を選択した場合にのみ使用できます。
ターゲットパッケージ	このグループにアサインされるシークレットによって共有されるパッケージ。このフィールドは、[基準タイプ] フィールドで [パッケージ] を選択した場合にのみ使用できます。
ターゲットテーブル	このグループにアサインされるシークレットによって共有されるテーブル。このフィールドは、[基準タイプ] フィールドで [テーブル] または [シークレット列] を選択した場合にのみ使用できます。
ターゲットスコープ	[ターゲットテーブル] フィールドで選択されたテーブルのアプリケーションスコープ。このフィールドは、[基準タイプ] フィールドで [テーブル]、[フィルター列]、または [シークレット列] を選択した場合にのみ表示されます。
シークレット列	このグループに含める pasword2 シークレットを含むテーブル列。このリストで使用できるフィールドは、[ターゲットテーブル] フィールドで選択したテーブルによって決まります。

フィールド	説明
	注: 選択したテーブルにシークレットを含む列がない場合、このフィールドには選択肢として [-- なし --] のみが表示されます。
列をフィルター	フィルターとして使用する [ターゲットテーブル] で選択されたテーブルの列。このフィールドを Password2 フィールドにすることはできません。
フィルター値	フィルターとして使用する値。このフィルターは、[フィルター列] フィールドのフィールド選択に適用されます。

Example: メールサーバーのすべてのメールアカウントのパスワードを含むインスタンスにアクセス可能なグループ



次のタスク

グループを作成すると、条件に一致するすべての新しいレコードが暗号化されます。このグループの暗号化モジュールを使用して既存のレコードを暗号化するには、セキュリティジョブを実行する必要があります。詳細については、「シークレット管理セキュリティジョブの実行」を参照してください。

クライアントアクセス可能グループには、シークレットを暗号化するための顧客提供の公開鍵が必要です。このキーをアップロードする手順については、「シークレット管理の公開鍵をアップロードする」を参照してください。

シークレット管理の公開鍵をアップロードする

シークレットを暗号化するための公開鍵をアップロードします。

始める前に

必要なロール：admin

手順

1. 移動先 すべて > シークレット管理 > シークレットグループをクリックし、シークレットグループレコードを開きます。
2. レコード内で、[クライアント側のアクセスを管理] ボタンを選択します。
   新しい ID グループレコードが開きます。
3. **[ID** キーをアップロード **(Upload Identity Key)]** ボタンを選択します。
   [ID 公開鍵証明書をインポート] ウィンドウが表示されます。
4. **[ID** キーエイリアス] フィールドにキーのエイリアスを入力します。
5. [インポート] ボタンを選択して、ローカル環境からキーをアップロードします。
6. **[OK]** ボタンをクリックします。
   [グループモジュールキー **(Group module key)]** フィールドでは、ID キーのエイリアス名が使用されます。
7. [送信] をクリックして ID グループレコードを保存します。

シークレット管理セキュリティジョブの実行

インスタンスのシークレットフィールドで暗号化タスクを実行するシークレット管理ジョブをスケジュールします。

始める前に
必要なロール：sn_kmf.admin、security_admin、sn_secrets.admin

次の手順を実行するには、security_admin ロールに昇格する必要があります。このプロセスの詳細については、「特権ロールへの昇格」を参照してください

手順

1. 移動先 すべて > システムセキュリティ > セキュリティジョブ > 新規作成.
2. [作成するセキュリティジョブのタイプは？] プロンプトで、[シークレット管理ジョブ] を選択します。
3. フォームのフィールドに入力します。

[シークレット管理ジョブ] フォーム

フィールド	説明
名前	セキュリティジョブの名前
状態	初期ジョブのステータスは新規です。ジョブがスケジュールどおりに実行された後、それに応じてステータスが更新されます。
期間開始	24 時間形式のジョブの開始時間。選択した時刻にジョブの実行が開始されます。
期間終了	24 時間形式のジョブの終了時間。この時点でジョブが終了していない場合は、ジョブが完了するまで、次の指定された処理ウィンドウでジョブが続行されます。
適用レベル	ジョブがすべてのテーブルに影響するか、特定のテーブルまたはフィールドに影響を与えるか。以下から選択します。

フィールド	説明
	○ All Tables ○ 特定のテーブル ○ 特定のフィールド ○ 特定のパッケージ ⚠ 警告: [すべてのテーブル] オプションを選択すると、インスタンスのパフォーマンスに影響を与える可能性があります。ピーク時間外にスケジュールすることを検討してください。
パッケージ	このジョブに含めるパッケージ。選択したパッケージに暗号化が適用されます。このオプションは、[適用レベル] フィールドが [特定のパッケージ **(Specific Packages)]** に設定されている場合にのみ表示されます
テーブル	このジョブに含めるテーブル。選択したテーブル内のすべての適用可能なフィールドに暗号化が適用されます。このオプションは、[適用レベル] が [特定のテーブル] に設定されている場合にのみ表示されます
フィールド	このジョブに含めるテーブル。選択したすべてのフィールドに暗号化が適用されます。このオプションは、[適用レベル] が [特定のフィールド] に設定されている場合にのみ表示されます
ジョブモード	以下から選択します。 Password2 からシークレット管理 各グループのモジュールアクセスポリシーが定義されている暗号化モジュールを使用して、シークレットグループ内のすべての Password2 フィールドを暗号化します。 シークレット管理から Password2 Password2 暗号化を使用して、シークレットグループのデータを再暗号化します。この暗号化タイプの詳細については、「KMF による Password2 暗号化」を参照してください。 シークレットグループの適用 [シークレットグループ] フィールドで選択したグループに一致する必要があるすべてのデータをクエリーします。クエリーによって見つかったすべてのデータが既にグループ内にある場合、ジョブは変更を行いません。クエリーでまだ

フィールド	説明
	グループに含まれていないデータが見つかった場合、ジョブはこのデータをシークレットグループ内で再暗号化します。 **注:** このクエリーで見つかったデータが既に暗号化されていて、インスタンスでそのデータを復号化できない場合は、データは暗号化されずにシークレットグループに追加されます。
シークレットグループ	暗号化するシークレットを含むシークレットグループ。このフィールドは、[ジョブモード] フィールドで [シークレットグループ] が選択されている場合にのみ使用できます。
データのリキーを強制	
サマリー	ジョブの進捗状況に関する情報を表示します。[サマリー] には、ジョブで暗号化できなかったレコードも表示されます。

4. [送信] を選択します。

次のタスク
このジョブは、選択したシークレットグループに一致する必要があるすべてのデータをクエリーします。クエリーによって見つかったすべてのデータが既にグループ内にある場合、ジョブは変更を行いません。クエリーでまだグループに含まれていないデータが見つかった場合、ジョブはこのデータをシークレットグループ内で再暗号化します。(インスタンスで復号化できる場合でも、クライアント側で暗号化されたシークレットは複合化できない可能性があります)。

コード署名

コード署名は、データの信頼性と整合性を確認するために後でチェックされるデータのデジタル署名を作成します。コード署名は、ServiceNow Vault のコンポーネントとしてライセンスされるモジュールです。

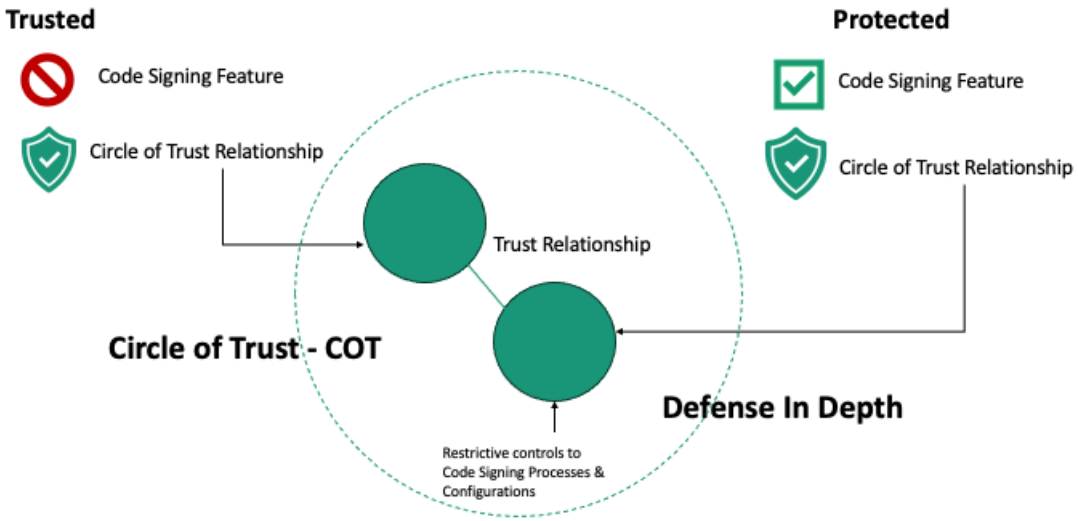
コード署名および信頼のサークル

信頼の輪 (Circle of Trust) (COT) は、信頼できるインスタンスと保護されたインスタンスの間に安全な通信を作成し、許可されたユーザーのみがコード署名機能にアクセスできるようにするためのコード署名の前提条件です。

複数のセキュリティ対策により、保護されたインスタンスが侵害された場合に、悪意のある攻撃者がコード署名を無効にしたり悪用したりするのを防ぐことができます。多層防御戦略の一環として、COT は次のコンポーネントを使用します。

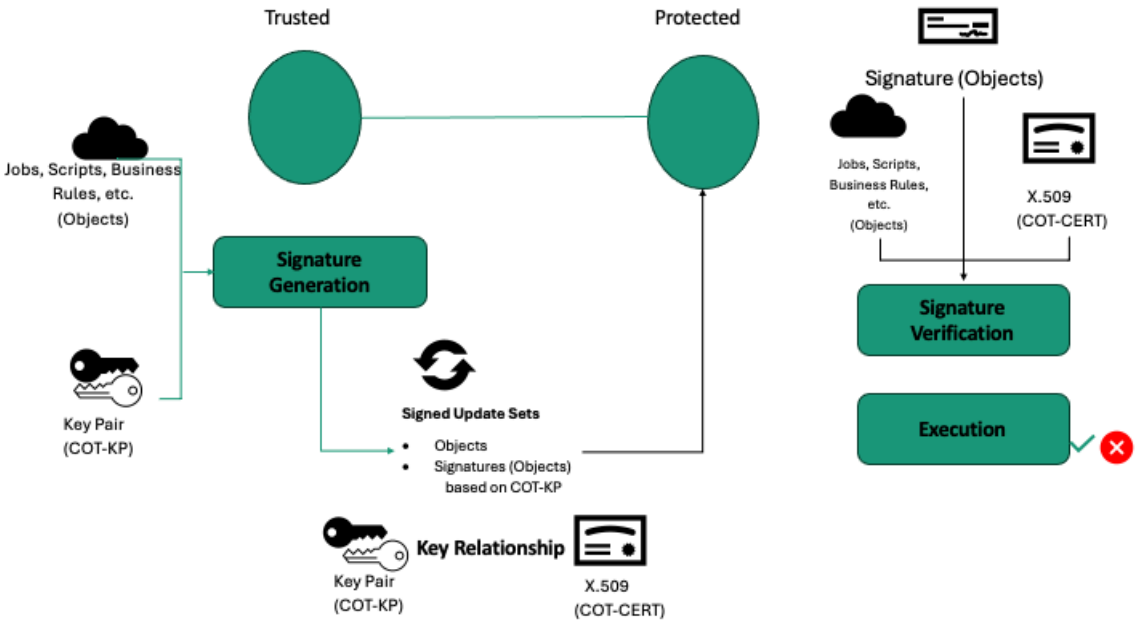
- 最も強力なアドミニストレーターアカウントも制限するコントロールが保護されたインスタンスに確立され、コード署名のプロセスと構成を保護できます。
- 信頼できるインスタンスは、保護されたインスタンスと連携して 信頼の輪 (Circle of Trust) 関係を確立する必要があります。少なくとも 1 つの信頼できるインスタンスが必要ですが、複数の信頼できるインスタンスが保護されたインスタンスと連携するように構成できます。

信頼のサークルの概要

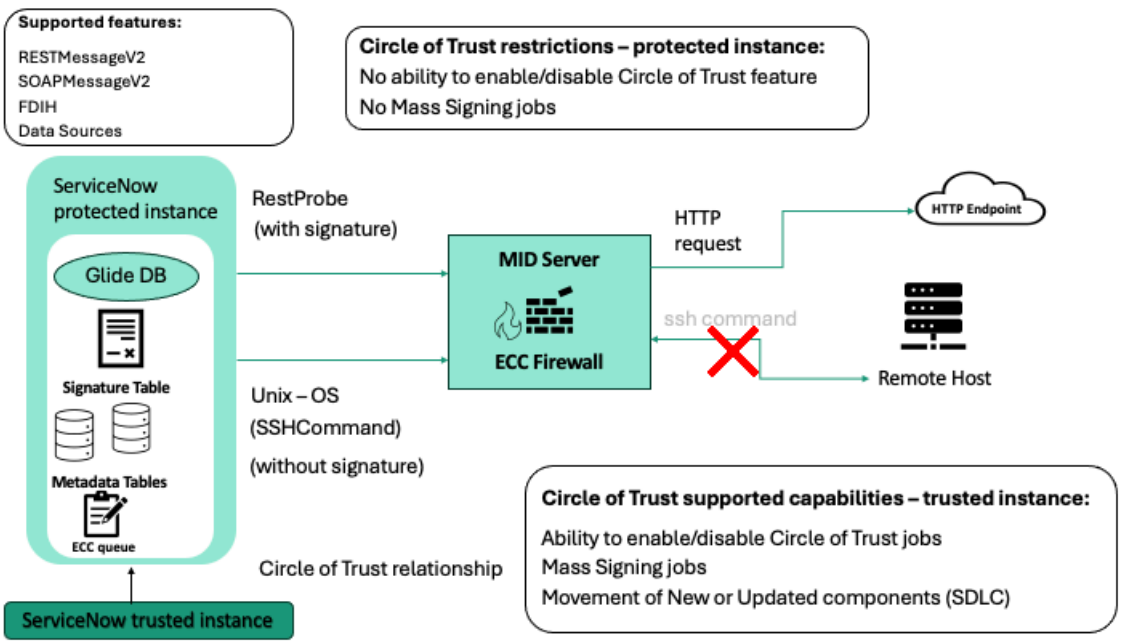


信頼の輪 (Circle of Trust)は、ジョブ、スクリプト、ビジネスルールをキーペアとともに使用して、保護されたインスタンスの更新セットに署名するための署名を生成します。ジョブが呼び出されると、保護されたインスタンスの更新を実行するために、信頼できる証明書とともに署名が検証されます。

信頼できる更新セットのプロセス

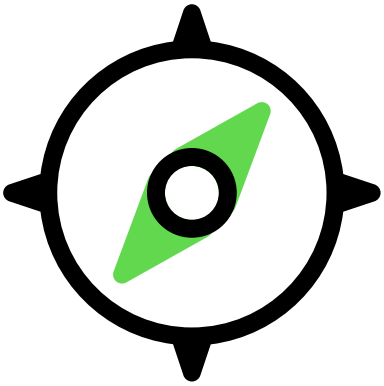
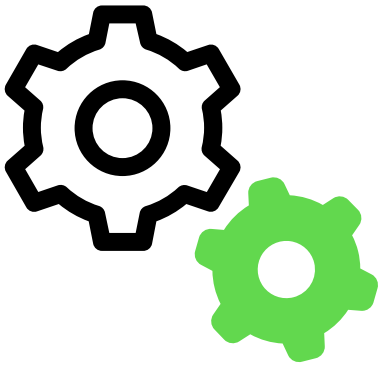
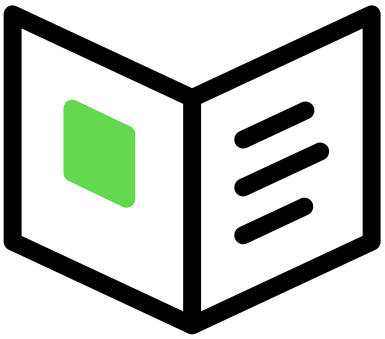


コード署名フロー



信頼の輪 (Circle of Trust)には、信頼できるインスタンスと保護されたインスタンスの間に初期信頼関係が必要です。これにより、任意の承認レベルの承認されていないユーザーが未承認のアクティビティにアクセスできなくなります。

開始

探索	構成	参照
		
コード署名の主な機能とビジネス価値について学びます。	コード署名をアクティブ化して構成します。	プロパティとトラブルシューティングの詳細を取得します

使用方法

コード署名を使用してデータの信頼性と整合性を検証する方法について説明します。

トラブルシューティングとサポート

- https://www.servicenow.com/community/secops/ct-p/security-operations
- Known Error Portal で既知のエラー記事を検索
- 連絡先 カスタマーサービス & サポート

コード署名の概要

コード署名は、許可されたスクリプトのみが MID サーバー上で実行できるようにするための暗号化検証を提供します。コード署名は、承認されていない、または改ざんされた ECC キューレコードが MID サーバーによって処理されるのを防ぎ、ServiceNow と外部システム間の統合の整合性を維持します。

コード署名は、データの信頼性と整合性を確認するために後でチェックされるデータのデジタル署名を作成します。コード署名は、ServiceNow Vault のコンポーネントとしてライセンスされるモジュールです。

注: カスタマーサービス & サポートチームはコード署名へのアクセスを許可する必要があります。

コード署名は、実行されている操作の背後にある意図を宣言し、リソースまたはレコードを、意図した目的で使用できるかどうかを検証します。コード署名を容易にするために、キー管理フレームワーク (KMF) は、デジタル署名のためにデジタル証明書と業界標準の非対称暗号化を使用します。

プラットフォームとインフラストラクチャ側で内部的にコード署名を使用します。コード署名は、特定のテーブルのコンテンツや、特定のメタデータテーブルにあるレコードのサブセットのコンテンツに署名する方法を提供します。

コード署名は、信頼できるインスタンスと保護されたインスタンスの間で安全な 信頼の輪 (Circle of Trust) (COT) を使用して、承認された安全な信頼できるインスタンスのみがコード署名機能にアクセスできるようにします。

注: コード署名は、信頼できるインスタンスではなく、保護されたインスタンスで有効になります。

コード署名が環境を保護する仕組み

コード署名がない場合、 ServiceNow レコードへのアクセス権を取得した攻撃者は、保護されたインスタンスの SQL ステートメントを変更できます。MID サーバーがこのデータソース要求を処理すると、悪意のある SQL コマンドが実行され、システムの整合性とセキュリティが侵害される可能性があります。

コード署名を使用して信頼のサークルアーキテクチャを実装する場合、MID サーバーへのデータ転送は次の検証プロセスに従います。このプロセスにより、信頼できるインスタンスに由来する許可されたコードのみが MID サーバーで実行できるようになります。このプロセスにより、システムを危険にさらす可能性のある潜在的な攻撃ベクトルが削減されます。

1. デジタル署名は、信頼できるインスタンス内で作成または更新されたデータソースに適用されます。
2. コード署名プロセスを使用して、署名されたデータを信頼できるインスタンスから保護されたインスタンスに転送します

3. MID サーバーは、すべての受信要求のデジタル署名を検証し、有効な署名のない要求は自動的に拒否します。
4. MID サーバーが要求を却下した場合、この却下をログに記録し、保護されたインスタンスに通知を送信します。

コード署名を実装するメリット

コード署名には、いくつかの重要な利点があります。

実行コントロール
: 暗号化検証済みスクリプトのみを MID サーバーで実行できます

改ざん検出
: 署名されたレコードへの変更は即座に識別され、ブロックされます。

自動保護
: システムは、手動操作を必要とせずにセキュリティ適用を処理します。

包括的なログ記録
: 署名検証が失敗すると、詳細な監査レコードが生成されます。

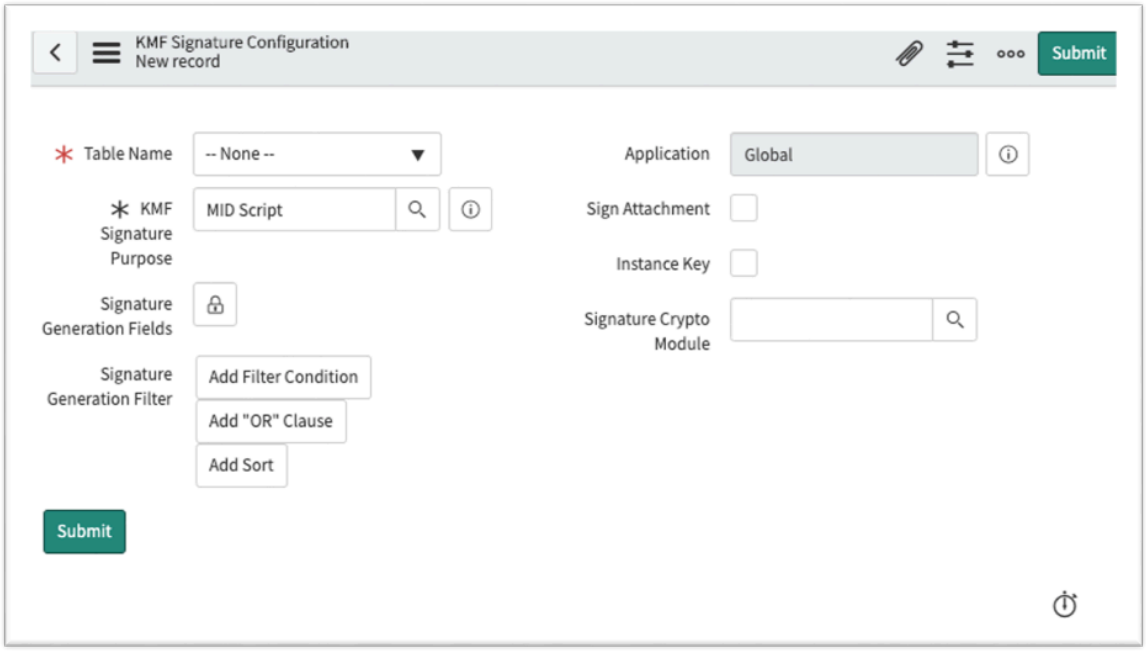
コード署名の検証とジョブ

有効な構成を持つすべてのメタデータテーブルは、コード署名メタデータプラグイン (`com.glide.code_signing`) を使用してビルド時に署名されます。テーブルに署名することを選択した場合、セキュリティアドミニストレーターロールを持つ admin ユーザーは、コード署名暗号化ジョブにアクセスできます。

- 更新セットに署名する。
- レコードに一括署名する。
- 添付ファイルに一括署名する。

更新セットに署名する。
: このジョブでは、更新セット内の署名構成に一致するレコードに署名します。このジョブでは、すべての新しい署名レコードと検証証明書も更新セットに追加します。

更新セットの **KMF** 署名レコード



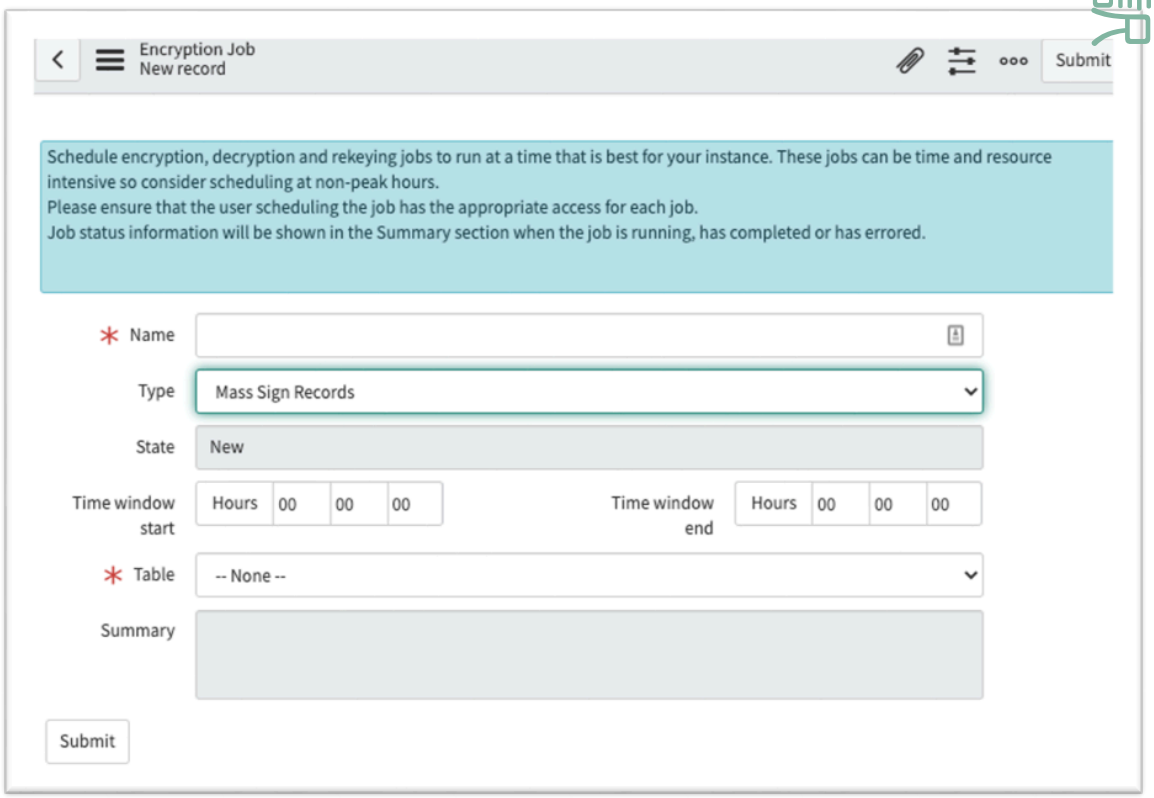
レコードに一括署名する

このジョブでは、特定のメタデータテーブルに適用される署名構成に一致するすべてのレコードに署名します。

添付ファイルに一括署名する

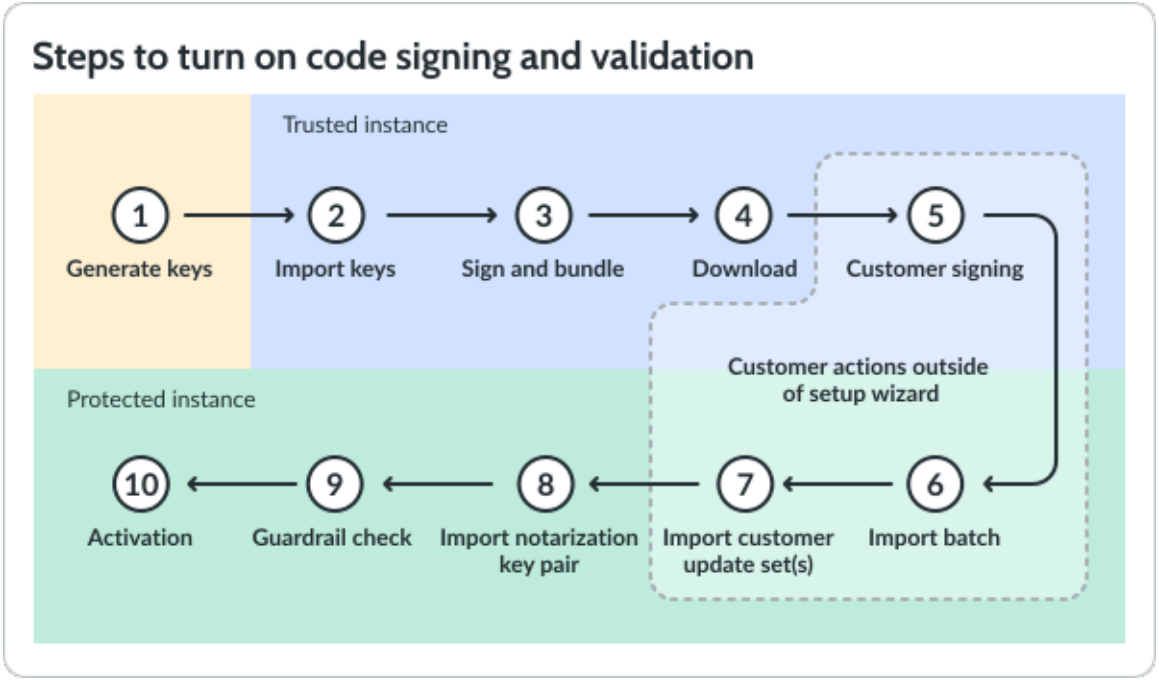
このジョブでは、指定された署名構成に一致するテーブルに添付されているすべての添付ファイルレコードに署名します。

レコードに一括署名する暗号化ジョブ



コード署名の構成

データの信頼性と完全性を検証するため、コード署名をアクティブ化して構成します。



コード署名エンタープライズには、信頼できるインスタンスと保護されたインスタンスの間に初期信頼関係が必要です。これにより、任意の承認レベルの承認されていないユーザーが未承認のアクティビティにアクセスできなくなります。

コード署名エンタープライズで信頼の輪 (Circle of Trust)を確立するための構成手順を実行するには、各トピックを参照してください。

コード署名アドミニストレーターロールのアサイン
: コード署名アドミニストレーターロールをユーザーにアサインして、コード署名構成エクスペリエンスにアクセスします。

信頼できるインスタンスでのコード署名エンタープライズの構成
: 信頼できるインスタンスでコード署名をオンにします。

コード署名構成ファイルを保護されたインスタンスにアップロードする
: 信頼できるインスタンスで生成された構成ファイルをアップロードします。

保護されたインスタンスでのコード署名エンタープライズの構成
: 保護されたインスタンスでコード署名をオンにして構成します。

証明書の検証をオンにする
: インスタンスで証明書の検証をオンにします。

コード署名を無効にする
: 保護されたインスタンスでコード署名を無効にします。

> **注:** このオプションのステップは、コード署名の初期構成の一部ではありません。

コード署名アドミニストレーターロールのアサイン

コード署名アドミニストレーターロールをユーザーにアサインして、コード署名構成エクスペリエンスにアクセスします。

始める前に
必要なロール：

- admin
- security_admin

重要: コード署名アドミニストレーターロールが割り当てられているユーザーは、グローバルスコープ内である必要があります。

手順

1. 使用する信頼できるインスタンスを選択します。
   信頼できるインスタンスは、保護されたインスタンスで署名検証を有効にする前に、ジョブの作成、レコードへの署名の署名、およびその他の必要なタスクを実行するために使用されます。
2. 移動先 すべて > コード署名 > アドミニストレーション > ロール管理.
3. [利用可能なユーザー **(Available Users)**] リストから [選択したユーザー **(Selected User(s))**] リストにユーザーを移動することで、ユーザーにコード署名アドミニストレーターロールを付与します。
4. [保存] を選択します。

重要: ロールにアサインされたユーザーは、コード署名タスクを実行するために、一旦ログアウトしてからログインし直す必要があります。

結果
選択したユーザーにコード署名アドミンロールが付与されます。このロールを持つユーザーは、コード署名構成エクスペリエンスにアクセスして、他のコード署名ロールをユーザーにアサインすることができます。

コード署名ロールの詳細については、「コード署名とともにインストールされるロール」を参照してください。

信頼できるインスタンスでのコード署名エンタープライズの構成

信頼できるインスタンスでコード署名をオンにして構成します。

始める前に
必要なロール：

- admin
- security_admin
- codesigning_admin

  注: codesigning_admin ロールは、「コード署名アドミニストレーターロールのアサイン」で詳述されているプロセスを使用してアサインできます。

- sn_kmf.cryptographic_manager

顧客署名と信頼のサークル (COT) 管理のために、少なくとも 1 つの暗号化キーペアと証明書 (p12 ファイル拡張子) が必要です。セキュリティを強化するために、顧客署名と COT 管理に別々の暗号化キーペアを使用することを検討してください。

⚠ 警告: このプロセスを完了すると、1 時間以内に保護されたインスタンスにインストールする必要がある構成ファイルがダウンロードされます。このプロセスの後に、保護されたインスタンスに構成ファイルをアップロードする時間があることを確認してください。このプロセスの詳細については、「コード署名構成ファイルを保護されたインスタンスにアップロードする」を参照してください。

手順

1. 信頼できるインスタンスで、次の場所に移動します: すべて > コード署名 > 設定 > ガイド付きセットアップ をクリックして、[コード署名] 設定ページを開きます。
2. [インスタンスタイプ] フィールドで、 信頼できるインスタンスを選択します。
3. [次へ] ボタンをクリックします。
4. [実行するアクションを選択します] の [アクション] フィールドで、[コード署名を有効にする] を選択します。
5. [顧客署名キー ペアと証明書] セクションの [添付ファイル] の横にある [+ファイルの追加] を選択して、顧客署名目的で使用する暗号化キー ペア (p12 ファイル拡張子) をアップロードします。

   ヒント: [+ファイルを追加] オプションが使用できない場合は、グローバルスコープであり、sn_kmf.cryptographic_manager ロールを持っていることを確認します。

6. [パスワード] フィールドに、アップロードしたキーペアのパスワードを入力します。
7. [インポート] を選択します。
8. [続行] を選択して次のセクションに進みます。
9. [COT 管理キーペアと証明書] セクションの [添付ファイル] の横にある [+ファイルを追加] を選択して、顧客署名目的で使用する暗号化キーペア (p12 ファイル拡張子) をアップロードします。
10. [パスワード] フィールドに、アップロードしたキーペアのパスワードを入力します。
11. [インポート] を選択します。
12. [続行] を選択して次のセクションに進みます。
13. [構成ファイルのエクスポート] ページで [エクスポート] を選択し、保護されたインスタンスでコード署名を有効にするために使用する構成ファイルを作成してダウンロードします。
    エクスポートプロセスでは、「保護されたインスタンスでのコード署名エンタープライズの構成」で詳述されている手順で使用するために、XML ファイルがローカルマシンにダウンロードされます。

    注: コード署名は、ユーザーエクスペリエンスを向上させるために大きな更新セットに制限を適用します。更新セットの最大サイズは 10,000 レコードです。

コード署名構成ファイルを保護されたインスタンスにアップロードする

信頼できるインスタンスで生成された構成ファイルをアップロードします。

始める前に
必要なロール：

- admin
- security_admin
- codesigning_admin

> 注: codesigning_admin ロールは、「コード署名アドミニストレーターロールのアサイン」で詳述されているプロセスを使用してアサインできます。

- sn_kmf.cryptographic_manager

また、「信頼できるインスタンスでのコード署名エンタープライズの構成」で詳述されている手順で生成された構成ファイルも必要です。

手順

1. 移動先 すべて > システムアップデートセット > 取得済み更新セット.
2. [取得済み更新セット] リストの下で、**[XML** から更新セットをインポート] を選択します。
3. [ファイルを選択] を選択し、構成ファイル (xml ファイル拡張子) を選択します。
4. [アップロード] を選択します。
   [取得済み更新セット] リストに [コード署名構成] 更新セットが表示されるはずです。
5. [コード署名構成] 更新セットを選択して開きます。
6. [更新セットバッチをプレビュー] を選択します。
   プレビュー中に競合が発生した場合は、[リモートの更新を承認] または [既知の **CS** 競合を解決 **(Resolve known CS conflicts)]** を選択して解決します。
7. 競合を解決したら、[更新セットバッチをコミット] を選択します。

保護されたインスタンスでのコード署名エンタープライズの構成

保護されたインスタンスでコード署名をオンにして構成します。

始める前に
必要なロール：

- admin
- security_admin
- codesigning_admin

> 注: codesigning_admin ロールは、「コード署名アドミニストレーターロールのアサイン」で詳述されているプロセスを使用してアサインできます。

- sn_kmf.cryptographic_manager

手順

1. PPI で、 すべて > コード署名 > 設定 > ガイド付きセットアップ をクリックして、[コード署名] 設定ページを開きます。
2. [インスタンスタイプ] フィールドで、[保護されたインスタンス] を選択します。
3. [次へ] ボタンをクリックします。
4. [コード署名構成の更新セットがインポートされ、コミットされたことを確認してください] チェックボックスを選択します。
5. [次へ] ボタンをクリックします。
6. 証明書のインストールが完了するのを待ちます。
   構成ファイルのアイテムがインストールされていることを示す青色のアラートメッセージが表示されます。インストールが完了すると、アラートは数秒で消えます。

7. [ランタイム/公証キーペアと証明書] セクションの [添付ファイル] の横にある [+ファイルを追加] を選択し、ランタイム/公証目的で使用する暗号化キーペア (p12 ファイル拡張子) をアップロードします。
   信頼できるインスタンスのキーペアを再利用することも、今回のために新しいキーペアをアップロードすることもできます。

   ヒント: [+ファイルを追加] オプションが使用できない場合は、グローバルスコープであり、sn_kmf.cryptographic_manager ロールを持っていることを確認します。

8. [パスワード] フィールドに、アップロードしたキーペアのパスワードを入力します。
9. [インポート] を選択します。
10. [続行] を選択して次のセクションに進みます。
11. インスタンスがガードレールチェックを実行する間待ちます。
    このチェックでは、インスタンスに無効な署名がないかスキャンします。完了するまでに時間がかかる場合があります。スキャンが完了するまで、ページを終了したり更新したりしないでください。

    このワークフローは、期限切れまたは非アクティブな証明書で生成された署名を識別し、影響を受けるレコードに再署名します。パフォーマンスを向上させるために、ワークフローはプロセスを複数のイベントに分割し、それらを並行して実行するようになりました。

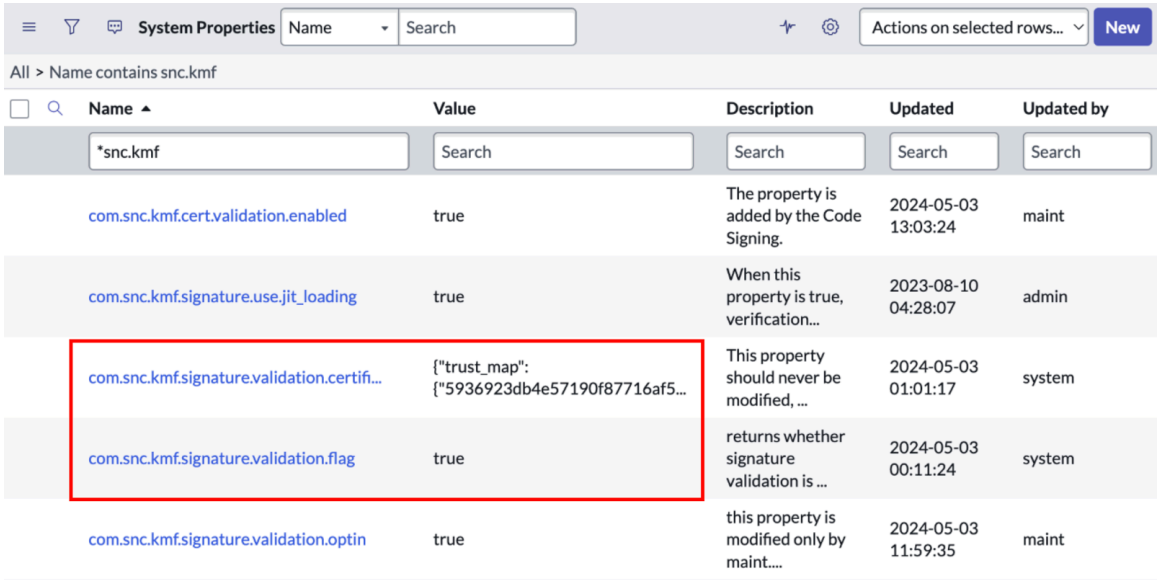
12. 無効な署名が見つかった場合は、[レポートをダウンロード] を選択します。

    [レポートをダウンロード] を選択すると、無効な署名の調査と修正に使用できるレポート (scan_report.txt) がダウンロードされます。

    エラーを解決してから、ページを更新してチェックを再実行します。

13. エラーがない場合は、[セットアップを完了] を選択します。
14. 構成ジョブが完了するまで待ちます。
    インスタンスは 1 つ以上のジョブを実行して、構成プロセスを完了します。ジョブが終了するまで、ページを終了したり更新したりしないでください。完了すると、「コード署名の構成が正常に完了しました」というメッセージが表示されます。

結果
本番インスタンスで署名の検証が実施されます。システムプロパティを確認することで、正常に完了したことを確認できます。システムプロパティ [sys_property] テーブルで、**com.snc.kmf.signature.validation.flag** プロパティを探し、値が **true** であることを確認します。**com.snc.kmf.signature.validation.certificate_trust** プロパティに空の値が含まれていないことを確認します。



証明書の検証をオンにする

証明書ベースの検証でインスタンスを保護します。

始める前に

必要なロール：

- codesigning_admin
- security_admin
- sn_kmf.cryptographic_manager

手順

1. 信頼できるインスタンスで、次の場所に移動します: すべて > コード署名 > 設定 > ガイド付きセットアップ をクリックして、[コード署名] 設定ページを開きます。
2. [インスタンスタイプ] フィールドで、 信頼できるインスタンスを選択します。
3. [次へ] ボタンをクリックします。
4. [実行するアクションを選択します] の [アクション] フィールドで、[証明書の検証をオンにする] を選択します。
5. [次へ] ボタンをクリックします。
6. [顧客署名キー ペアと証明書] セクションの [添付ファイル] の横にある [+ファイルの追加] を選択して、顧客署名目的で使用する暗号化キー ペア (p12 ファイル拡張子) をアップロードします。

   ヒント: [+ファイルを追加] オプションが使用できない場合は、グローバルスコープであり、sn_kmf.cryptographic_manager ロールを持っていることを確認します。

7. [パスワード] フィールドに、アップロードしたキーペアのパスワードを入力します。
8. [インポート] を選択します。
9. [続行] を選択して次のセクションに進みます。
10. [COT 管理キーペアと証明書] セクションの [添付ファイル] の横にある [+ファイルを追加] を選択して、顧客署名目的で使用する暗号化キーペア (p12 ファイル拡張子) をアップロードします。
11. [パスワード] フィールドに、アップロードしたキーペアのパスワードを入力します。

12. [インポート] を選択します。
13. [続行] を選択して次のセクションに進みます。
14. [信頼できるインスタンスタスクを実行] セクションで、すべてのタスクが完了するまで待機します。

    インスタンスは、これらのタスクを自動的に生成して実行します。Vancouver リリースより前にコード署名を使用していた場合は、署名を更新するタスクが作成され、実行されます。

    場合によっては、タスクは必要ありません。このページに「タスクは必要ありません **(No tasks needed)**」と表示されます。
15. [続行] を選択して次のセクションに進みます。
16. [構成ファイルのエクスポート] ページで [エクスポート] を選択し、保護されたインスタンスでコード署名を有効にするために使用する構成ファイルを作成してダウンロードします。
    エクスポートプロセスでは、「保護されたインスタンスでのコード署名エンタープライズの構成」で詳述されている手順で使用するために、XML ファイルがローカルマシンにダウンロードされます。
17. 保護されたインスタンスで、 すべて > システムアップデートセット > 取得済み更新セット.
18. [取得済み更新セット] リストの下で、**[XML** から更新セットをインポート] を選択します。
19. [ファイルを選択] を選択し、構成ファイル (xml ファイル拡張子) を選択します。
20. **[Upload** (アップロード)] を選択します。
21. のコード署名設定ページに戻る すべて > コード署名 > コード署名構成.
22. ウィザードを使用して構成を完了し、証明書の検証のアクティブ化を完了するためのオプションを選択します。

クォーラム制御された証明書の失効

コード署名証明書のクォーラム制御による証明書の失効は、コード署名管理者がコード署名証明書を取り消すための安全なメカニズムを提供します。取り消しプロセスでは、複数のステークホルダーからの承認を必要とする要求を送信します。このワークフローは、偶発的または不正な取り消しを防ぐのに役立ちます。

このトピックでは、クォーラムベースの承認ワークフローを使用して、信頼できるインスタンスと保護されたインスタンスの間で証明書を安全に取り消すためのプロセスの概要について説明します。

- 要求の作成とエクスポート (信頼できるインスタンス):必要な構成プロパティを使用して要求を作成し、クォーラム証明書の失効を開始します。取り消し要求と関連データを含む更新セットをエクスポートします。「失効要求構成をエクスポート」を参照してください。
- インポートと承認 (保護されたインスタンス):保護されたインスタンスに更新セットをインポートします。承認者は通知を受け取り、証明書を削除する前に承認ワークフローを完了する必要があります。「失効要求構成のインポート」を参照してください。
- オプションの MID サーバーの再起動:構成で有効にすると、証明書の失効後に保護されたインスタンスの MID サーバーが再起動されます。これにより、証明書の即時再同期が強制されますが、未処理のイベントやキャッシュされたイベントによりデータが失われる可能性があります。「証明書の失効を承認」を参照してください。

注:

*update set*com.snc.kmf.signature.validity_windowプロパティで定義された期間の後に有効期限が切れます。有効期限が切れた場合は、信頼できるインスタンスから新しい署名付き更新セットをエクスポートできます。この有効期間は、エクスポート、インポート、 コード署名の有効化など、すべての更新セット操作に適用されます。この有効期間は、要求の作成時に指定した要求期間とは無関係です。

失効要求構成をエクスポート

取り消す証明書を選択して、証明書の失効プロセスを開始します。必要な構成プロパティを指定します。このトランザクションを更新セットの一部としてエクスポートします。更新セットは、承認および実行のために保護されたインスタンスにインポートされます。

始める前に

必要なロール:sn_cse.codesigning_admin、sn_cse.quorum_requester、security_admin

手順

1. 信頼できるインスタンスで、次の場所に移動します: すべて > コード署名 > 設定 > ガイド付きセットアップ をクリックし、[コード署名 設定] ページを開きます。
2. [インスタンスタイプ] フィールドで、 信頼できるインスタンスを選択します。
3. [次へ] ボタンをクリックします。
4. [実行するアクションを選択] の [アクション] フィールドで、[証明書の失効] を選択します。
   [認定失効要求] ページが表示されます。
5. [失効する証明書を選択] ページで、失効する証明書を選択し、[失効を開始] を選択します。
6. 認定取り消しの承認要件を設定します。

   a. [クォーラム要件] メニューで、テキストフィールドに適切な情報を入力します。

   クォーラム要件プロパティ

プロパティ	説明
最小承認数	証明書の失効要求を承認するために必要な承認者の最小数。
期間	取り消し要求の有効期限。
承認者	取り消し要求を承認する権限を持つユーザーのメールアドレス。

   b. [要求の説明] フィールドに、証明書の失効を開始する理由を入力します。

   c. **[Save** (保存)] を選択します。
   [構成ファイルのエクスポート] メニューが表示されます。

   d. [構成ファイルのエクスポート] ページで [エクスポート] を選択して、保護されたインスタンスで証明書失効ワークフローを実行するために使用される構成ファイルを作成してダウンロードします。

結果
エクスポートプロセスでは、「保護されたインスタンスでのコード署名エンタープライズの構成」で詳述されている手順で使用するために、XML ファイルがローカルマシンにダウンロードされます。

失効要求構成のインポート

保護されたインスタンスに更新セットをインポートして、証明書の失効プロセスを開始します。承認者はメール通知を受信し、証明書が取り消される前に承認ワークフローを完了する必要があります。承認とは、セキュリティとコンプライアンスの目的で失効が確認および承認され、追跡可能であることを意味します。

始める前に
必要なロール:sn_cse.codesigning_admin、sn_cse.quorum_requester、security_admin

手順

1. 保護されたインスタンスにログインし、次に移動します: すべて > システムアップデートセット > 取得済み更新セット > **XML** のインポート.
   [XML のインポート] ページが表示されます。
2. [ファイルを選択] を選択し、ローカルシステムから構成 XML ファイルを選択します。
3. **[Upload** (アップロード)] を選択します。
4. のコード署名設定ページに戻る すべて > コード署名 > コード署名構成.

   a. [バッチの顧客アップデート] タブで、次の構成ファイルを確認します。
      - コード署名構成プロパティ (time_window)
      - コード署名構成プロパティ (approver_email_ids)
      - コード署名構成プロパティ (minimum_approvals)
      - コード署名構成プロパティ (restart_mid_servers)
      - コード署名クォーラム要求 (CSEQCxxxxxxx)
      - コード署名トランザクション (CSETRANSxxxxxxx)
      - KMF 署名レコード

   b. 選択 バッチ更新セットプレビュー > 更新セットバッチをコミット.

5. のコード署名設定ページに戻る すべて > コード署名 > 設定 > ガイド付きセットアップ.
6. [インスタンスタイプ] フィールドで、[保護されたインスタンス] を選択し、[次へ] を選択します。
   [構成を開始または続行 **(Start or continue your configuration**)] ページが表示されます。
7. [コード署名構成の更新セットがインポートされ、コミットされたことを確認する] チェックボックスをオンにします。 チェックボックスをオンにして、[次へ] を選択します。
   [クォーラムコントロールの構成] ページで、入力した情報を確認し、[クォーラム承認のトリガー] を選択します。
8. 保護されたインスタンスで、[クォーラム承認をトリガー] ページの [認定失効ステータス] を選択して、クォーラムコントロール要求承認の更新を確認します。
9. [承認者の詳細] を選択し、クォーラム要求のステータスを確認します。

証明書の失効を承認

登録済みメールアドレスに送信されるメール承認通知から、証明書の失効要求を確認して承認します。承認通知を確認し、[ここをクリックして承認] または [ここをクリックして却下] リンクを選択

し、保護されたインスタンスにアクセスして対処します。承認要求とコード署名クォーラム要求にメールから直接アクセスすることもできます。

始める前に
必要なロール:sn_cse.codesigning_admin、approver_user

手順

1. 通知メールに記載されているリンクに従って、保護されたインスタンスにアクセスします。
2. [承認] メニューでトランザクション要求をレビューします。
3. [承認] を選択してタスクを承認します。
4. 承認確認ページで、[はい] を選択して特定のコード署名クォーラム要求にアクセスします。
5. コード署名クォーラム要求を確認し、必要に応じて [コメント] メニューにメモを追加します。
6. **[Update** (更新)] を選択します。

結果
コード署名クォーラム要求が承認されます。取り消し要求のエクスポート中に MID サーバーの再起動を有効にした場合は、要求を有効にするために MID サーバーが再起動します。

コード署名を無効にする

保護されたインスタンスでコード署名を無効にします。

始める前に
必要なロール：admin、codesigning_admin

手順

1. 信頼できるインスタンスで、次の場所に移動します: すべて > コード署名 > 設定 > ガイド付きセットアップ をクリックして、[コード署名] 設定ページを開きます。
2. [インスタンスタイプ] フィールドで、 信頼できるインスタンスを選択します。
3. [次へ] ボタンをクリックします。
4. [実行するアクションを選択します] の [アクション] フィールドで、[コード署名を無効にする] を選択します。
5. [構成ファイルのエクスポート] パネルで、[エクスポート] を選択して更新セットをダウンロードします。
   エクスポートプロセスでは、次の手順で使用するために XML ファイルがローカルマシンにダウンロードされます。
6. **[Done** (完了)] を選択します。
7. 保護されたインスタンスにログインします。
8. 「コード署名構成ファイルを保護されたインスタンスにアップロードする」の手順を使用して構成ファイルをアップロードします。
9. 移動先 すべて > コード署名 > 設定 > ガイド付きセットアップ をクリックして、[コード署名] 設定ページを開きます。
10. [インスタンスタイプ] フィールドで、[保護されたインスタンス] を選択します。
11. [次へ] ボタンをクリックします。
12. [コード署名構成の更新セットがインポートされ、コミットされたことを確認してください] チェックボックスを選択します。

13. [次へ] ボタンをクリックします。
14. 証明書のインストールが完了するのを待ちます。
    完了すると、「コード署名の構成が正常に完了しました」というメッセージが表示されます。

結果
システムプロパティを確認することで、正常に完了したことを確認できます。システムプロパティ [sys_property] テーブルで、**com.snc.kmf.signature.validation.flag** プロパティを探し、値が **false** であることを確認します。

コード署名に必要なキーペアと証明書をロードする

コード署名を使用して、指定された信頼できるインスタンスで関係を確立します。この最初のステップでは、2 つの暗号化キーを信頼できる環境にロードして、本番インスタンスを更新するための信頼できるソースを確立します。

始める前に
必要なロール：security_admin、sn_kmf.cryptographic_manager

このタスクについて
関係を確立するための最初のステップは、コード署名を使用して、指定された信頼できるインスタンスに信頼の基盤を確立することです。このタスクを実行するには、以下が必要です。

- コード署名暗号化モジュールにロードするには、2 つの 4096 ビットの RSA 公開鍵/秘密鍵のペアが必要です。
  - cm_code_signing 暗号化モジュール用の 1 つのペア
  - cm_code_attest 暗号化モジュール用の 1 つのペア

  これらのキーの詳細については、「コード署名キーペアと証明書を作成する」を参照してください。

  重要: これらのキーペアは、パブリック認証局によって署名されているか、組織の内部認証局によって署名されている必要があります。証明書に自己署名することはできません。

- リーフと中間証明書を含む公開鍵暗号化標準 #12 (.p12) ファイル

手順

1. キーストアからキーをインポートします。
   a. 移動先 すべて > キー管理 > 暗号化モジュール > すべて.
   b. cm_code_signing という名前の暗号化モジュールを見つけて開きます。
   c. [暗号化仕様] リストで、暗号化仕様の名前を選択して開きます。
   d. [キーストアからキーをインポート] 画面で、[キーのインポート] を選択します。
2. 最初のステップを繰り返して、cm_code_attest という名前の暗号化モジュールをインポートします。
3. [キーストアパスワードを入力してください] フィールドに、RSA 証明書の生成時に作成したチャレンジパスワードを入力します。

   注: 作成したチャレンジパスワードは、ここではキーストアパスワードと呼ばれます。プロセスの他の部分では、これはインポートパスワードまたはエクスポートパスワードと呼ばれる場合があります。どの場合でも、このパスワードは前の手順で作成したチャレンジパスワードと同じです。

4. [キーストア/証明書のインポート] の横にある [参照] ボタンを選択します。
5. 配布証明書を含む公開鍵暗号標準 #12 (.p12) ファイルを選択します (このドキュメントの上部にある「開始する前に」セクションで説明されています)。
6. **[OK]** を選択します。

> **重要:** 独自の内部認証局を使用している場合は、ステップ 5 ～ 6 のプロセスを使用して、内部認証局の中間証明書をアップロードする必要があります。

キーと証明書のインポートが成功すると、確認メッセージが表示されます。

キーと証明書が [X.509 証明書] [sys_certificate] テーブルのインスタンスに存在することを検証できます。これらのレコードのタイプは [信頼ストア証明書]です。

[暗号化モジュール] [sys_kmf_crypto_module] テーブルでキーを検証できます。

次のタスク
証明書を本番環境にエクスポートします。詳細については、「信頼のサークル証明書の準備」を参照してください。

信頼のサークル証明書の準備

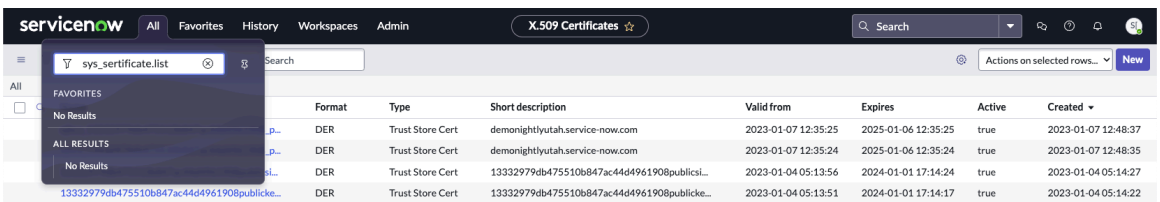
信頼できる環境で、信頼できる証明書を本番環境にエクスポートするための更新セットを作成します。

始める前に
必要なロール：アドミン、security_admin

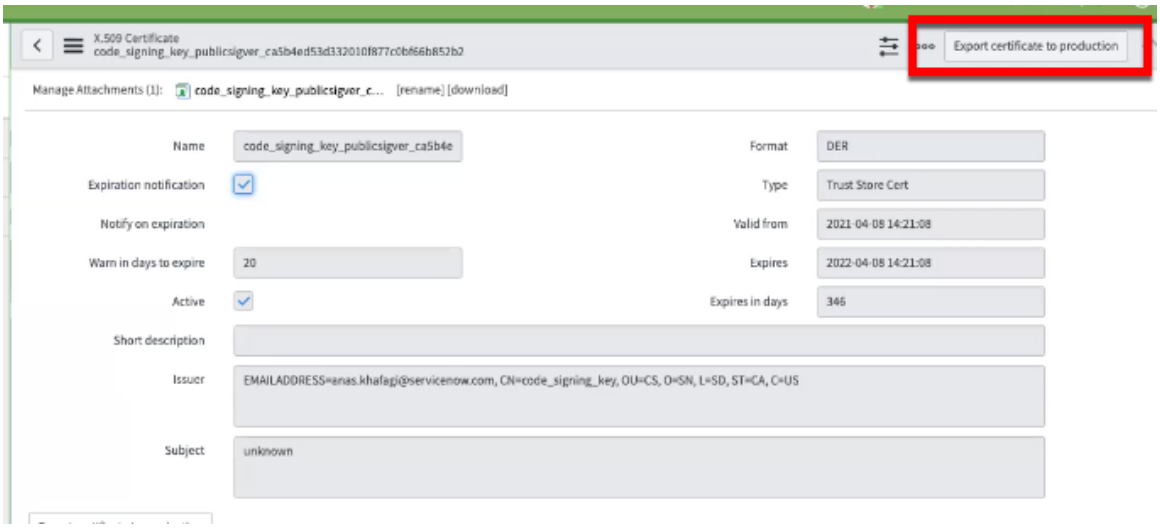
始める前に
信頼できるインスタンス

手順

1. 信頼できる環境で、次の場所に移動します: **sys_certificate.list**.



2. [信頼ストア証明書]タイプで生成された、最後に作成された X.509 証明書を開きます。
   最新のレコードを見つけるために、[作成日時] フィールドをリストに追加する必要がある場合があります。「パーソナルリスト」を参照してください。
3. [証明書を本番環境にエクスポート] を選択します。



証明書とともに署名が作成されます。

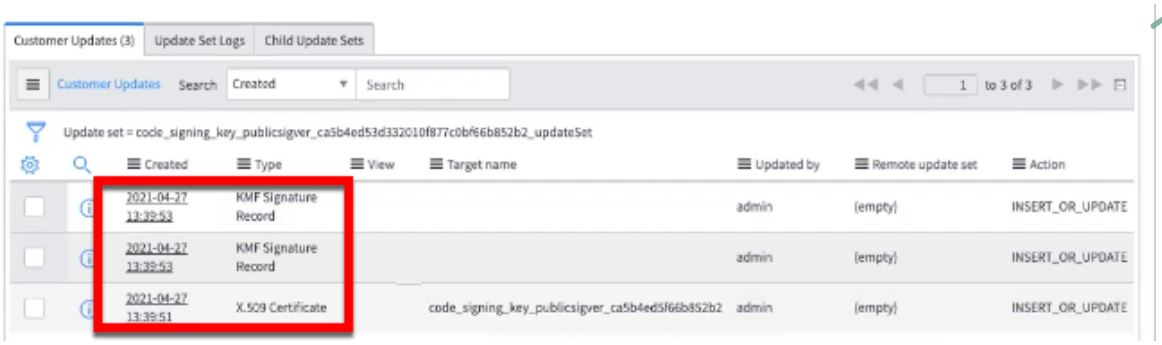
4. 移動先 システムアップデートセット > ローカル更新セット.
5. コード署名の更新を見つけて開きます。

   この更新は、先頭が「code_signing_key_publicsigver」というテキストです。

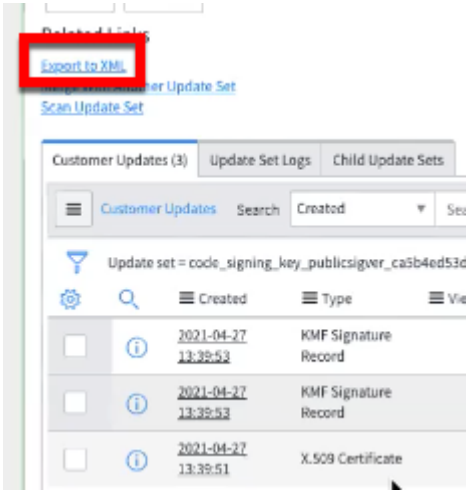
   ここまでの手順によって、新しいコード署名更新セットレコードが作成されました。このレコードを見つけるには、[作成者] フィールドを使用してリストをソートし、[名前] フィールドでを含むレコードを検索します。

6. 関連する署名レコードと X.509 証明書を表示します。

   更新セットには、署名レコードの添付ファイル、テーブル内の署名のエントリ、および証明書が含まれています。



7. **[XML** へのエクスポート] 関連リンクを選択します。



8. 本番環境で更新セットを取得します。
   詳細は、「更新セットを取得」を参照してください。

> **重要:** 2 番目のキーペアに対してこれらの手順を繰り返します。cm_code_attest 暗号モジュールと cm_code_signing 暗号モジュールの両方に対するキーがあることにご注意ください。

信頼のサークルの証明書のインポートとインストール

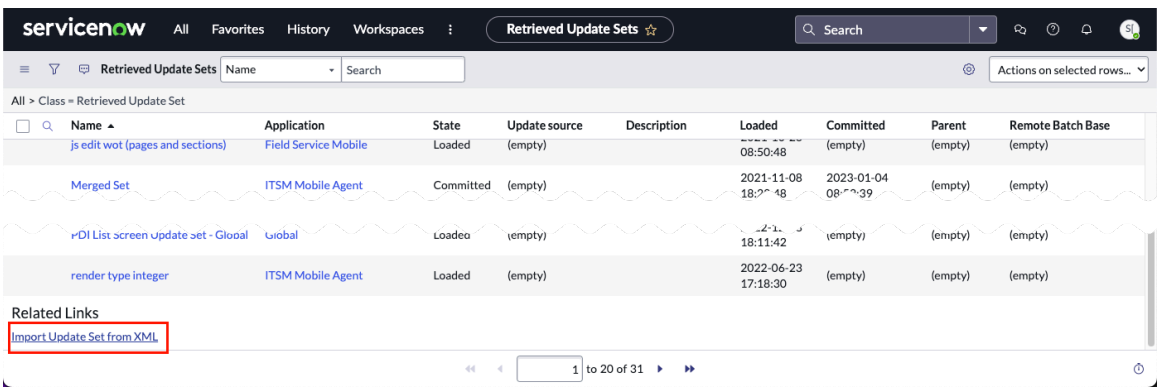
本番環境で更新セットを取得して、2 つのインスタンス間に信頼関係を確立します。信頼できるインスタンスで信頼を表すために作成された証明書が、保護されたインスタンスに受け入れられる必要があります。

始める前に

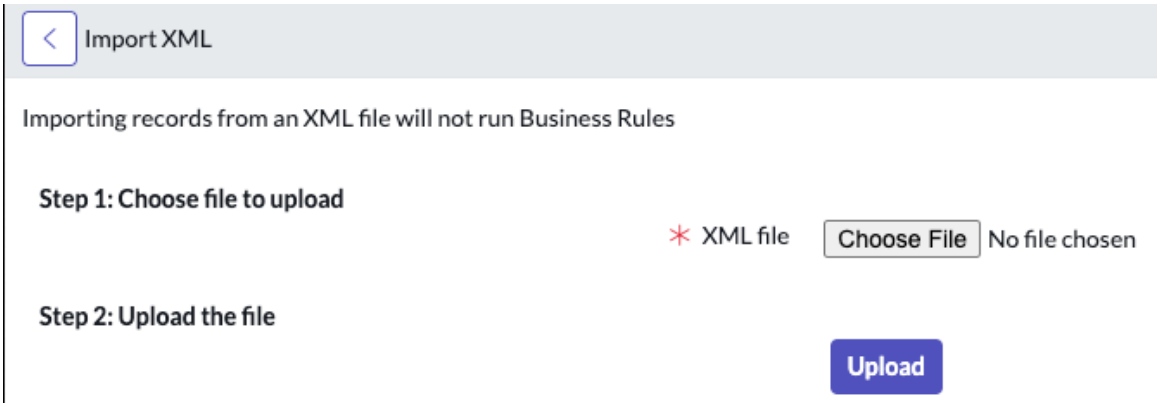
必要なロール：アドミン、security_admin

手順

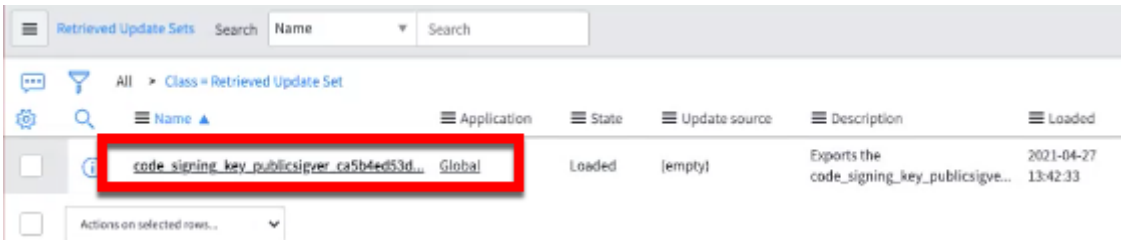
1. 保護されたインスタンスで、次の場所に移動します: システムアップデートセット > 取得済み更新セット.
2. 画面の左下隅にある **[XML** から更新セットをインポート] ボタンを選択します。



3. [ファイルを選択してください] を選択し、「信頼のサークル証明書の準備」でエクスポートした XML ファイルを参照して選択します。

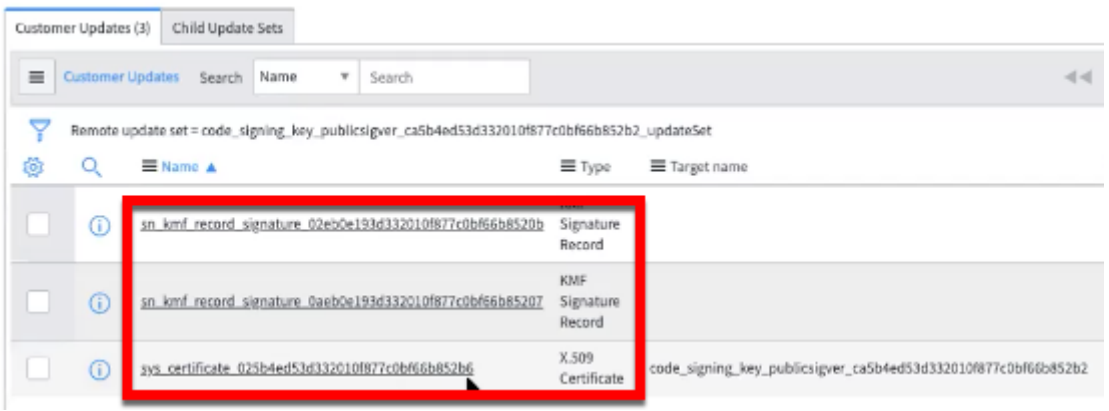


4. **[Upload]** を選択します。
   コード署名更新セットが [取得済み更新セット] テーブルに追加されます。

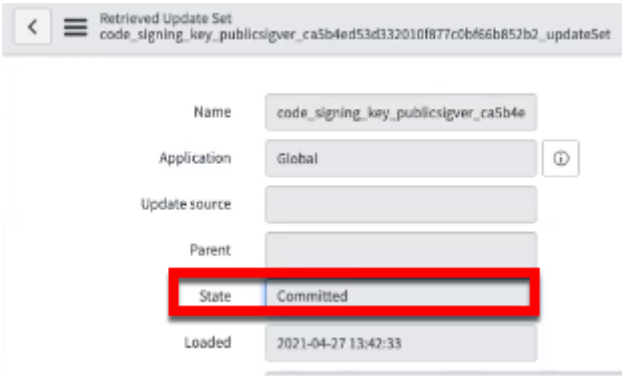


> **注:** 更新セットが表示されない場合は、ロード済みステータスのレコードのリストをフィルタリングし、[ロード完了] フィールドでリストをソートします。

5. 更新セットレコードを開き、顧客アップデートテーブルに次の 3 つのレコードが含まれていることを確認します。
   - 2 つの KMF 署名レコード
   - X.509 証明書

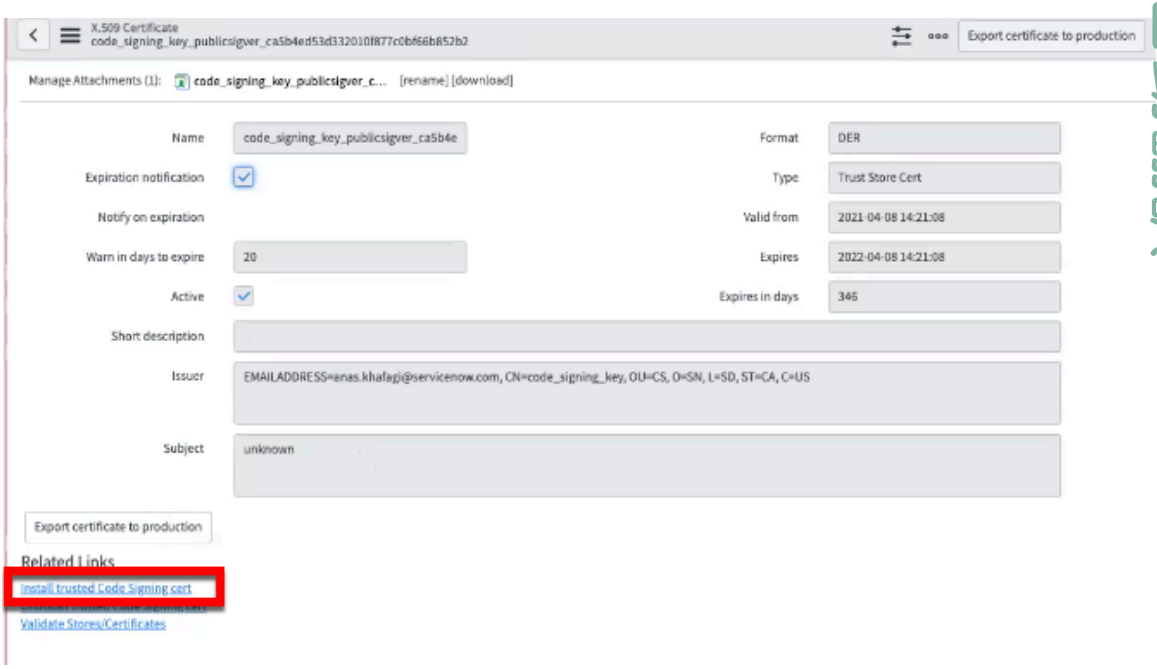


6. [更新セットのプレビュー] を選択します。
7. **[Commit Update Set]** を選択します。



取得済み更新セットの [ステータス] が [コミット済み] に更新されます。

8. ナビゲーターで、 「sn_kmf_record_signature.list」と入力して **[KMF** 署名レコード] リストを開きます。
9. 2 つの KMF 署名レコードを見つけます。
これらは最近作成されたレコードであり、**[KMF** 署名の目的] フィールドの値は [信頼のサークル (Circle of Trust)] です。[作成済み] フィールドをリストに追加するか、現在の日付で作成されたレコードをフィルタリングする必要がある場合があります。
10. ナビゲーターで「sys_certificate.list」と入力し、[簡単な説明] の値が *[code_signing_key]* で、 [タイプ] の値が [トラストストア証明書] になっているレコードを開きます。
11. [信頼できるコード署名証明書をインストールします] を選択します。



証明書タイプが [信頼できる *CodeSigning* 証明書] に更新されます。

> **重要:** 2 番目の証明書についても、この手順を繰り返します。cm_code_attest 暗号モジュールと cm_code_signing 暗号モジュールの両方に対する証明書があることにご注意ください。

結果
信頼できるインスタンスと保護されたインスタンスの間に信頼関係が確立されます。コード署名を有効にする準備ができました。「コード署名を有効にする」を参照して手順を実行します。

コード署名を有効にする

信頼できる非本番インスタンスでコード署名をオンにして、本番インスタンスにリンクする信頼できるインスタンスを特定します。

始める前に

必要なロール：security_admin、および either sn_kmf.crypto_manager または sn_kmf.admin

顧客アドミニストレーターは、ServiceNow プラグインポータルからコード署名 [com.glide.code_signing_enterprise] プラグインにアクセスしてインストールできます。コード署名のユースケースの詳細については、「 コード署名の概要」を参照してください。

このタスクについて

署名済み更新セットを含むコード署名ジョブを使用して、コード署名機能をオンまたはオフにします。この機能を使用するための他の方法はありません。このプロセスには以下が含まれます。

- 信頼できるインスタンスで 2 つのコード署名ジョブを作成します。1 つはコード署名をオンにし、もう 1 つはコード署名をオフにします。
  - [オンにする] スケジュール済みジョブは、署名済みコードの MID サーバー検証プロセスを開始します。
  - [オフにする] スケジュール済みジョブは、署名済みコードの MID サーバー検証を停止します。

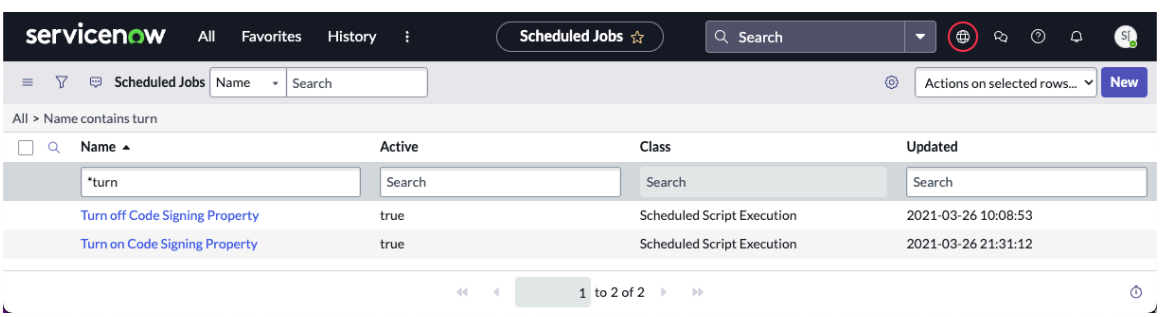
  デフォルトでは、コード署名はすべての MID サーバーに適用されます。ただし、コード署名を MID サーバーの特定のサブセットに制限する必要がある場合は、ECC ファイアウォールを使用してこれを実現できます。

  > **注:** コード署名をオフにすると、システムプロパティが **false** にマークされますが、コード署名の信頼できる友人リストは引き続き使用できます。

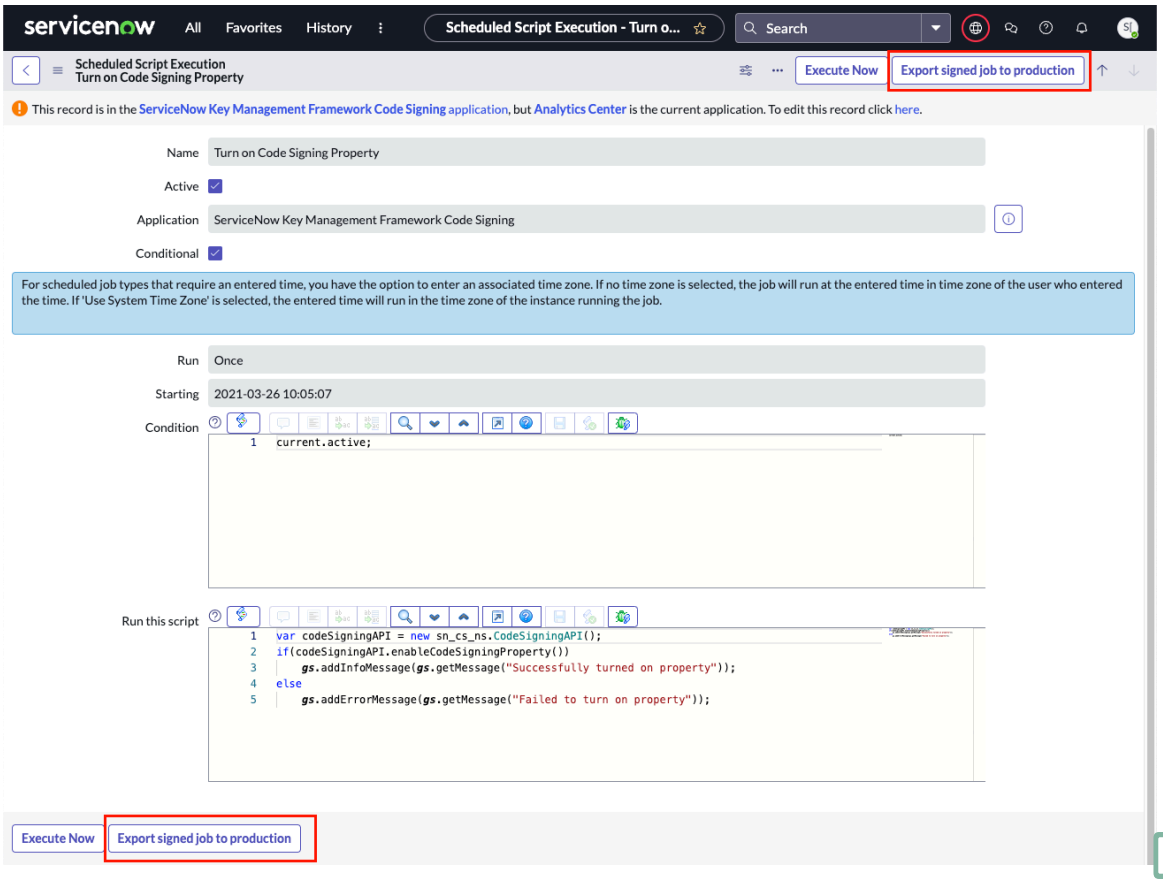
- [コード署名プロパティをオンにする **(Turn on Code Signing Property)]** ジョブを更新セットに配置します。
- ジョブを本番環境に移行します。
- 署名が信頼できるインスタンスからのものであることが検証された場合は、[コード署名プロパティをオンにする **(Turn on Code Signing Property)]** ジョブを使用します。

手順

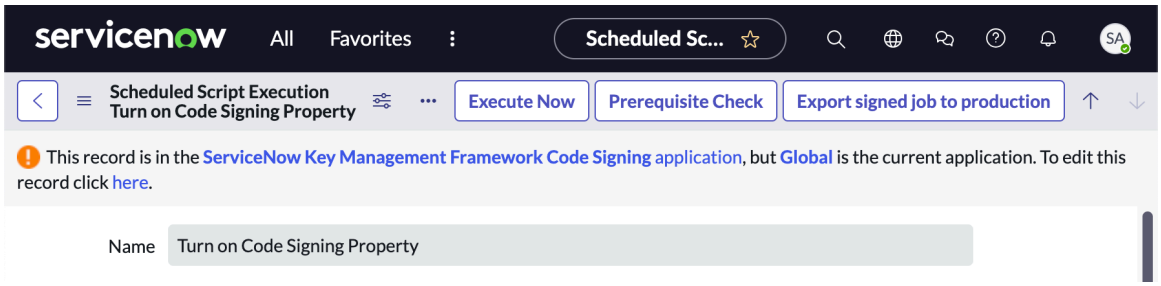
1. 信頼できるインスタンスで、次の場所に移動します: すべて > システム定義 > スケジュール済みジョブ.
2. 名前フィールドで「*Turn」を検索します。
3. [コード署名プロパティをオンにする **(Turn on Code Signing Property)]** を選択します。



[スケジュール済みスクリプト実行] フォームがロードされます。このフォームには [コード署名] プロパティを有効にするための情報が含まれます。ジョブは、ジョブとコード署名プロセスを介した検証済みの署名を含む更新セットを作成します。



4. スクリプトをすぐに実行するには、証明書に署名し、更新セットを作成して、[署名済みジョブを本番環境にエクスポートします] を選択します。
   指定されたスケジュールで実行するようにスクリプトを構成することもできます。
5. 移動先 システムアップデートセット > ローカル更新セット.
6. 各コード署名プロパティの更新セットを開き、**[XML にエクスポート (Export to XML)]** を選択します。
7. 本番インスタンスにログインします。
8. 移動先 システムアップデートセット > 取得済み更新セット.
9. **[XML** から更新セットをインポート] ボタンを選択し、コード署名プロパティ更新セットを選択します。
10. [ファイルを選択] を選択し、更新セットをアップロードしてコミットします。
11. に移動して、スケジュール済みジョブのリストに戻ります すべて > システム定義 > スケジュール済みジョブ.
12. [コード署名プロパティをオンにする **(Turn on Code Signing Property)]** ジョブレコードを開きます。
13. フォームの上部にある [前提条件チェック] ボタンを選択します。



14. 前提条件チェックが完了したら、[今すぐ実行] ボタンを選択します。
[コード署名プロパティをオンにする **(Turn on Code Signing Property)]** スケジュール済みジョブは、署名済みコードの MID サーバー検証プロセスを開始します。
15. ナビゲーターで、「sn_kmf_record_signature.list」と入力して、**[KMF** 署名レコード] リストを開き、**[KMF** 署名の目的] が [信頼のサークル (Circle of Trust)] になっているレコードをフィルタリングします。
信頼関係によってジョブが移動され、ジョブが使用されると署名検証プロセスが実行されます。ジョブ、署名、および証明書がすべて 信頼の輪 (Circle of Trust) の一部である場合、信頼の輪 (Circle of Trust) によるコード署名をオンにすることができます。
16. ナビゲーターで、「sys_properties.list」と入力して、システムプロパティリストを開きます。
17. *com_snc_kmf_signature.validation.flag* を検索し、値が **true** に設定されていることを確認します。
18. 新しいプロパティ *com_snc_kmf_signature.validation.certificate* がテーブルに表示されていることを確認します。

システムプロパティ

com.snc.kmf.signature.validation.certifi...	{"trust_map": {"025b4ed53d332010f877c0bf6...	string	ServiceNow Key Management Framework
com.snc.kmf.signature.validation.flag	true	true | false	Global

本番環境で信頼の輪 (Circle of Trust)ジョブを使用して、信頼関係を確認します。コードに署名しようとする直接ジョブは本番環境では実行できません。設定オプションについては、「コード署名の構成」を参照してください。

コード署名キーペアと証明書を作成する

署名付き証明書に 2 つのキーペアを作成して、保護されたインスタンスと信頼できるインスタンスの間で信頼を確立します。

インスタンス間で信頼を確立するには、cm_code_attest および cm_code_signing の各暗号化モジュールにキーペアと証明書を作成する必要があります。

キーペアと証明書の作成は、OpenSSL ツールなどの、ローカルデバイスにインストールされている暗号化ツールを使用して行われます。このツールの詳細については、「https://www.openssl.org」を参照してください。組織で LibreSSL や GnuTLS などの他の暗号化ツールを使用している場合は、それらの製品のドキュメントで類似した手順について参照してください。

キーペアの仕様

作成するキーペアは、これらの要件を満たしている必要があります。

タイプ	RSA
キー長	4096

署名アルゴリズム	RSASSA_PKCS1_V1_5_SHA_512

証明書の仕様

証明書は、パブリック認証局によって署名されている必要があります。

ECC ファイアウォールでのカスタムルールの指定

MID サーバーで外部通信チャネル (ECC) ファイアウォールを構成するには、着信メッセージを選択的に許可または拒否するカスタムルールを指定して、コード署名の構成をオーバーライドします。

セキュリティアドミニストレーターは、ECC ファイアウォールタグを使用してコード署名の構成をオーバーライドし、MID サーバーに対する特定の操作を許可または拒否することができます。これらのカスタムルールは、agent/boot-config.yaml にある の YAML ファイルで指定する必要があります。

これらのタグはプロトコルに固有です。親タグに指定された構成が、子タグに適用されます。たとえば、HTTP プロトコルが許可されている場合は、REST プロトコルと SOAP プロトコルも許可されます。次の表に、使用可能な親タグと子タグを示します。

親タグ	子タグ
DNS	
HTTP	• REST • SOAP
DIRECTORY_SERVICES	LDAP
SNMP	
SSH	• SCP • SFTP
SYSLOG	
WINDOWS	• CIM • POWERSHELL • WMI • WINRM
JAVASCRIPT	
GROOVY	
VCS	GIT
DATABASES	JDBC
DATA_SOURCES	
INTEGRATION_HUB	

親タグ	子タグ
ITOM	• CLOUD_PROVISIONING_GOVERNANCE • ディスカバリー • EVENT_MANAGEMENT • Health_LOG_ANALYTICS • SERVICE_MAPPING
オーケストレーション	

カスタムルールを設定するには、次のようにします。

1. MID サーバー で、boot-config-sample.yaml ファイルを見つけます。
2. YAML ファイルの名前を boot-config.yaml に変更し、そのファイルを agent/boot-config.yaml の場所に移動します。
3. YAML ファイルで、カスタムルールを指定して変更を保存します。YAML ファイルの例：

```
security:
  eccFirewall:
    mode: enforcing
    rules:
      - tags: [rest]
        action: accept
      - tags: [soap]
        action: accept
      - tags: [jdbc]
        action: reject
```

4. MID サーバー を再起動します。

Root of Trust 構成の変更

Root of Trust (ROT) を使用するように変更して、ServiceNow ビルド証明書 (デフォルト) に依存せず、独自の証明書を信頼して使用します。スクリプトインクルード、ビジネス ルールなどのServiceNowコンポーネントは、ビルド時に ServiceNow ビルド時キーを使用して署名されます (検証証明書は ServiceNow ビルド証明書です)。

Root of Trust の変更

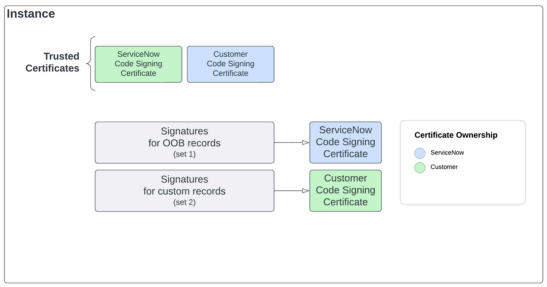
これらのレコードの署名の Root of Trust を変更するには、Root of Trust の変更プロセスに従う必要があります。

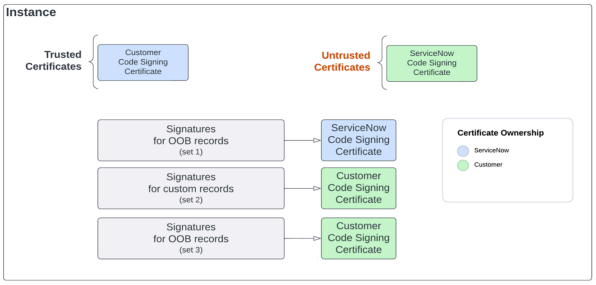
- 提供された証明書を使用して、提供されたすべてのコンポーネントの新しい署名セットを生成して移行します。
- スケジュール済みジョブを使用して Root of Trust プロパティを無効にします。

これらのステップの詳細については、「署名を移行して顧客証明書を使用する」および「ServiceNow Root of Trust の無効化」を参照してください。

署名の生成と検証プロセスへの影響

デフォルトでは、コード署名ビルド証明書は署名検証プロセス中に信頼されます。この変更を行うと、インスタンスは独自のコード署名証明書からの署名のみを受け入れます。





false に設定された **ROT** プロパティ (デフォルト)	**true** に設定された **ROT** プロパティ
• 検証時には、ビルド証明書を含む署名が信頼されます。 • 署名時にキーを指定しない場合、インスタンス署名キーがバックアップキーとして使用されます。 • 署名 REST エンドポイント *api/sn_kmf/signature/certificates* は、インスタンスに存在する他の証明書とともに ServiceNow コード署名ビルド証明書を返します。	• 検証時には、ビルド証明書を含む署名が信頼されません。 • 署名時にキーを指定しないと、署名は実行されません。 • 署名 REST エンドポイント *api/sn_kmf/signature/certificates* では、ServiceNow ビルド証明書 (San Diego、Vancouver PKI、W PKI) が除外されます。

MID サーバーへの影響

ROT プロパティが false に設定されている場合

ROT プロパティをデフォルト値 (false) のままにすることを選択した場合、MID サーバーに影響はありません。

コード署名が有効で、ROT プロパティが true に設定されている場合

- *isTrusted()* API は、ビルド証明書を含む署名に対して false を返します。
- *isTrusted()* API は、証明書を含む署名に対して true を返します。
- 証明書の REST API 呼び出しでは、ビルド証明書が除外されます。
- signature validation failed メッセージなど、MID サーバーの問題がログに表示される場合があります。

署名を移行して顧客証明書を使用する

署名ジョブを実行して、署名を顧客 Root of Trust (ROT) に移行します。

始める前に
必要なロール：admin、security_admin、および sn_kmf.cryptographic_manager

保護された信頼できるインスタンスでコード署名を有効にする必要があります。**com.snc.kmf.signature.validation.flag** システムプロパティが true に設定されていることを確認することで確認できます。

この手順は、インスタンス上の顧客の Root of Trust（ROT）を変更する一連の手順の一部です。このプロセスの概要については、「Root of Trust 構成の変更」を参照してください。

自動翻訳

手順

1. 保護されたインスタンスにログインします。
2. 移動先 すべて > システム定義 > スケジュール済みジョブ.
3. **[ROT -** 顧客証明書を使用して署名を移行するためのレコードの更新セットを生成 **(ROT - Generate Updateset of records to migrate signatures using customer certificate)]** スケジュール済みジョブを見つけて開きます。
4. フォームの下部にある [今すぐ実行] を選択します。
5. 移動先 すべて > システムセキュリティ > セキュリティジョブ > 新規作成.
6. プロンプトで、[署名ジョブ] を選択します。
7. 必要に応じて、[署名ジョブ] フォームのフィールドに入力します。

フィールド	値
名前	ジョブの一意の名前を作成します。
タイプ	[更新セットに署名する **(Sign Update Set)]** を選択します。
テーブル	これまでのステップで作成した更新セットを選択します。更新セットの名前は change_root_of_trust_updateSet で始まります。

8. フォームヘッダーを右クリックし、[保存] を選択してこのレコードを保存します。
9. フォームヘッダーを右クリックし、 エクスポート > **XML** (このレコード) をクリックして、このレコードを XML ファイルとしてエクスポートします。
10. 信頼できるインスタンスにログインします。
11. 移動先 すべて > システムセキュリティ > セキュリティジョブ > すべて.
12. リストヘッダーを右クリックして、**[XML** のインポート] を選択します。
13. **[XML** のインポート] フォームで、[ファイルを選択] を選択し、以前の手順でダウンロードした XML ファイルを選択します。
14. [アップロード] を選択します。
15. リストから、インポートされたセキュリティジョブを開きます。
16. [コード署名ジョブを本番環境にエクスポートします] を選択します。
このアクションにより、ジョブが署名され、保護されたインスタンスにインポートできる新しい更新セットに配置されます。

> **重要:** ジョブに署名した後、10 分以内に次の手順を実行する必要があります。この期間を超えた場合は、これらの手順を使用してジョブに再署名すると、別の署名済み更新セットが作成されます。

17. 移動先 すべて > システムアップデートセット > ローカル更新セット.
18. これまでの手順で作成した更新セットを見つけます。
名前は SIGN_UPDATE_SET_updateSet で始まります。
19. 更新セットを XML ファイルとしてエクスポートするには、**[XML** をエクスポート **(Export XML)]** を選択します。
20. 保護されたインスタンスにログインします。
21. 移動先 すべて > システムアップデートセット > 取得済み更新セット.

22. リストの下部で、**[XML** から更新セットをインポート] を選択します。
23. **[XML** のインポート] フォームで、[ファイルを選択] を選択し、以前の手順でダウンロードした XML ファイルを選択します。
24. **[Upload** (アップロード)] を選択します。
25. 移動先 すべて > システムアップデートセット > 取得済み更新セットをクリックし、SIGN_UPDATE_SET_updateSetで始まる更新セットを開きます。
26. [**Preview Update Set** (更新セットのプレビュー)] をクリックします。
27. プレビューが完了したら、**[Commit Update Set]** を選択します。
28. 移動先 すべて > システムセキュリティ > セキュリティジョブ > すべて.
29. インポートされたセキュリティジョブを開きます。
30. [開始] を選択して署名ジョブを実行します。

    セキュリティジョブが完了すると、ジョブのステータスに関する情報が [サマリー] フィールドに表示されます。

    ジョブの状態が「終了」の場合、更新セットレコードのすべての署名で、顧客が提供した証明書を検証証明書として使用する必要があります。これは、KMF 署名レコード [sn_kmf_record_signature] テーブルで確認できます。

次のタスク

Root of Trust の構成プロセスを続行するには、「ServiceNow Root of Trust の無効化」を参照してください。

ServiceNow Root of Trust の無効化

信頼できるインスタンスでスケジュール済みジョブを実行して、Root of Trust を無効にします。

始める前に

必要なロール：admin、security_admin、および sn_kmf.cryptographic_manager

保護された信頼できるインスタンスでコード署名を有効にする必要があります。**com.snc.kmf.signature.validation.flag** システムプロパティが true に設定されていることを確認することで確認できます。

この手順は、インスタンス上の顧客の Root of Trust（ROT）を変更する一連の手順の一部です。このプロセスの概要については、「Root of Trust 構成の変更」を参照してください。

手順

1. 信頼できるインスタンスにログインします。
2. 移動先 すべて > システム定義 > スケジュール済みジョブ.
3. **[ServiceNow Root of Trust** の無効化] スケジュール済みジョブを開きます。
4. [署名済みジョブを本番環境にエクスポートします] を選択します。
5. 移動先 すべて > システムアップデートセット > ローカル更新セット.
6. **[ServiceNow Root of Trust** の無効化] 更新セットを見つけて開きます。
7. 更新セットを XML ファイルとしてエクスポートするには、**[XML** をエクスポート **(Export XML)]** を選択します。
8. 保護されたインスタンスにログインします。
9. 移動先 すべて > システムアップデートセット > 取得済み更新セット.

10. リストの下部で、**[XML** から更新セットをインポート] を選択します。
11. **[XML** のインポート] フォームで、[ファイルを選択] を選択し、以前の手順でダウンロードした XML ファイルを選択します。
12. **[Upload** (アップロード)] を選択します。
13. 移動先 すべて > システムアップデートセット > 取得済み更新セットをクリックし、[**ServiceNow** の信頼の起点を無効にする] 更新セットを開きます。
14. [**Preview Update Set** (更新セットのプレビュー)] をクリックします。
15. プレビューが完了したら、**[Commit Update Set]** を選択します。
16. 移動先 すべて > システム定義 > スケジュール済みジョブ.
17. 更新セットにインポートされたスケジュール済みジョブを開きます。
18. [今すぐ実行] ボタンを選択してジョブを実行します。

結果
スケジュール済みジョブを実行すると、ROT プロパティが true に設定されます。インスタンスは、顧客の Root of Trust を使用するように構成されます。

コード署名の使用

レコード、メッセージ、および添付ファイルに署名して、データの信頼性と整合性を検証する方法について説明します。

保護されたインスタンスの JDBC データソースレコードに署名する

保護されたインスタンスと信頼できるインスタンスでコード署名を有効にすることで、更新セットを使用して JDBC データソースに署名して検証します。

本番インスタンスで REST および SOAP メッセージに署名する

保護されたインスタンスと信頼できるインスタンスでコード署名を有効にすることで、更新セットを使用して REST および SOAP メッセージに署名して検証します。

保護されたインスタンスのフロー、サブフロー、およびアクションに署名する

更新セットを使用して、保護された信頼できるインスタンスでコード署名を有効にすることで、フロー、サブフロー、およびアクションに署名して検証します。

特定のレコードまたは添付ファイルへの署名

テーブル上のすべてのレコードまたは添付ファイルではなく、特定のレコードまたは添付ファイルに署名するセキュリティジョブを作成します。

スタンドアロン署名ツール

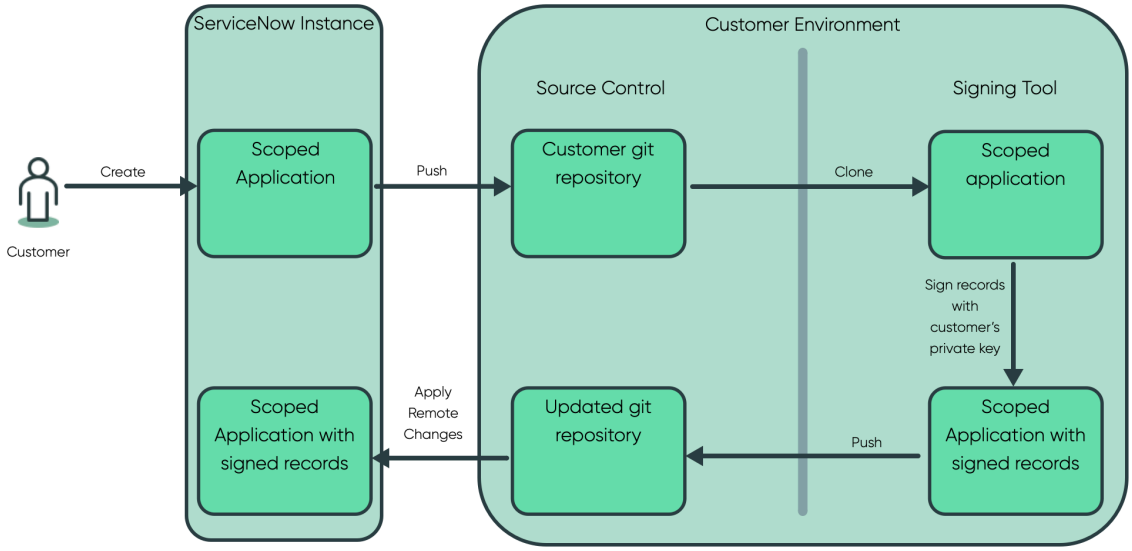
スタンドアロン署名ツールを使用して、独自の秘密鍵を使用して ServiceNow アプリケーションでサポートされているレコードに署名します。

スタンドアロン署名ツール

スタンドアロン署名ツールを使用して、独自の秘密鍵を使用して ServiceNow アプリケーションでサポートされているレコードに署名します。

署名ツールを使用して、ServiceNow アプリのレコードに署名します。このツールでは、独自の秘密鍵を使用してローカル環境のレコードの署名が生成されます。

署名ツールのワークフロー



1. ビジネスルールやスクリプトインクルードなど、署名するレコードを含む既存の ServiceNow アプリケーションを作成または選択します。
2. 環境に存在する Git リポジトリにアプリケーションをプッシュします。

   注: ソースコントロール統合を使用して、Git リポジトリとインスタンス間でアプリケーションを同期できます。この統合の構成と使用の詳細については、「従来 - ソースコントロールの統合」を参照してください。

3. ローカル環境でアプリケーションのクローンを作成します。
4. (同様にローカル環境で) 署名ツールを使用し、秘密鍵を使用してクローンされた ServiceNow アプリケーションからサポートされているレコードに署名します。署名ツールにより署名レコードと X.509 証明書 [sys_certificate] レコードが作成されます。署名ツールの使用の詳細については、「署名ツールの使用」を参照してください。
5. 更新されたアプリケーションを Git リポジトリにプッシュします。
6. インスタンスで、リモート変更を適用することにより、更新されたアプリケーションをインポートします。

署名ツールの使用

署名ツールを使用して ServiceNow アプリケーションでサポートされているレコードに署名する方法について説明します。

始める前に

必要なロール：admin

これらのステップを実行するには、以下が必要です。

- 署名するレコードを含む ServiceNow アプリケーション
- レコードに署名するための秘密鍵
- 実行権限があるローカル環境の *signRecords.sh* スクリプト

重要: *signRecords.sh* スクリプトは署名ツールの jar ファイルの一部として含まれており、次から要求する必要があります ServiceNow カスタマーサポート .

自動翻訳

手順

1. ローカル環境で *signRecords.sh* スクリプトを含むディレクトリに移動します。
2. レコードに署名するには、次のコマンド形式を使用します。

```
./signRecords.sh -d [Path to the root directory of the ServiceNow Application to Sign] -f [Path to the Keystore file]
```

例：

```
./signRecords.sh -d /users/abc/ServiceNow-App-1 -f /users/abc/codesigning.p12
```

3. プロンプトが表示されたら、キーストアのパスワードを入力します。
パスワードがない場合は Enter キーを押します。
4. 出力をレビューして、署名が成功したことを確認します。

```
Sep 26, 2022 2:41:09 PM com.snc.java.commands.ACommand start
INFO: CODESIGN: executing codesigning...
Sep 26, 2022 2:41:09 PM com.snc.core.codesigning.CodeSignerSupplier get
INFO: CODESIGN: signing record for documentId: 65e811327702111057416efe7c5a994f
Sep 26, 2022 2:41:11 PM com.snc.java.commands.ACommand start
INFO: CODESIGN: codesigning successfully completed!
```

前の出力例では、署名ツールは提供されたキーストアファイルを使用してレコードに署名しました。以下についても確認できます。

- スクリプトによりサポートされているレコード 65e811327702111057416efe7c5a994f が検出され、署名されています。
- ServiceNow-App-1 ディレクトリに、sys_certificate.xml と sn_kmf_record_signature.xml という 2 つのレコードが作成されています。

5. スタジオ でリモート変更を適用することにより、更新されたアプリケーションをインスタンスにインポートして戻します。
詳細については、「従来 - リモート変更を適用」を参照してください。

署名ツールの引数

署名ツールで利用可能な引数について説明します。

コマンドラインの引数

引数	必須	説明
-d	はい	署名するプロジェクトのルートディレクトリ。プロジェクトディレクトリ (ランダムな 32 文字の英数字)、sn_source_control.properties ファイル、および <project_name>.iml ファイルが含まれている必要があります。
-f	はい	キーストアのファイルパス。
-a	いいえ	キーストア内の特定のエントリにアクセスするために使用されるエイリアス。
-c	いいえ	レコード署名を 1 つのファイルに連結します。
-k	いいえ	キーストア内に格納されているキーにアクセスするためのパスワード。引数内ではなく、プロンプトが表示されたときにこのパスワードを入力することもできます。

コマンドラインの引数 (続く)

引数	必須	説明
-o	いいえ	既存の sys_cert ファイルの代わりに新しい証明書を使用して署名します。
-p	いいえ	キーストアにアクセスするためのパスワード (パスワードがある場合)。引数内ではなく、プロンプトが表示されたときにこのパスワードを入力することもできます。
-w	いいえ	既存のすべての署名レコードファイルをワイプします。
-h	いいえ	このヘルプメッセージを表示して終了します。

保護されたインスタンスの JDBC データソースレコードに署名する

保護されたインスタンスと信頼できるインスタンスでコード署名を有効にすることで、更新セットを使用して JDBC データソースに署名して検証します。

- 保護されたインスタンスと信頼できるインスタンスの間に 信頼の輪 (Circle of Trust) を確立します。
- 必要なロール：security_admin

注:

- MID サーバー はファイルデータソースを処理しないため、これらのデータソースはコード署名されていません。
- LDAP データソースはコード署名できません。

JDBC タイプの既存のデータソースに署名する

更新セットを使用して、保護されたインスタンスに一括署名ジョブを取り込みます。

始める前に
必要なロール：security_admin

手順

1. 信頼できるインスタンスで、データソースに署名するように KMF 署名ジョブを設定します。

   a. 移動先 システムセキュリティ > セキュリティジョブ > すべて.

   b. [**New**] をクリックします。

   c. フォームに、これらの値を入力します。

フィールド	説明
名前	レコードを識別する名前。
タイプ	暗号化ジョブのタイプ。[レコードに一括署名する**(Mass Sign Records)**] を選択します。
テーブル	レコードの署名元のテーブル。[データソース] を選択します。

d. [コード署名ジョブを本番環境にエクスポートする **(Export Code Signing job to production)**] をクリックします。
更新セットが署名されたことを示す確認メッセージが表示されます。

e. 生成された更新セットを XML ファイルにエクスポートします。

2. 保護されたインスタンスで、更新セットをインポートしてコミットし、信頼できるインスタンスから一括署名ジョブを取得します。

a. 移動先 システムセキュリティ > セキュリティジョブ > すべて.

b. 信頼できるインスタンスからエクスポートされた更新セットを開きます。

c. [開始] をクリックします。
レコードが署名されたことを示す確認メッセージが表示されます。

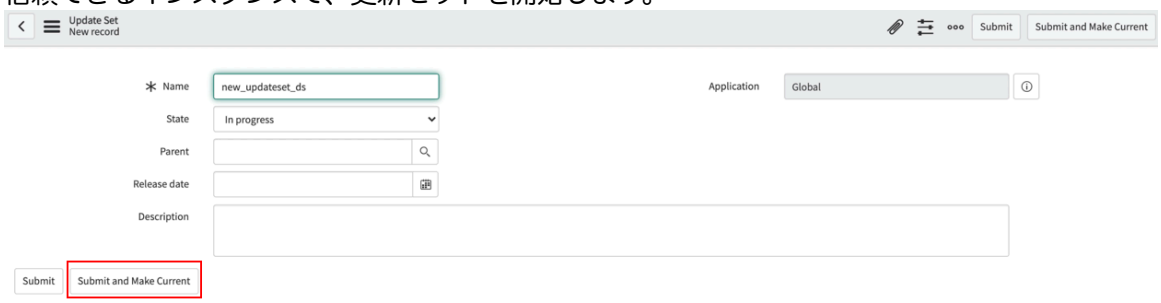
JDBC タイプの新しいデータソースに署名する

更新セットを使用して、署名付き更新セットを保護されたインスタンスに移動します。

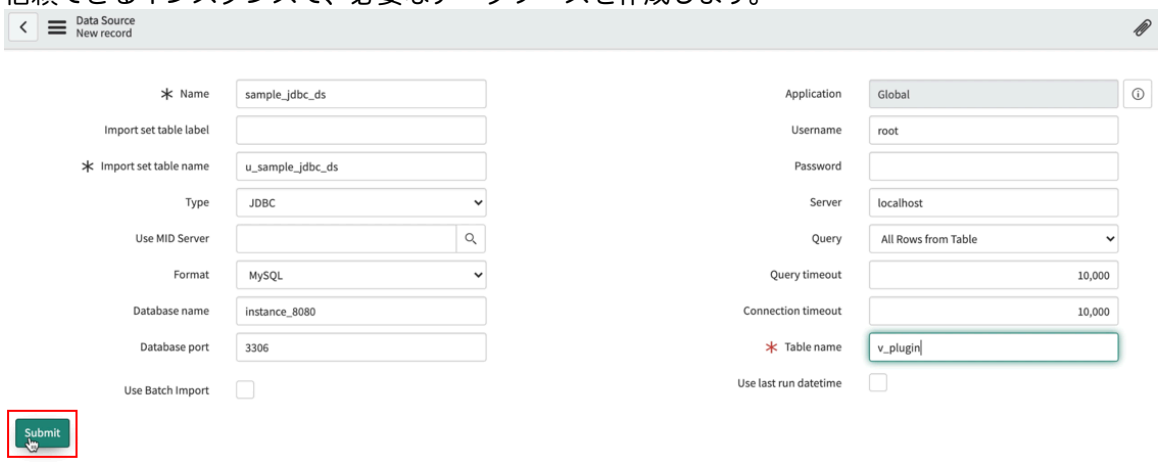
始める前に
必要なロール：sn_kmf.cryptographic_manager

手順

1. 信頼できるインスタンスで、更新セットを開始します。

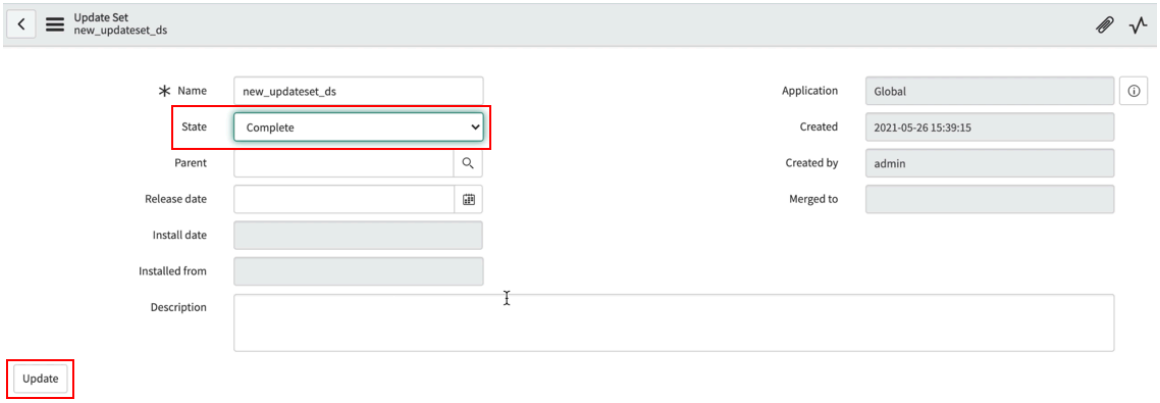


2. 信頼できるインスタンスで、必要なデータソースを作成します。



データソースが更新セットに追加されます。

3. 信頼できるインスタンスで、更新セットのステータスを [完了] に変更し、[更新] をクリックします。



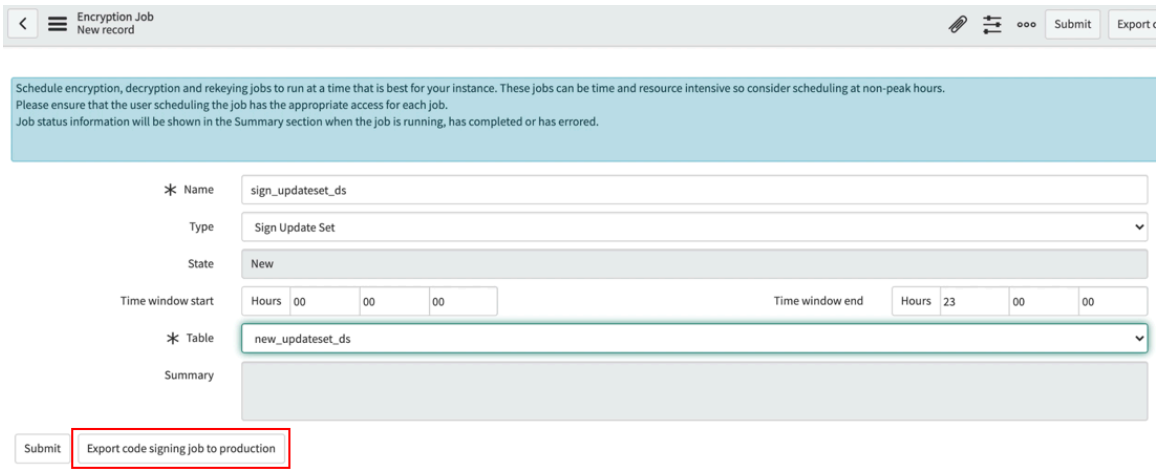
4. 信頼できるインスタンスで、暗号化ジョブを作成して更新セットに署名します。

a. 移動先 システムセキュリティ > セキュリティジョブ > すべて.

b. [**New**] をクリックします。

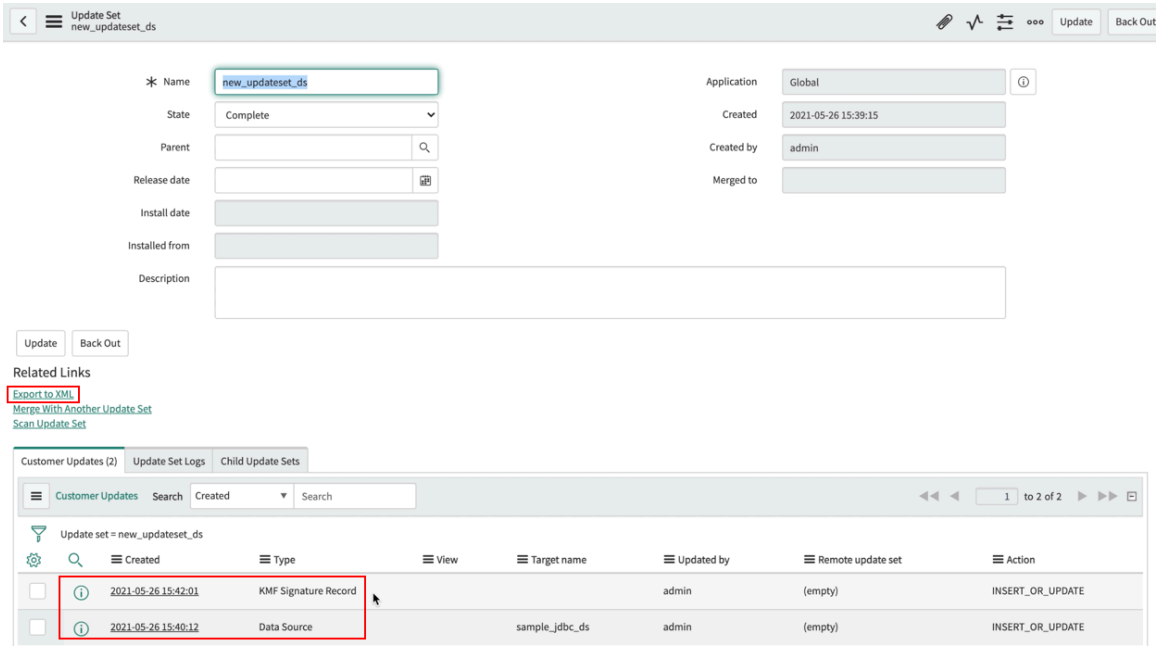
c. フォームに、これらの値を入力します。

フィールド	説明
名前	レコードを識別する名前。
タイプ	暗号化ジョブのタイプ。[更新セットに署名する **(Sign Update Set)**] を選択します。
テーブル	レコードの署名元の更新セット。



d. [送信] をクリックします。

e. [開始] をクリックして更新セットに署名します。



- サマリーが更新され、レコードが署名されます。
- 更新セットが更新され、署名が付加されます。

5. 信頼できるインスタンスで、署名付き更新セットレコードを開き、XML にエクスポートします。
6. 保護されたインスタンスで、署名付き更新セットをインポートします。

   a. 移動先 システムアップデートセット > 取得済み更新セット.

   b. **[XML** から更新セットをインポート] 関連リンクをクリックして、信頼できるインスタンスからエクスポートされた更新セットをインポートします。
   詳細については、「クイックスタート更新セットのインポートとコミット」を参照してください。
   更新セットが正常にコミットされます。

本番インスタンスで **REST** および **SOAP** メッセージに署名する

保護されたインスタンスと信頼できるインスタンスでコード署名を有効にすることで、更新セットを使用して REST および SOAP メッセージに署名して検証します。

始める前に

- 保護されたインスタンスと信頼できるインスタンスの間に 信頼の輪 (Circle of Trust) を確立します。
- 必要なロール：security_admin

既存の **REST** および **SOAP** メッセージに署名する

保護された信頼できるインスタンスでコード署名を有効にすることで、既存の REST および SOAP メッセージに署名して検証します。

始める前に

必要なロール：sn_kmf.cryptographic_manager

手順

1. 信頼できるインスタンスで、UI アクションに署名するように KMF 署名ジョブを設定します。

   a. [KMF 署名構成] に移動します。

   b. フォームに、これらの値を入力します。

   [KMF 署名構成] フォーム

フィールド	説明
テーブル名	Glide テーブル名。たとえば、**[UI** アクション] **[sys_ui_action]** を選択します。
KMF 署名の目的	レコードに署名する目的。**[ECC** キュー] を選択します。
署名生成フィールド	署名するデータソースのフィールド。これらのフィールドの 1 つ以上の値が変更されると、以前に生成された署名が無効になります。[名前] と [スクリプト] を選択します。
署名生成フィルター	レコードに署名するために満たす必要があるフィルター基準。
添付ファイルに署名	Glide レコードの添付ファイルに署名するオプション。
インスタンスキー	インスタンスキーを使用するオプション。

   c. フォームヘッダーを右クリックし、[保存] をクリックします。

2. 信頼できるインスタンスで、必要なレコードに署名します。

   a. 移動先 システムセキュリティ > セキュリティジョブ > すべて.

   b. [**New**] をクリックします。

   c. フォームに、これらの値を入力します。

フィールド	説明
名前	レコードを識別する名前。
タイプ	暗号化ジョブのタイプ。[レコードに一括署名する**(Mass Sign Records)]** を選択します。
テーブル	レコードの署名元のテーブル。**[UI** アクション] を選択します。

   d. [コード署名ジョブを本番環境にエクスポートする **(Export Code Signing job to production)]** をクリックします。
   2 つのローカルで署名された更新セットが作成されます。

- UI アクション構成用の 1 つの更新セット。
- コード署名ジョブをエクスポートするための暗号化ジョブからの別の更新セット。

3. 信頼できるインスタンスで、ローカル更新セットを XML にエクスポートします。
   a. 移動先 システムアップデートセット > ローカル更新セット.
   b. レコードへの一括署名用に作成した更新セットを開きます。
   c. **[XML** へのエクスポート] 関連リンクをクリックし、XML ファイルを保存します。
4. 保護されたインスタンスで、更新セットをインポートします。
   a. 移動先 システムアップデートセット > 取得済み更新セット.
   b. **[XML** から更新セットをインポート] 関連リンクをクリックして、信頼できるインスタンスからエクスポートされた更新セットをインポートします。
   詳細については、「クイックスタート更新セットのインポートとコミット」を参照してください。
   更新セットが正常にコミットされます。
5. 保護されたインスタンスで、[開始] を選択して、信頼できるインスタンスで以前に作成した暗号化ジョブを実行します。
   レコードが署名されたことを示す確認メッセージが表示されます。

新しい **REST** および **SOAP** メッセージに署名する

保護されたインスタンスと信頼できるインスタンスでコード署名を有効にすることで、信頼できるインスタンスからの新しい REST および SOAP メッセージに署名して検証します。

始める前に
必要なロール：security_admin

手順

1. 信頼できるインスタンスで、更新セットを開始します。
2. 信頼できるインスタンスで、必要な REST または SOAP メッセージを作成します。
   メッセージが更新セットに追加されます。
3. 信頼できるインスタンスで、更新セットのステータスを [完了] に変更し、[更新] をクリックします。
4. 信頼できるインスタンスで、暗号化ジョブを作成して更新セットに署名します。
   a. 移動先 システムセキュリティ > セキュリティジョブ > すべて.
   b. [**New**] をクリックします。
   c. フォームに、これらの値を入力します。

フィールド	説明
名前	レコードを識別する名前。
タイプ	暗号化ジョブのタイプ。[更新セットに署名する **(Sign Update Set)]** を選択します。

フィールド	説明
テーブル	レコードの署名元の更新セット。[新しい **Rest V2** 更新セット-1 に署名する **(Sign new Rest V2 update set-1)]** を選択します。

d. [送信] をクリックします。

e. [開始] をクリックして更新セットに署名します。

- サマリーが更新され、レコードが署名されます。
- 更新セットが更新され、署名が付加されます。

5. 信頼できるインスタンスで、署名付き更新セットレコードを開き、XML にエクスポートします。

6. 保護されたインスタンスで、更新セットをインポートします。

a. 移動先 システムアップデートセット > 取得済み更新セット.

b. [XML から更新セットをインポート] 関連リンクを選択して、信頼できるインスタンスからエクスポートされた更新セットをインポートします。
詳細については、「クイックスタート更新セットのインポートとコミット」を参照してください。
更新セットが正常にコミットされます。

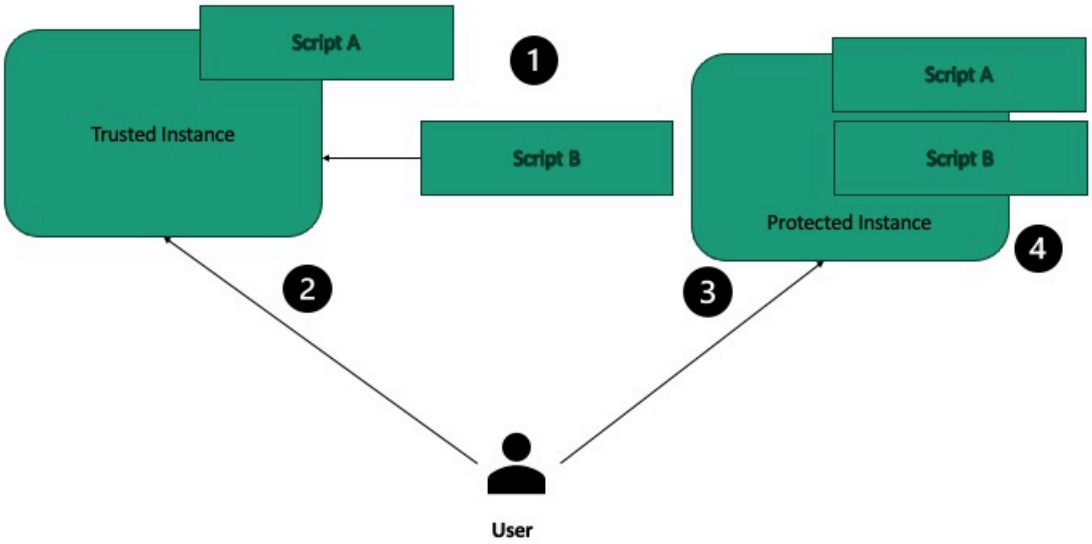
特定のレコードまたは添付ファイルへの署名

テーブル上のすべてのレコードまたは添付ファイルではなく、特定のレコードまたは添付ファイルに署名するセキュリティジョブを作成します。

Vancouver リリース以降、セキュリティアドミニストレーターはセキュリティジョブを使用して、テーブルのすべてのレコードではなく、テーブルの特定のレコードに署名できます。レビュー前のコードに誤って署名しないように、特定のレコードに署名します。

これらのジョブには、署名ジョブの最後に生成される添付ログファイルが含まれています。このログファイルには、署名されたレコードと使用された署名構成に関する情報が含まれています。

署名プロセス



この図は、署名プロセスの使用例を示しています。この例では、 スクリプト B というラベルの付いたスクリプトは保護されたインスタンスにのみ存在し、信頼できるインスタンスにインポートして署名する必要があります。Script A は両方のインスタンスに既に存在し、署名する必要はありません。既にレビューおよび署名されているか、まだレビューされておらず署名すべきではありません。

1. レコードを信頼できる環境にインポートします。
2. 信頼できるインスタンスでレコードに署名するように署名ジョブを作成します。このプロセスの詳細については、「信頼できるインスタンスで特定のレコードまたは添付ファイルに署名するジョブを作成する」を参照してください。
3. 更新セットを使用して、署名された署名ジョブを保護されたインスタンスにインポートします。
4. 保護されたインスタンスで、インポートされた署名ジョブを実行します。

信頼できるインスタンスで特定のレコードまたは添付ファイルに署名するジョブを作成する

信頼できるインスタンスで定義した特定のレコードまたはレコードグループに署名します。

始める前に

必要なロール：security_admin または sn_kmf.cryptographic_manager

手順

1. 移動先 すべて > システムセキュリティ > セキュリティジョブ > 新規作成.
2. [作成するセキュリティジョブのタイプは？] プロンプトで、[署名ジョブ **(Signing Job)]** を選択します。
   新しい [署名ジョブ **(Signing Job)]** レコードが表示されます。
3. 必要に応じて、フォームのフィールドに入力します。

[署名ジョブ **(Signing Job)]** フィールド

フィールド	説明
名前	このジョブの内容を端的に表す名前。

自動翻訳

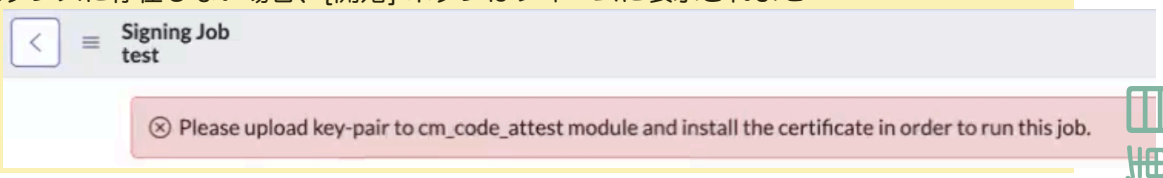
フィールド	説明
タイプ	セキュリティジョブのタイプ。特定のレコードに署名するには、[特定のレコードに署名] オプションを選択します。特定の添付ファイルに署名するには、[特定の添付ファイルに署名] を選択します。
状況	このジョブのステータス。このフィールドは、[新規] という値で始まります。このフィールドは読み取り専用です。
テーブル	署名するレコードまたは添付ファイルを含むテーブル。 添付ファイルに署名する場合は、添付ファイル [sys_attachment] テーブルではなく、添付ファイルが関連付けられているレコードを含むテーブルを選択します。 ヒント: Key Management Framework (KMF) 署名レコード [sn_kmf_record_signature] テーブルをチェックして、選択したテーブルの署名がまだないことを確認します。
レコードのフィルタリング	[署名するレコードを選択 **(Select records for signing)**] テーブルに表示するレコードを制限するために使用するフィルター条件。
署名するレコードを選択	[テーブル] フィールドで選択されたテーブルのレコードリスト。[レコードのフィルタリング] フィールドで作成されたフィルターによって制限されます。 レコードを [利用可能] ウィンドウから [選択済み] ウィンドウに移動して、署名ジョブに含めます。
期間開始	このジョブを実行する期間の開始時間。ジョブは、このフィールドに入力された時間の後に実行されます。 有効な時間値は、24 時間表記に基づく世界標準時です。
期間終了	このジョブを実行する期間の終了時間。ジョブは、このフィールドに入力された時間まで実行されます。ジョブがまだ完了していない場合、ジョブは一時停止し、次の期間の開始時に再開されます。終了時間は開始時間よりも後でなければなりません 有効な時間値は、24 時間表記に基づく世界標準時です。
サマリ	このジョブの実行のサマリ。このフィールドは読み取り専用です。

4. フォームヘッダーを右クリックし、[保存] を選択します。
5. フォームの下部で、[コード署名ジョブを本番環境にエクスポートします] を選択します。
   このアクションにより、署名ジョブが署名され、エクスポートの準備が整います。
6. 移動先 すべて > システムアップデートセット > ローカル更新セット
7. 署名ジョブの更新セットを見つけて開きます。
   [顧客アップデート] タブで、この更新セットに署名ジョブと署名レコードが含まれていることを確認できます。
8. **[XML** にエクスポート **(Export to XML)]** を選択します。

このアクションにより、ローカルデバイスで更新セットを含む XML ファイルが作成されます。

9. 本番インスタンスで、 すべて > システムアップデートセット > 取得済み更新セット > .
10. ページの下部で、**[XML** から更新セットをインポート] を選択します。
11. [ファイルを選択] ボタンを選択し、前のステップで作成した XML ファイルを選択します。
12. [アップロード] を選択します。
    更新セットがロードされ、[取得済み更新セット] リストに表示されます。
13. [取得済み更新セット] リストで、インポートされた更新セットのレコードを開きます。
14. [更新セットのプレビュー] ボタンを選択します。
    プレビューが正常に完了すると、[更新セットのコミット] ボタンが表示されます。
15. **[Commit Update Set** (更新セットのコミット)] をクリックします。
16. すべて > システムセキュリティ > セキュリティジョブ > すべて.
17. 署名ジョブを開きます。
18. [開始] を選択して署名ジョブを実行します。

⚠ 警告: キーペアが本番インスタンスにアップロードされていない場合は、レコードフォームの上部に警告が表示されます。このキーペアがインスタンスに存在しない場合、[開始] ボタンはフォームに表示されません。



次のタスク
完了すると、[サマリ] フィールドにジョブの結果が表示されます。「Mass_Sign_Records-<sys_id>」という名前の署名ジョブレコードに添付されたログファイルもあります。ジョブの詳細については、このレコードと、各署名済みレコードの sys_id のリストを確認してください。

保護されたインスタンスのフロー、サブフロー、およびアクションに署名する

更新セットを使用して、保護された信頼できるインスタンスでコード署名を有効にすることで、フロー、サブフロー、およびアクションに署名して検証します。

- 保護されたインスタンスと信頼できるインスタンスの間に 信頼の輪 (Circle of Trust) を確立します。
- 必要なロール：security_admin

既存のフロー、サブフロー、およびアクションに署名する

本番インスタンスおよび信頼できるインスタンスでコード署名を有効にすることで、既存のフロー、サブフロー、およびアクションに署名して検証します。

始める前に
必要なロール：sn_kmf.cryptographic_manager

手順

1. 信頼できるインスタンスで、ステップインスタンステーブルのレコードに署名します。

   a. 移動先 システムセキュリティ > セキュリティジョブ > すべて.

   b. [**New**] をクリックします。

   c. フォームに、これらの値を入力します。

フィールド	説明
名前	レコードを識別する名前。
タイプ	暗号化ジョブのタイプ。[レコードに一括署名する**(Mass Sign Records)**] を選択します。
テーブル	レコードの署名元のテーブル。[ステップインスタンス] を選択します。

   d. [コード署名ジョブを本番環境にエクスポートする **(Export Code Signing job to production)**] をクリックします。
   2 つのローカルで署名された更新セットが作成されます。
   - KMF 署名の 1 つの更新セット。
   - コード署名ジョブをエクスポートするための暗号化ジョブからの別の更新セット。

2. 信頼できるインスタンスで、ローカル更新セットを XML にエクスポートします。
   a. 移動先 システムアップデートセット > ローカル更新セット.
   b. レコードへの一括署名用に作成した更新セットを開きます。
   c. **[XML** へのエクスポート] 関連リンクをクリックし、XML ファイルを保存します。
3. 保護されたインスタンスで、XML ファイルをインポートします。

   a. 移動先 システムアップデートセット > 取得済み更新セット.

   b. **[XML** から更新セットをインポート] 関連リンクをクリックして、信頼できるインスタンスからエクスポートされた更新セットをインポートします。
   詳細については、「クイックスタート更新セットのインポートとコミット」を参照してください。
   更新セットが正常にコミットされます。

4. 保護されたインスタンスで、以前に信頼できるインスタンスで作成した暗号化ジョブを実行します。
   a. 移動先 システムセキュリティ > セキュリティジョブ > すべて.
   b. 以前に信頼できるインスタンスで作成した暗号化ジョブを開きます。
   c. [開始] をクリックしてジョブを開始します。
   レコードが署名されたことを示す確認メッセージが表示されます。

自動翻訳

新しいフロー、サブフロー、およびアクションに署名する

保護されたインスタンスと信頼できるインスタンスでコード署名を有効にすることで、新しいフロー、サブフロー、およびアクションに署名して検証します。

始める前に
必要なロール：sn_kmf.cryptographic_manager

手順

1. 信頼できるインスタンスで、更新セットを開始します。
2. 信頼できるインスタンスで、必要なフロー、サブフロー、またはアクションを作成して公開します。
   フロー、サブフロー、またはアクションが更新セットに追加されます。
3. 信頼できるインスタンスで、更新セットのステータスを [完了] に変更し、[更新] をクリックします。
4. 信頼できるインスタンスで、暗号化ジョブを作成して更新セットに署名します。

   a. 移動先 システムセキュリティ > セキュリティジョブ > すべて.

   b. [**New**] をクリックします。

   c. フォームに、これらの値を入力します。

フィールド	説明
名前	レコードを識別する名前。
タイプ	暗号化ジョブのタイプ。[更新セットに署名する **(Sign Update Set)]** を選択します。
テーブル	レコードの署名元の更新セット。

   d. [送信] をクリックします。

   e. [開始] をクリックして更新セットに署名します。
      - サマリーが更新され、レコードが署名されます。
      - 更新セットが更新され、署名が付加されます。

5. 信頼できるインスタンスで、署名付き更新セットレコードを開き、XML にエクスポートします。
6. 保護されたインスタンスで、署名付き更新セットをインポートします。

   a. 移動先 システムアップデートセット > 取得済み更新セット.

   b. **[XML** から更新セットをインポート] 関連リンクを選択して、信頼できるインスタンスからエクスポートされた更新セットをインポートします。
   詳細については、「クイックスタート更新セットのインポートとコミット」を参照してください。
   更新セットが正常にコミットされます。

コード署名の健全性とステータスダッシュボード

コード署名の健全性とステータスダッシュボードは、コード署名環境の健全性と構成を一元的に分かりやすいビューで確認できます。これを使用して、問題を特定し、構成の正確性を検証し、中断のない安全で中断のないコード署名操作をサポートします。

コード署名の健全性とステータスダッシュボードは、構成の問題を強調表示し、それらを迅速に解決するのに役立つ直感的なガイダンスを含んでおり、エスカレーションの必要性を減らします。ダッシュボードを使用して、 コード署名 構成が正しく設定され、安全に実行されていることを確認できます。障害につながる可能性のある潜在的な問題を特定し、関係するステークホルダーに正確なガイダンスを提供できるようにします。

ダッシュボードでは、次のパラメーターを検索、フィルタリング、および再スキャンできます。

- 構成
- 証明書ステータス
- プラグインの詳細
- 署名検証証明書
- MID サーバーのセットアップ

コード署名の健全性とステータスダッシュボードにアクセスするには、 すべて > コード署名 > システム健全性 > ダッシュボード.

概要ダッシュボード

概要ダッシュボードは、 コード署名 環境を一元的に表示し、主要コンポーネントとそのステータスに関するリアルタイムのインサイトを提供します。

概要ダッシュボードには、次のレポートが表示されます。

概要ダッシュボード

タイトル	タイプ	説明
Enterprise プラグイン	テキストフィールド	コード署名機能に必要な重要なプラグインとシステムプロパティのステータス。**Enterprise** プラグイン は、署名操作に必要なコアプラグインを表示します。
オプトインプロパティ	テキストフィールド	コード署名機能を有効にする構成フラグ。 **注:** コード署名が特定の要件に従って機能することを確認するには、この設定を アクティブ にする必要があります。
署名検証ステータス	テキストフィールド	信頼できるスクリプトのみが実行されることを確認するために、コード実行時に署名検証を適用するかどうかを示します。

概要ダッシュボード (続く)

タイトル	タイプ	説明
署名状況	円グラフ	現在の検証ステータスを含む、すべてのスクリプト署名の包括的な概要。署名は次のように分類されます。 • 信頼できる 署名は有効であり、信頼できるコード署名証明書に関連付けられています。 • 信頼できない 署名は検証に失敗するか、信頼できない検証証明書にリンクされています。 • 孤立 署名は存在しますが、既知の証明書またはアクティブな証明書にリンクされなくなりました。 注: この情報を使用してスクリプトの整合性を評価し、必要に応じて是正処置を行います。
MID サーバー構成の状態	円グラフ	インスタンス内のすべての MID サーバーのステータスと信頼関係。このセクションを使用して、MID サーバーの合計数を表示し、そのステータスを確認します。 • アクティブな **MID** サーバー: 現在実行され、正常に接続されているサーバーの数。 • 非アクティブな **MID** サーバー:実行されていない、または切断されているサーバーの数。
タイプ別の署名済みレコード	円グラフ	レコードタイプ別の署名済みレコードの分布。次のカテゴリのカバー率 (%) を示します。

概要ダッシュボード (続く)

タイトル	タイプ	説明
		• ビジネスルール:署名されたビジネスルールレコードの割合。 • スクリプトインクルード:署名されたスクリプトインクルードレコードの割合。 • その他:サポートされている他のカテゴリの署名済みレコードの割合。
アプリケーション別の署名済みレコード	円グラフ	さまざまなアプリケーションモジュールにわたる署名済みレコードの分布。コード署名範囲の割合を示します。 • ユーザー管理:ユーザー管理モジュールの署名済みレコードの割合。 • フォームの検証:フォームの検証に関連する署名済みレコードの割合。 • その他:他のすべてのアプリケーションモジュールの署名済みレコードの割合。

署名検証ステータス

さまざまなアプリケーションの有効、無効、孤立した署名のステータスを表示して、コード署名の範囲を評価します。この情報を使用して、追加の注意またはアクションが必要な領域を特定します。

署名検証ステータスダッシュボードには、次のレポートが表示されます。

署名検証ステータスダッシュボード

タイトル	タイプ	説明
スクリプト名	テキストフィールド	システム内の特定のスクリプトに割り当てられた識別子またはタイトル。
タイプ	テキストフィールド	システム内のスクリプトのカテゴリ。スクリプトのロールと機能を定義するのに役立ちます。 例： • ビジネスルール • スクリプトインクルード

署名検証ステータスダッシュボード (続く)

タイトル	タイプ	説明
		• クライアントスクリプト • フローアクション
アプリケーション	テキストフィールド	スクリプトが適用されるシステム内のモジュールまたは領域。 例： • ユーザー管理 • レポート • フォームの検証 • ワークフロー • 通知
ステータス	テキストフィールド	スクリプト署名の現在の検証ステータスを示します。 • 有効:スクリプトの署名は検証され、信頼されています。 • 無効:スクリプト署名が検証されていないか、信頼できないと見なされます。
前回スキャン	テキストフィールド	署名検証のためにスクリプトが最後にスキャンされた日時の形式: DD/MM/YY/H:S (日/月/年/時:分)

コード署名 **MID** サーバー構成

MID サーバーの信頼関係と証明書の設定を管理および構成します。

コード署名 MID サーバー構成ダッシュボードには、次のレポートが表示されます。

コード署名 **MID** サーバー構成

タイトル	タイプ	説明
MID サーバー	テキストフィールド	特定の MID サーバーインスタンスの名前。
ステータス	テキストフィールド	MID サーバーの現在の運用ステータス。 • **Active** (アクティブ) • 非アクティブ
バージョン	テキストフィールド	現在使用中の MID サーバーのソフトウェアバージョン。この

コード署名 **MID** サーバー構成 (続く)

タイトル	タイプ	説明
		情報は、次の目的で使用します。 • インスタンスとの互換性を確認します。 • サーバーがサポートされている最新の機能とセキュリティ更新プログラムを実行していることを確認します。
最終チェックイン	テキストフィールド	MID サーバーがインスタンスと正常に通信した最新の日時。DD/MM/YY/H:S (日/月/年/時:分) の形式

キーペアと証明書

[キーペアと証明書] ダッシュボードには、 コード署名に使用される暗号化キーとデジタル証明書の詳細が表示されます。これには、キータイプ、証明書発行者、有効期限、有効ステータスなどの情報が含まれています。このダッシュボードを使用して、コード署名証明書の認証情報を管理し、その有効性を検証して、安全で信頼できる コード署名 運用を確保します。

キーペアと証明書の構成ダッシュボードには、次のレポートが表示されます。

キーペアと証明書の構成

タイトル	タイプ	説明
証明書名	テキストフィールド	コード署名に使用されるデジタル証明書に割り当てられた一意の識別子 (名前)。
暗号化されたモジュール	テキストフィールド	コード署名秘密鍵を生成、保存、および管理するために使用される暗号化コンポーネント。安全なキー操作を保証し、機密性の高い暗号化マテリアルを不正アクセスから保護します。
タイプ	テキストフィールド	コード署名に使用される暗号化キーまたは証明書の分類。 例： • ServiceNow • サードパーティ
有効期限	テキストフィールド	デジタル証明書の有効期限が切れて使用できなくなる日付。
ステータス	テキストフィールド	デジタル証明書の現在の有効性。

キーペアと証明書の構成 (続く)

タイトル	タイプ	説明
		• 有効:証明書はアクティブであり、 コード署名に使用できます。 • 期限切れ:証明書は有効期限を過ぎており、無効になっています。 • まもなく期限切れ:証明書の有効期限が近づいています。このステータスを使用して、コード署名操作の中断を回避するための更新のための積極的な措置を講じます。
鎖	テキストフィールド	デジタル証明書と、信頼パスを確立する中間証明書およびルート証明書を含む証明書チェーン。 [**View Chain**] を選択して、証明書チェーンを表示します。

コード署名構成

コード署名構成ダッシュボードには、フラグや適用ポリシーなど、環境内の コード署名 を制御するシステムプロパティとキー設定が表示されます。これらの設定により、機能が有効になり、署名の検証が適用され、信頼できるソースが定義されます。

コード署名構成ダッシュボードには、次のレポートが表示されます。

コード署名構成

タイトル	タイプ	説明
設定	テキストフィールド	コード署名に関連する機能または動作を制御する特定の構成プロパティの名前。各設定では、署名の検証、適用、信頼できるソースなどの側面をシステムが処理する方法を定義します。
値	テキストフィールド	特定の設定に割り当てられた現在のステータスまたは入力。これにより、設定の動作が決まります。たとえば、 コード署名 が有効 (true) か無効 (false) かなどです。この値は、システムの コード署名 操作と適用に直接影響します。
最終更新日	テキストフィールド	設定が変更された最新の日時を次の形式で指定します:

コード署名構成 (続く)

タイトル	タイプ	説明
		DD/MM/YY/H:S (日/月/年/時:分)

コード署名の参照情報

参照トピックには、コード署名の管理とトラブルシューティングを行うための追加情報が記載されています。

コード署名とともにインストールされるプロパティ

コード署名により、次のプロパティが追加されます。

トラブルシューティングとログへのアクセス

さまざまなログにアクセスしてトラブルシューティングを行い、失敗の理由を特定します。

コード署名とともにインストールされるプロパティ

コード署名により、次のプロパティが追加されます。

プロパティ	タイプ	説明
com.glide.codesigning.expanded.tracking.enabled	true | false	true の場合、**com.glide.codesigning.expanded_tracking_topic.list** プロパティにリストされている ecc_queue トピックについて、メタスタックの検証の長さが増加します。 **重要:** このプロパティを変更するには、昇格されたセキュリティが必要です。
com.glide.codesigning.expanded.tracking.length	整数	コード署名が有効になっている場合に発生する、コード署名検証のレベル。デフォルト値は 3 です。 **重要:** このプロパティを変更するには、昇格されたセキュリティが必要です。
com.glide.codesigning.expanded_tracking_topic.list	文字列	増加したメタスタックトラッキングの対象となるトピックのカンマ区切りリスト。 **重要:** このプロパティを変更するには、昇格されたセキュリティが必要です。
com.glide.codesigning.tracking.agent.validation.exclusion_list	文字列	コード署名をスキップする必要がある ecc_queue エージェントのカンマ区切りリスト

プロパティ	タイプ	説明
com.glide.codesigning.tracking.debug	true | false	true の場合、コード署名トラッカーのデバッグログが有効になります。
com.glide.codesigning.tracking.enabled	true | false	true の場合、コード署名呼び出し元追跡を有効にします。 **重要:** このプロパティを変更するには、昇格されたセキュリティが必要です。
com.glide.codesigning.tracking.logging.enabled	true | false	true の場合、コード署名追跡のログ記録を有効にします。
com.glide.codesigning.tracking.unsupported_script_tracking.enabled	true | false	true の場合、サポートされていないスクリプトを介して挿入された ecc_queue レコード (検出された場合) は認証されません。 **重要:** このプロパティを変更するには、昇格されたセキュリティが必要です。
com.glide.codesigning.tracking.validation.fail_fast	true | false	true の場合、すべてのスクリプトを検証せずに、最初のスクリプト検証エラーでコード署名検証が失敗します。 **重要:** このプロパティを変更するには、昇格されたセキュリティが必要です。
com.glide.event_handler.code_signing.tracking	文字列	コード署名を新たに有効にする顧客の安全が可能な限り守られるよう設定するイベントハンドラーを定義します。 **重要:** このプロパティを変更するには、昇格されたセキュリティが必要です。
com.glide.web_service_outbound.impl.codesigning.tracking	true | false	true の場合、SOAPMessageV2 コード署名追跡を有効にします **重要:** このプロパティを変更するには、昇格されたセキュリティが必要です。
com.snc.csf.maximum_update_size	整数	コード署名更新セットで許可されているレコードの最大数。この値は、6000 ～ 10000 の値に制限する必要があります。この値を超えると、複数の更新セットが生成され、バッチ処理を有効にするために同じ親更新セットにリンクされます。この制限により、 KB0557104 で説明されている UI の問題が防止されます。

プロパティ	タイプ	説明
com.snc.csf.servicenow_root_of_trust.disabled	true | false	Root of Trust 機能がアクティブかどうか。デフォルト値は false で、ServiceNow ビルド証明書が信頼されていることを意味します。 **重要:** このプロパティは、admin、security admin、および KMF マネージャーのロールを持つユーザーからの署名済みのスケジュール済みジョブを使用することによってのみ変更できます。Root of Trust の変更の詳細については、「Root of Trust 構成の変更」を参照してください。
com.snc.kmf.signature.validation.optin	true | false	true の場合、インスタンスでコード署名が有効になります。 **重要:** このプロパティは、カスタマーサービス & サポートへの要求によってのみ変更できます。
glide.jdbcprobeloader.tracking	true | false	JDBC データソースのコード署名のオン/オフを切り替えます。 **重要:** このプロパティを変更するには、昇格されたセキュリティが必要です。
glide.rest.codesigning.tracking	true | false	true の場合、RESTMessageV2 コード署名追跡を有効にします。 **重要:** このプロパティを変更するには、昇格されたセキュリティが必要です。

コード署名とともにインストールされるロール

コード署名には次のロールが含まれます。

コード署名アドミン **(codesigning_admin)**

コード署名のアドミンロールを使用して、codesigning_manager ロールと codesigning_auditor ロールを他のユーザーにアサインします。

ロールを含む

ロール内に含まれるロールのリスト。

なし。

グループ

このロールがデフォルトでアサインされているグループのリスト。

なし。

特別な考慮事項

より分化したロールが利用可能な場合は、管理者ロールを付与しないでください。

コード署名マネージャー **(codesigning_manager)**

コード署名マネージャーロールを使用して、署名構成を作成および更新し、コード署名ジョブを作成して実行します。

ロールを含む

ロール内に含まれるロールのリスト。

なし。

グループ

このロールがデフォルトでアサインされているグループのリスト。

なし。

特別な考慮事項

なし。

コード署名監査人 **(codesigning_auditor)**

コード署名監査人ロールを使用して、署名構成と署名ジョブを表示します。監査人ロールには、コード署名資産の作成または書き込みアクセス権がありません。

ロールを含む

ロール内に含まれるロールのリスト。

なし。

グループ

このロールがデフォルトでアサインされているグループのリスト。

なし。

特別な考慮事項

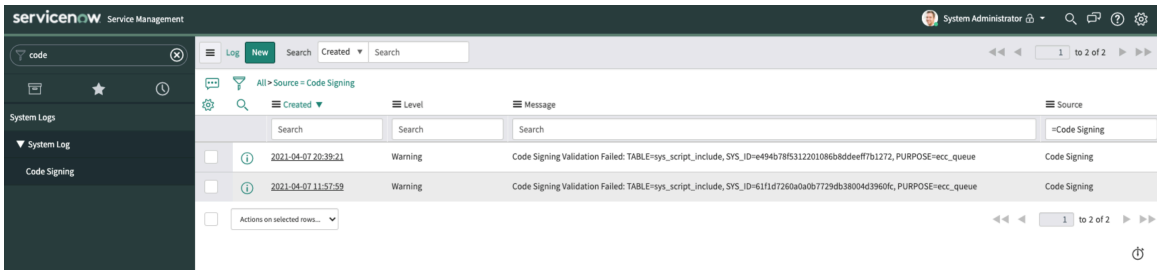
なし。

トラブルシューティングとログへのアクセス

さまざまなログにアクセスしてトラブルシューティングを行い、失敗の理由を特定します。

コード署名ログ

ECC キューレコードのいずれかがコード署名トラッカー API によって署名されていない場合、署名されていないメッセージと必要な詳細がコード署名モジュールに表示されます。移動先 システムログ > システムログ > コード署名 信頼できないレコードのリストにアクセスします。

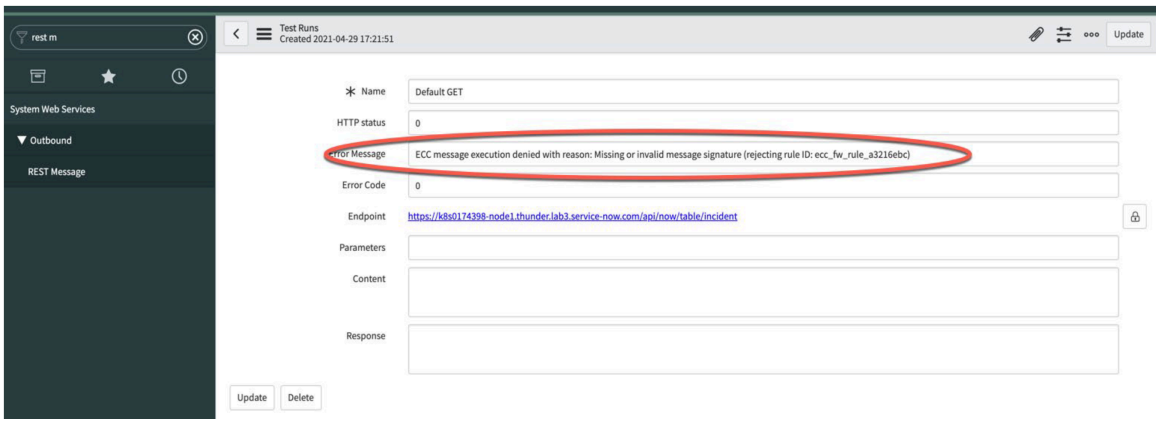


追加のデバッグノードログを表示するには、*com.glide.codesigning.tracking.debug* を有効にし、その値を true に設定します。

MID サーバー での REST メッセージ署名の検証の失敗

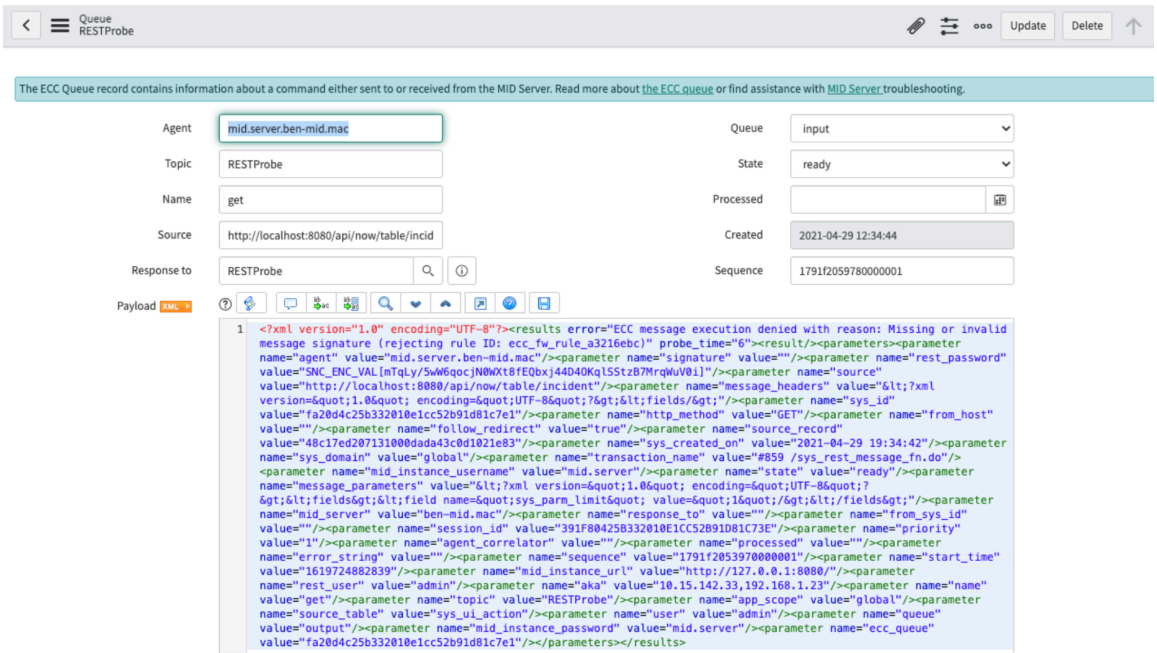
署名検証の失敗に関するエラーメッセージにアクセスするには、 に移動します。 システム **Web** サービス > アウトバウンド > **REST** メッセージ 必要な REST メッセージレコードを開きます。

MID サーバー での署名検証の失敗

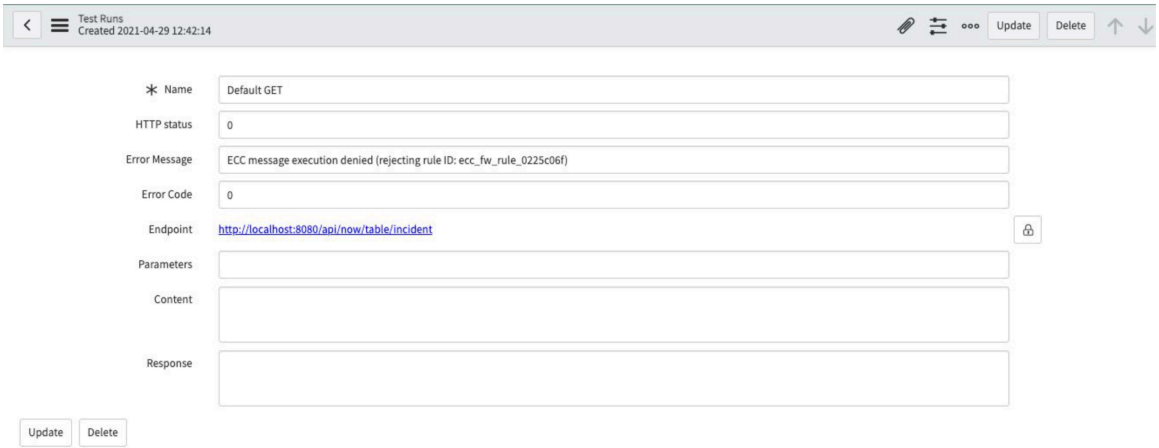


注: ECC ファイアウォール拒否に関連するエラーメッセージは、最初が「ECC メッセージの実行が拒否されました (ECC message execution denied)」です。

MID サーバー で署名検証に失敗した場合の ECC キュー



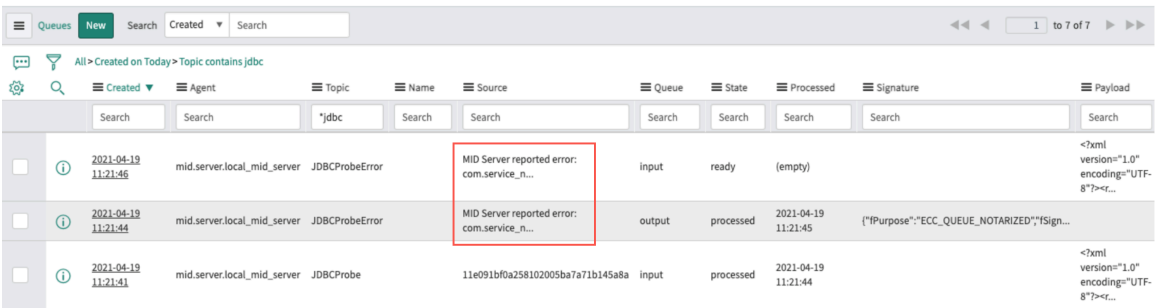
ユーザールールによって **ECC** メッセージがブロックされた場合のエラーメッセージ



JDBC プローブ

署名が無効または欠落している JDBC データソースが MID サーバー で実行された場合、必要な詳細を含むエラーメッセージが表示されます。

ソースには、エラーメッセージの詳細も表示されます。



MID サーバー ログ

詳細な ECC ファイアウォールログ記録を有効にするには、MID サーバー 構成パラメーター *mid.log.level* の値を TRACEに設定します。詳細ログは、次の情報を提供します。

- MID サーバー がブート構成ファイルからロードしたルール。
- ルールの詳細な実行トレース。
- ECC メッセージが承認または却下された原因となった特定のルール。

注: boot-config.xml が無効な場合、MID サーバー は起動に失敗し、失敗の詳細が MID エージェントログに記録されます。

自動翻訳

アンチウイルススキャン

アンチウイルススキャン を使用すると、インシデント、問題、ストーリーなど、システムレコードに添付ファイルによって持ち込まれるウイルス感染からインスタンスを保護できます。

アンチウイルススキャンの詳細

アンチウイルススキャンの価値について説明します。

ウイルス対策保護の構成

ウイルス対策保護の設定方法について説明します。

自動翻訳

感染ファイルの解決

感染ファイルの処理方法について説明します。

アンチウイルススキャンの参照

アンチウイルススキャンの辞書属性に関する情報にアクセスします。

アンチウイルススキャンの詳細

アンチウイルススキャン を使用すると、インシデント、問題、ストーリーなど、システムレコードに添付ファイルによって持ち込まれるウイルス感染からインスタンスを保護できます。

アンチウイルススキャン は添付ファイル [sys_attachment] テーブルに保存されている添付ファイルをスキャンして、ユーザーが感染したファイルをアップロードおよびダウンロードするのを防ぎます。プラットフォームでサポートされているすべてのドキュメントタイプが、アンチウイルススキャンでスキャンされます。

アンチウイルススキャンを有効にすると、添付ファイルテーブル [sys_attachment.do] のすべての添付ファイルがデフォルトでスキャンされます。

ウイルス対策保護 プラグイン (com.glide.snap) は、インスタンスでデフォルトでアクティブ化されています。アドミニストレーターは、スイッチを切り替えることで、インスタンス全体で アンチウイルススキャン 機能を非アクティブ化および再アクティブ化し、構成オプションを設定して、インスタンスのウイルス対策アクティビティを確認することができます。

注:

- アンチウイルススキャン は、政府機関向けコミュニティクラウド (GCC) および商用環境の顧客も利用できます。

  GCC ユーザーは、機能の使用を開始するために (com.glide.snap.fed_enable_scan) プロパティを `true` に設定する必要があります。

  商用ユーザーは *`com.glide.snap.enable_scan`* を `true` に設定する必要があります。

- HTTP および HTTPS 通信プロトコルがサポートされています。
- Edge で暗号化されたファイルはこのスキャンから除外されます。
- アンチウイルス定義は毎日更新されます。
- ファイルサイズが 100 MB を超えるファイルはスキャンされません。

メールスキャン

受信メールは、アンチウイルススキャン ではなく、 システムのメールフィルター によってウイルススキャンが行われます。

テーブルの [ファイルの添付] フィールド

テーブルに [添付ファイル] フィールドを追加すると、zz_yyテーブルが生成されます。これらのテーブルは動的で仮想的なものです。これらは、列タイプ **file_attachment** が親テーブルに追加されると自動的に生成されます。

ユーザーの写真を [ユーザー] テーブルに追加し、フォームビューに組み込むことを検討してください。写真がレコードにアップロードされると、添付ファイルがsys_attachmentテーブルに自動的にアップロードされます。sys_attachmentテーブルは、写真をzz_yyUsersテーブルにマップします。

デフォルトでは、zz_yylive_profile テーブルに添付された添付ファイルのみがスキャンされます。列タイプ **file_attachment** を含む他のテーブルをスキャンするには、システムプロパティ *`com.glide.snap.scan.zz_yytables`* を作成し、テーブル名を挿入します。

例

「zz_yyincident」テーブルと「zz_yycase」テーブルは、列が親テーブル (インシデントとケース) に追加されたときに作成される動的テーブルです。プロパティ値は *`zz_yyincident,zz_yycase`* である必要があります。

このプロパティを設定すると、zz_yyincident テーブルと zz_yycase テーブルの添付ファイルがスキャンされます。

スキャンシナリオ

これらのアップロードおよびダウンロードのシナリオを確認し、システムがレコードに添付されたファイルから潜在的なセキュリティ上の脅威を識別する方法を理解してください。

シナリオ 1 - ファイルのアップロード

1. ユーザーは感染したファイルを知らないうちにレコードにアップロードします。
2. システムはファイルをスキャンして隔離します。
3. ファイルは [添付ファイル] ウィンドウに表示され、利用不可とマークされます。

自動翻訳

4. ユーザーがファイルを選択すると、「Infected_testing.txt ファイルはセキュリティスキャンに合格しませんでした」というエラーメッセージが表示されます。レコード INC0000059 からファイルを削除して、再試行してください。
5. ユーザーとウイルス対策アドミニストレーターにメール通知が送信されます。
6. ユーザーが [添付ファイル] ウィンドウを閉じてレコードに戻ります。感染ファイルは利用不可としてヘッダーに表示されます。例：infected_testing123.txtZ [unavailable]

シナリオ 2 -ファイルのダウンロード

1. ユーザーがレコードを開いて、添付されているファイルをダウンロードします。
2. ファイルが感染していることに気付かず、ユーザーはそのファイルを選択してダウンロードします。
3. システムはファイルをスキャンして、隔離状態に移し、次のようなメッセージを表示します。ファイル infected_testing123.txt はセキュリティスキャンをパスしていないため、ダウンロードすることはできません
4. ユーザーがメッセージを閉じると、画面が更新され、ファイルを使用できないことが表示されます。
5. ユーザーとウイルス対策アドミニストレーターにメール通知が送信されます。

シナリオ 3 - ZIP ファイルのダウンロード

1. ユーザーがレコードを開き、添付されている ZIP ファイルをダウンロードします。
2. システムは ZIP ファイルを個別にスキャンします。
3. 1 つのファイルがセキュリティスキャンに合格せず、利用不可としてマークされます。残りのファイルは圧縮され、正常にダウンロードされます。
4. ユーザーが ZIP ファイルを開くと、正常にダウンロードされたファイルに加えて「error.txt」ファイルが表示されます。このファイルには、スキャンに合格しなかったために ZIP に含まれなかったファイルを示すエラーメッセージが含まれています。
5. ユーザーがレコードを再度開くと、使用できないファイルが [添付ファイル] ウィンドウの [潜在的なセキュリティリスク] セクションに移動され、ダウンロードできないことがわかります。

アンチウイルススキャン の構成

インスタンス全体およびテーブルレベルで アンチウイルススキャン を構成します。

始める前に

必要な役割：antivirus_admin または admin

このタスクについて

アンチウイルススキャン はインスタンスでデフォルトで有効になっており、添付ファイルを自動的にスキャンして、ウイルスに感染しているファイルを特定します。インスタンス全体でスキャンが有効になっていることを確認し、スキャンから除外するテーブルを特定して、機能を設定します。

手順

1. 移動先 すべて > ウイルス対策 > 設定.
2. 機能を設定するときは、次の点を考慮してください。

オプション	説明
アンチウイルススキャンの有効化	アンチウイルススキャンはデフォルトでインスタンスで有効になっているため、そのトグルはオンの位置に設定され、緑色で表示されます。 **注:** プロパティを false に設定する場合は、カスタマーサポートにお問い合わせください。
ウィルス対策スキャナーが利用可能でないときに添付ファイルのダウンロードを許可	このオプションがオンの位置に設定されている場合、スキャナーがタイムアウトして応答を取得できないと、アンチウイルススキャンはバイパスされます。この場合、ファイルのダウンロードはスキャンを完了せずに続行されます。このオプションがオフに設定されている場合、スキャンが正常に完了するまでファイルのダウンロードは禁止されます。
除外されたテーブルのリスト	このリスト内のテーブルに関連付けられている添付ファイルは、アンチウイルススキャンから除外されます。システムがスキャンから除外するテーブルを定義する場合は、ステップ 4 に進みます。

3. [保存] を選択します。
4. 除外されたテーブルのリストに除外するテーブルを追加して、ウイルス対策スキャンからテーブルを除外します。
   a. [システム定義] → [ディクショナリー] に移動します。
   b. スキャンから除外するテーブルを検索し、[タイプ] が [コレクション] に設定されているテーブルを選択します。
   c. [属性] タブで、[新規] を選択します。
   d. [属性] フィールドに Exclude_from_antivirus_scan を追加し、[値] フィールドに「True」と入力します。
   e. [送信] を選択します。

結果
アンチウイルススキャン がインスタンスで有効になっており、[ウイルス対策構成] ページの [除外されたテーブル] のリストに、スキャンから除外したすべてのテーブルが入力されています。

隔離されたファイルの確認

隔離された添付ファイルを確認し、必要に応じてさらにアクションを実行します。

始める前に
必要な役割：antivirus_admin または admin

このタスクについて
[ウイルス対策隔離] ページで隔離されたファイルのエントリを定期的に監視して、次のいずれかのアクションを実行します。

手順

1. 移動先 すべて > ウイルス対策 > 隔離.
2. 使用可能なアクションを実行する、隔離された各エントリの横にあるチェックボックスをオンにします。
3. バナーの [選択した行でアクションを実行] ドロップダウンを選択して、選択した行の隔離されたエントリに実行するアクションを選択します。

アクション	説明
削除	ユーザーが隔離されたファイルを必要としなくなった場合に削除するには、このアクションを選択します。
復元	ステータスのアセスメントまたはユーザー要求に基づいてファイルを復元するには、このアクションを選択します。ただし、ウイルスの脅威を軽減するために、復元する前にサードパーティのウイルス対策製品でファイルをスキャンしてください。復元が完了すると、ユーザーはファイルをダウンロードできるようになります。
ダウンロード	さらにスキャンして分析するために、このアクションを選択してファイルをローカルシステムにダウンロードします。

結果
確認が求められ、入力に応じて選択されたアクションを実行します。

関連トピック

インスタンスセキュリティセンター

ウイルス対策メトリクス

ウイルス対策アクティビティの確認

感染した可能性があるファイルに対して発生したすべてのアクティビティを、検出されて隔離された時点から追跡する、ウイルス対策アクティビティログを確認します。

始める前に
必要な役割：antivirus_admin または admin

このタスクについて
このログは、ディスカバリー、削除、その他の隔離されたファイルイベントなどのウイルス対策アクティビティをキャプチャするレポートとして機能します。

手順

1. 移動先 すべて > ウイルス対策 > アクティビティ.
2. 隔離されたファイルのログを確認します。
   隔離された各ファイルに対して実行されたアクションを [イベント] 列に表示できます。

次のタスク
削除、復元、ダウンロード、またはログに保持するレコードを決定します。「隔離されたファイルの確認」を参照してください。

の辞書属性について アンチウイルススキャン

辞書属性は、辞書レコードが記述するテーブルまたは要素の動作を変更します。アドミニストレーターは、辞書属性の値を設定して、デフォルトのアンチウイルススキャン設定の動作を変更できます。

アンチウイルススキャン の辞書属性

名前	値	ターゲット要素	説明
Exclude_from_antivirus_scan	True/False	任意のテーブル	true の場合、テーブルの添付ファイルはウイルス対策スキャンから除外されます。「アンチウイルススキャン の構成」を参照してください。
Supress_antivirus_email_notification	True/False	任意のテーブル	true の場合、感染した可能性があるファイルが特定されると、プラットフォームで生成されたメール通知の送信を停止します。
Suppress_antivirus_ui_notification	True/False	任意のテーブル	true の場合、感染した可能性のあるファイルが特定されると、プラットフォームで生成された UI 通知を停止します。

関連トピック

辞書属性を使用したテーブルとフィールドの変更

自動翻訳

HTML サニタイザー

HTML フィールドと翻訳された HTML フィールドの HTML マークアップをサニタイズして不要なコードを削除し、クロスサイトスクリプト攻撃などのセキュリティ上の懸念から保護します。

HTML サニタイザーの詳細

HTML サニタイザーの仕組みをご覧ください。

HTML サニタイザーを構成

HTML サニタイザーを設定します。

HTML サニタイザーをアクティブ化

自動翻訳

HTML サニタイザーを有効にする方法について説明します。

HTML サニタイザーの詳細

HTML フィールドと翻訳された HTML フィールドの HTML マークアップをサニタイズして不要なコードを削除し、クロスサイトスクリプト攻撃などのセキュリティ上の懸念から保護します。

HTML のサニタイズを使用して、インスタンス内の HTML コンテンツに潜在的に有害なコンテンツが含まれないようにします。HTML のサニタイズは、インスタンスで不要なスクリプトを実行したり、ユーザーを不要なコンテンツに誘導したりするために使用できる <![CDATA[<script>]]>、<link>、<embed> タグなど、インスタンスを侵害するために使用される可能性のある HTML タグを削除することで機能します。コンテンツのフォーマットを制御する安全なタグは保持されます。アドミニストレーターは、削除または保持するコンテンツをカスタマイズできます。また、サニタイズをすべてのコンテンツに適用するか、指定したフィールドのみに適用するかを制御することもできます。

HTML サニタイザーは、常に保持する必要があるマークアップのビルトイン包含リストをチェックすることで機能します。サニタイザーは、アドミニストレーターがビルトイン包含リストを変更するために使用できる *HTMLSanitizerConfig* スクリプトインクルードを提供します。アイテムを除外リストに追加して、HTML マークアップを削除することもできます。除外リストの内容で包含リストは上書きされます。

次のタイプのアイテムを包含リストと除外リストに追加できます。

- グローバル属性
- HTML 要素

注: デフォルトでは、href や src などの URL 属性は次のプロトコルのみをサポートしています。

- http
- https
- mailto
- data

例：

```
<a href="https://community.servicenow.com/community">ServiceNow コミュニティ</a>
```

注: HTML サニタイザーの使用を制御する *glide.html.sanitize_all_fields* プロパティの詳細については、「インスタンスセキュリティ強化設定」の「HTML サニタイザーを有効にする (セキュリティセンター 1.3 で更新)」を参照してください。

urlAttributes とプロトコルを設定する

HTMLSanitizer スクリプトインクルードで urlAttributes とそのプロトコルを設定できます。例：

```
HTML_WHITELIST : {
urlAttributes: { "protocols" : [ "file", "notes" ] },
            - -
            - -
    }
```

この例では、`メモ` が包含リストに含まれているため、この URL はサニタイズされません。

```
<a title=“Lotus”
 href=“Notes://
ABC/X575C90019DE33/ABC594DCB76D86EB4925653E0011C4C1/ZZ90B7E2D33964749257EE
A003456FD”>Lotus</a></p>
```

デフォルト包含リスト

注: デフォルト包含リストはシステムリストであり、インスタンス内のユーザーはアクセスできません。

```
BUILTIN_HTML_WHITELIST :{

   globalAttributes:{ attribute:["id","class","lang","title","style"],

                              attributeValuePattern:{}},

   label:{ attribute:["for"]},

   font:{ attribute:["color","face","size"]},

   a:{ attribute:["href","nohref","name","shape"]},

   img:{ attribute:["src","name","alt","border","hspace","vspace","align","height","width"},

   table:{ attribute:["border","cellpadding","cellspacing","bgcolor","background","align","no
 resize","height","width","summary","frame","rules"]},


 th:
{ attribute:["background","bgcolor","abbr","axis","headers","scope","nowrap","height","width","al
ign","valign","char off","char","colspan","rowspan"]},


 td:
{ attribute:["background","bgcolor","abbr","axis","headers","scope","nowrap","height","width","al
ign","valign","char off","char","colspan","rowspan"]},

   tr:{ attribute:["background","height","width","align","valign","char off","char"]},

   thead:{attribute:["align","valign","char off","char"]},

   tbody:{attribute:["align","valign","char off","char"]},

   tfoot:{attribute:["align","valign","char off","char"]},

   colgroup:{attribute:["align","valign","char off","char","span","width"]},

   col:{attribute:["align","valign","char off","char","span","width"]},

   p:{attribute:["align"]},

   style:{attributeValuePattern:{"type":"text/css"}}

   canvas:{ attribute:["height","width"]},
```

自動翻訳

```
  details:{ attribute:["open"]},

  summary:{ attribute:["open","valign","char off","char"]},

  button:{ attribute:["disabled","accesskey","type"]},

  form:{},

 input:{ attribute:["size","maxlength","checked","alt","src","type","disabled","readonly","accesskey","border","usemap"]},

  select:{ attribute:["disabled","multiple","size"]},

  textarea:{ attribute:["rows","cols","disabled","readonly","accesskey"]},

  option:{ attribute:["disabled","label","selected"]},

  div:{ attribute:["align"]},

  ol:{ attribute:["start","type","square"]},

  ul:{ attribute:["type","square","itemscope","itemtype","itemref"]},

  li:{ attribute:["value","fb__id","itemprop"]},

  span:{ attribute:["color","size","data-mce-bogus","itemprop","face"]},

  br:{ attribute:["clear"]},

  h3:{ attribute:["itemprop"]},

  html:{ attribute:["xmlns","lang","xml:lang"]},

  link:{ attribute:["rel","type","href","charset"]},

  meta:{ attribute:["name","content","scheme","charset","http-equiv"]},

  pre:{ attribute:["xml:space"]},

  noscript:{},  h1:{},  h2:{},  h4:{},  h5:{},  h6:{},

  i:{},  b:{},  u:{},  strong:{},  em:{},  small:{},  big:{},

  pre:{},  code:{},  cite:{},  samp:{},  sub:{},  sup:{},

  strike:{},  center:{},  blockquote:{},  hr:{},  map:{},

  dd:{},  dt:{},  dl:{},  fieldset:{},  legend:{}, figure:{},  tt:{},

  body:{},  caption:{},  head:{},  title:{},  shape:{},},
```

自動翻訳

HTML フィールドでの変数とテンプレートの使用

HTML/翻訳された HTML フィールドは、デフォルトで HTML サニタイズされます。このプロセスは、入力 HTML をサニタイズして、クロスサイトスクリプティング (XSS) および関連するセキュリティ攻撃から保護します。${description}や{{description}}などのテンプレートまたは変数を格納し、サニタイズ後に真の説明に置き換えると、サニタイズプロセスの有効性が低下します。これは、サニタイズがプレースホルダーテンプレートでのみ呼び出され、HTML コンテンツでは呼び出されないためです。[HTML]/[翻訳された HTML] フィールドに HTML コンテンツのみを格納すると、効果的なサニタイズプロセスを確保できます。

HTML サニタイザーの構成

HTML サニタイザーの構成を変更するには、スクリプトインクルードを変更する必要があります。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > システム定義 > スクリプトインクルード.
2. **HTMLSanitizerConfig** を開きます。
3. 除外リストにアイテムを追加するには、HTML_BLACKLIST クラスを使用します。

   包含リストにアイテムを追加するには、HTML_WHITELIST クラスを使用します。

   次の形式を使用します。

```
HTML_XXXXLIST :{
      globalAttributes :{

         attribute:[attribute-name1,...],
         attributeValuePattern:{ attribute-name2:attribute-value-regex-pattern,...}

      },<html-element-name>:{// Same as Above},----}
```

   - *globalAttributes* には、すべての HTML 要素にグローバルに適用可能な属性または attributeValuePattern アイテムが含まれています。
   - *attribute* は属性のカンマ区切りリストです。
   - *attributeValuePattern* は、属性-値-正規表現-パターンのペアに関する属性のディクショナリです。属性-値-正規表現-パターンは、属性値と一致する必要がある正規表現です。

Example:
次の例を考えてみましょう。

```
HTML_WHITELIST:{
      globalAttributes:{
         attribute:["id","name"],},
      img:{
         attribute:["style","align"],
         attributeValuePattern:{src:".*jpeg"}},
      iframe:{},}
```

次のアイテムが包含リストに追加されます。

自動翻訳

- グローバル属性の ID と名前。これは、すべての要素にグローバルに適用できる文字列のリストです。
- 属性がスタイルと配置である img 要素。
- 画像のソース属性が .jpeg 拡張子のファイルである img 要素。これは、属性値と一致する正規表現パターンの例です。
- iframe 要素。

HTML サニタイザーの有効化

HTML サニタイザーは、システム内のすべての HTML フィールドのサニタイザーを有効または無効にするプロパティを提供します。

始める前に
必要なロール：admin

このタスクについて
デフォルトでは、このプロパティは新しいインスタンスで true に設定されます。

手順

1. ナビゲーションフィルターで、「sys_properties.list」と入力します。
2. *glide.html.sanitize_all_fields* プロパティと *glide.translated_html.sanitize_all_fields* プロパティを **true** に設定します。

   注: このプロパティの詳細については、「インスタンスセキュリティ強化設定」の「HTML サニタイザーを有効にする (セキュリティセンター 1.3 で更新)」を参照してください。

   Trouble?

   [システムのプロパティ] テーブルにプロパティが存在しない場合は追加できます。

個々のフィールドのサニタイズを有効にする

フィールド属性を使用して、個々のフィールドでサニタイザーを有効または無効にすることができます。

始める前に
必要なロール：admin

このタスクについて
最初にサニタイザープロパティを false に設定してから、任意のフォームのフィールドごとにサニタイザーを有効にする必要があります。

手順

1. sys_properties テーブルに移動し、*glide.html.sanitize_all_fields* を **false**に設定します。
   これにより、システム内のすべての HTML フィールドのサニタイザーが無効になります。
2. HTML フィールドを含むフォームに移動します。
3. HTML フィールドラベルを右クリックし、[ディクショナリを設定] を選択します。
   HTML フィールドのディクショナリエントリフォームが開きます。
4. [属性] フィールドに次のいずれかを入力します。

- サニタイズを無効にするには、「html_sanitize=false」と入力します。
- サニタイズを有効にするには、「html_sanitize=true」と入力します。

5. [更新] をクリックします。
6. 翻訳された HTML フィールドの HTML サニタイザーを有効にするには、*glide.translated_html.sanitize_all_fields* プロパティを **true**に設定します。

HTML サニタイザーログ記録を有効にする

HTML サニタイザーが要素または属性を削除すると、それらはシステムログに追加されます。

始める前に
必要なロール：admin

このタスクについて
/syslog_list.do?sysparm_query=source%3DHTMLSanitizer をインスタンス URL に追加して、これらのサニタイズされた要素を確認することができます。

手順

1. これらのサニタイズされた要素を確認するには、/syslog_list.do?sysparm_query=source%3DHTMLSanitizer をインスタンス URL に追加します。
2. ログ記録を有効または無効にするには、glide.html_sanitize.discarded_log.enable プロパティをシステムプロパティに追加し、値を **true** (有効) または **false** (無効) に設定します。
このプロパティはデフォルトで **true** に設定されています。

監査

監査が有効にされたテーブルのレコードの変更を追跡します。デフォルトでは、インシデントテーブル、変更テーブル、および問題テーブルなどに対して行われた変更が追跡されます。

探索

監査の機能とビジネス価値について説明します。

構成

監査の構成方法について説明します。

自動翻訳

表示	参照
	
システム監査テーブルおよびリレーションシップ変更の監査テーブルを確認します。	履歴セットに関する情報にアクセスします。

監査の詳細

監査が有効にされたテーブルのレコードの変更を追跡します。デフォルトでは、インシデントテーブル、変更テーブル、および問題テーブルなどに対して行われた変更が追跡されます。

監査の概要

監査を有効にすると、テーブル内のすべてのレコードの作成、更新、および削除が追跡されます。テーブル内の個々のフィールドを監査する場合は、辞書属性を使用して追跡しないフィールドを非表示にすることができます。

監査情報は次のテーブルに保持されます。

- Audit (監査)
- 履歴セットについて

⚠ **警告:** ワークフローコンテキスト [wf_context] やイベント管理アラート [em_alert] などの大量のトラフィックを受信するシステムテーブルの監査では、パフォーマンスに影響を与える可能性があります。そのため、em_alert テーブル全体を監査することはできません。代わりに、次の方法で選択した対象フィールドを監査します。em_alert テーブルと選択したフィールドの両方で `audit=true` を設定します。監査するフィールドはできる限り少なくしてください。

ユーザーの監査

監査には次のユーザーがいます。

- アドミン
- セキュリティ_アドミン

監査のメリット

メリット	機能	ユーザー
テーブルのすべてまたは一部のフィールドに対する変更を追跡するには、テーブルの監査を有効にします	テーブルの監査の構成	アドミン
監査機能の定義と構成のより高度な方法を体験	Audit Management Console を使用した監査の構成	アドミン
監査データの削除を自動化および簡素化	監査保持のセットアップ	セキュリティ_アドミン

次に探索する内容

監査の使用の詳細については、以下を参照してください。

- テーブルの監査の構成
- Audit Management Console を使用した監査の構成
- システム監査テーブルおよびリレーションシップ変更の監査テーブルの表示
- 履歴セットについて

監査コンポーネント

テーブル、削除、および免除の監査について理解を深めるために、次の監査コンポーネントを確認します。

親テーブルと子テーブルの監査

テーブルは、親または子の監査対象テーブルから監査フラグを取得しません。

- たとえば、構成アイテム [cmdb_ci] テーブルの監査を有効にすると、そのベーステーブルに格納されている CI のみが監査されます。
- 同様に、コンピューター [cmdb_ci_computer] テーブルの監査を有効にすると、構成アイテム [cmdb_ci] テーブルから派生したコンピューター [cmdb_ci_computer] テーブルのフィールドを含む、コンピューターの CI レコードのみが監査されます。

監査システムテーブル

デフォルトでは、システムテーブルからのレコードの削除は監査されません。システムテーブルを監査するには、*glide.ui.audit_deleted_tables* プロパティリスト内のテーブルのリストにそのシステムテーブルを追加します。

フォームまたはリストからの削除の監査

デフォルトでは、フォームから個々のレコードの削除が監査されます。監査されないようにするには、テーブルの辞書属性 *no_audit_delete* を設定します。

テーブル辞書で監査が選択されており、テーブルが *glide.db.audit.ignore.delete* プロパティにリストされていない場合、リストからの削除が監査されます。

> **注:** デフォルトでは、この *glide.db.audit.ignore.delete* プロパティはシステムプロパティ [sys_properties] テーブルにありません。プロパティとその関連付けられた値を変更するには、まず手動で追加する必要があります。ただし、手動で追加すると、次のデフォルト値が上書きされます。
>
> glide.db.audit.ignore.delete = sys_mutex,sys_db_cache,sys_lucene_block,sys_lucene_file,sys_lucene_directory,sys_user_preference,sys_auc cmdb_ci_windows_service, cmdb_sam_sw_install, cmdb_software_instance, cmdb_sam_sw_usage, sam_sw_counter_detail
>
> システムプロパティの追加の詳細については、「システムプロパティを追加する」を参照してください。

デフォルトでは、フォームビュー、リストビュー、またはスクリプト/スケジュール済みジョブのどれを使用してレコードが削除されたかどうかに関係なく、削除の監査が有効になることに注意してください。

監査された情報

監査では次のレコードの変更を追跡します。

- 変更されたレコードの一意のレコード識別子 (sys_id)
- 変更されたフィールド
- 新しいフィールド値
- 以前のフィールド値
- このレコードとフィールドが更新された回数
- 変更された日時
- 変更を加えたユーザー
- 変更理由 (変更と関連する理由がある場合)
- レコードに複数のバージョンがある場合、レコードの内部チェックポイント ID。

監査から除外される情報

テーブルの監査を有効にしていても、一部の更新は監査されません。たとえば、レコードの履歴には 132 件の更新が表示されていても、監査された更新は 7 件のみであるとします。

監査では、次の情報が除外されます。

- アップグレードによる更新。
- インポートセットによる更新。
- 親または子テーブルのレコード。
- no_audit 辞書属性を持つフィールド。
- *glide.ui.audit_deleted_tables* プロパティリストにリストされていないシステムテーブル。
- sys_class_name 列と sys_domain_id 列を除く sys_ prefix (システムフィールド) で始まるフィールド。
- UI ページは、監査ログを作成せずにレコードの更新をトリガーすることがあります。
- 滞留アクティビティモニターがレコードに接触したとき。これにより、有用なデータに干渉するノイズが生じて、インシデントに対してリストにされる何百もの更新が表示されなくなります。

自動翻訳

- パフォーマンスアナリティクス スコアの手動変更。
- 更新セットの適用
- XML のインポート

テーブルの監査

テーブルの監査方法の手順については、「テーブルの監査の構成」を参照してください。

デフォルトでは、監査対象テーブルのすべてのフィールドが追跡されます。次の 2 つの方法のいずれかで、テーブル内のフィールドのサブセットを監査できます。

- テーブル全体の監査を有効にしてから、含めないフィールドを除外することができます。この方法は、すべてではないがほとんどのフィールドを監査する場合に適しており、除外リストと呼ばれます。詳細については、「フィールドを監査対象から除外する (除外リスト)」を参照してください。
- 指定されたフィールドに対してのみ、テーブルの監査を有効にすることもできます。この方法は、少数のテーブルのフィールドのみ監査する場合に適しており、包含リストと呼ばれます。包含リストを使用してフィールドを含める方法の詳細については、「監査にテーブルフィールドを含める (包含リスト)」を参照してください。

キャンセル不可能な監査レコード

新しいデフォルト設定でトランザクションがキャンセルされたときに監査レコードが記録されない可能性を減らします。

ターゲットレコードの書き込み操作と同じトランザクションですぐにレコードを作成するように監査が設定されています。ターゲットレコードが削除されても、監査は **NCA** テスト監査削除モジュールで作成および保持されます。

注: 拡張監査プロセスはデフォルトで有効になっています。glide.db.audit.lazy プロパティが True に設定されている場合、拡張監査プロセスは無効になります。

Zurich リリース以前では、トランザクションがキャンセルされると、特定の監査可能な操作が記録されませんでした。これは、レコードの変更の間にプラットフォームが何らかの操作を実行し、監査の作成前にキャンセルされるためです。しかし現在では、レコードが変更された直後に監査が作成されるため、キャンセルされたトランザクションによって、監査が記録される前に操作が中止される可能性が低くなります。

監査は、トランザクションと同じスレッドに記録されるようになりました。以前の監査はバックグラウンドスレッドで作成されていました。この変更により、 glide.db.audit.lazy プロパティのデフォルト値が True から False に再定義されます。このプロパティは通常、プロパティテーブルで定義されていません。大半のインスタンスで新しいデフォルト値と動作の使用が開始されるからです。インスタンスによっては、このプロパティがすでに存在し、True に設定されている可能性があります。これは、これらのインスタンスがこの変更を使用して動作を監査できないことを意味します。

注: 更新を利用するには、このプロパティを削除することをお勧めします。

テーブルの監査の構成

テーブルの監査を有効にして、テーブルのすべてまたは一部のフィールドの変更を追跡できます。

始める前に

注: 暗号化されたフィールドは設計上、監査されません。この動作は設定できません。

必要なロール：admin。

手順

1. 移動先 すべて > システム定義 > ディクショナリ.
   辞書エントリのリストが表示されます。リストには、各テーブルの行と、テーブルの各列 (フィールド) の行が含まれます。
2. 辞書エントリのリストで、監査するテーブルに対応する列 (cmdb_ci_computer) を探します。
   正しいテーブル名、列名の空のエントリ、および コレクションの種類の行を探して、テーブルの列の行とテーブル自体の行を区別できます。
3. テーブルに対応する辞書エントリを選択します。
   辞書エントリのフォームが表示されます。
4. [監査] チェックボックスをオンにします。
5. [更新] を選択します。

次のタスク
テーブル内のいくつかのフィールドのみを監査する場合、テーブルの包含リスト監査を有効にする。ほとんどのフィールドを監査して、一部のフィールドを除外する場合、「フィールドを監査対象から除外する (除外リスト)」を参照してください。

テーブルの包含リスト監査を有効にする

明示的に指定したフィールドのみをテーブルで監査できるようにします。これは、監査対象テーブル内の少数のフィールドのみを監査する場合に便利です。

始める前に
テーブルで監査を有効にする必要があります。

必要なロール：admin

手順

1. 移動先 すべて > システム定義 > ディクショナリ.
   辞書エントリのリストが表示されます。リストには、各テーブルの行と、テーブルの各列 (フィールド) の行が含まれます。
2. 必要に応じて、リストビューをカスタマイズして、属性の列を表示します。
3. 辞書エントリのリストで、監査するテーブルに対応する列 (cmdb_ci_computer) を探します。
   正しいテーブル名、列名の空のエントリ、および *collection* の種類の行を探して、テーブルの列の行とテーブル自体の行を区別できます。
4. 該当する行の [属性] フィールドに「*audit_type=whitelist*」と入力します。

次のタスク
テーブルのどのフィールドを監査するか指定します。

フィールドを監査対象から除外する (除外リスト)

これらのフィールドを監査から除外することで、ServiceNow AI Platform が監査対象テーブルのフィールドのサブセットを追跡しないようにすることができます。

始める前に
テーブル内のフィールドを監査から除外するには、まずそのテーブルの監査を有効にする必要があります。

必要なロール：admin

このタスクについて
監査可能なテーブルのほとんどのフィールドを監査する場合は、一連のフィールドを除外リストに追加します。いくつかのフィールドのみを監査する場合は、代わりに包含リストの手順に従います。

注: ジャーナルベースのフィールドの監査を無効にすると、アクティビティフォーマッターなどの機能に影響を与える可能性があります。詳細については、「KB0743142」を参照してください。

手順

1. 移動先 すべて > システム定義 > ディクショナリ.
2. 必要に応じて、リストビューをカスタマイズして、属性の列を表示します。
3. 監査から除外するテーブルとフィールド (列) に対応する行に移動します。
4. その行の属性の列で、「no_audit」と入力します。

監査にテーブルフィールドを含める (包含リスト)

監査対象テーブルのフィールドのサブセットを追跡するには、それらのフィールドを包含リストに追加します。

始める前に
テーブルのフィールドを包含リストに追加するには、まずそのテーブルの監査を有効にし、そのテーブルに対して包含リストの監査を有効にする必要があります。

必要なロール：admin

このタスクについて
監査対象テーブルの少数のフィールドのみを監査する場合は、一連のフィールドを包含リストに追加します。ほとんどのフィールドを監査し、少数のフィールドのみ除外する必要がある場合は、代わりに、除外リストの手順に従います。

手順

1. 移動先 すべて > システム定義 > ディクショナリ.
2. 必要に応じて、含めるリストビューをカスタマイズして、属性の列を表示します。
3. 包含リストに追加するテーブルとフィールド (列) に移動します。
4. [属性] フィールドに「*audit=true*」と入力します。

システムテーブルの監査を有効にする

sys_ prefix があるテーブルからの削除は、デフォルトでは監査されません。これらのテーブルからの削除を追跡するには、テーブル名を *glide.ui.audit_deleted_tables* プロパティに追加します。削除済みレコードの復元プラグインを有効にすると、このプロパティにいくつかのデフォルト値が追加されます。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > システムプロパティ > **UI** プロパティ.
2. 削除の監査プロパティを持つシステムテーブルのリスト (「sys_」で始まる、カンマ区切り) を見つけます。

3. テーブル名を追加または削除します。
   テーブル名は、スペースを入れずにカンマで区切る必要があります。
4. **[Save** (保存)] を選択します。

> 注: 監査の詳細については、「sys 監査テーブルの理解」を参照してください。

Audit Management Console を使用した監査の構成

監査管理コンソールモジュールを使用すると、インスタンス内で監査機能を定義および構成するためのより高度な方法を体験できます。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > 監査管理コンソール.
   インスタンス内のすべてのテーブルのリストが表示されます。

   > 注: デフォルトでは、監査が有効になっているテーブルのリストが表示されます。

   [無効] タブを選択すると、監査が無効になっているテーブルの一覧を表示できます。[すべて] タブを選択して、監査オプションを有効または無効にしたインスタンス内のすべてのテーブルを表示することもできます。
2. 監査構成を更新するテーブルをリストから選択します。
   それぞれの列を含むテーブルが表示されます。選択したテーブルの列の合計数を確認することもできます。

   > 注: [列] と [保持] は、テーブルに表示されるタブです。

3. 選択したテーブルの監査を有効にするか無効にするかに応じて、[監査] トグルを変更します。

   > 注: テーブルで監査が有効になっている場合、テーブル内のすべての列とフィールドがデフォルトで有効になります。

4. オプション: 監査が有効なテーブル内のすべての列を有効にしない場合は、[監査ステータスの編集] を選択します。
   [編集する列を選択] モーダルが表示されます。
5. オプション: 有効にしない列の選択を解除します。
   [利用可能な列] リストから列をオンにして、任意の列を追加することもできます。
6. [保存] を選択して、最新の変更を保存します。
   [すべてクリア] を選択して、有効になっているすべての列を削除します。監査データの保持の詳細については、 監査保持のセットアップ を参照してください。

監査保持のセットアップ

[保持] オプションを使用して、要件に応じて監査データの削除を自動化および簡素化します。

始める前に
必要なロール：security_admin

手順

1. 移動先 すべて > 監査管理コンソール.
2. リストから保持ポリシーを更新するテーブルを選択します。

   注: デフォルトでは、この手順の最後に [列] タブが表示されます。

3. [保持] タブを選択して、選択したテーブル監査データの保持ポリシーを更新します。
   保持オプションを示すモーダルが表示されます。
4. [監査レコードの自動消去] 切り替えを有効にします。
5. [期間] ドロップダウンメニューから期間を選択します。
   選択した期間を続行する場合は、[はい] を選択します。別の期間を選択する場合は、[キャンセル] を選択することもできます。

   注: 設定した期間よりも古いログは消去され、復元することはできません。

   選択した期間を確認する [保持ポリシー] モーダルが表示されます。
6. [保存] を選択して、選択したテーブルの保持ポリシーを更新します。
7. オプション: テーブルの監査レコードの削除を無効にするには、[監査レコードの自動消去] トグルを無効にして [保存] を選択します。

   注: 選択した保持期間のために以前に削除された監査レコードは、永久に使用できなくなります。

システム監査テーブルおよびリレーションシップ変更の監査テーブルの表示

ServiceNow AI Platform は、システム監査 (sys_audit) テーブルとリレーションシップ変更の監査 (sys_audit_relation) テーブルの挿入と更新を追跡します。

ServiceNow AI Platform は監査テーブルを追跡します。テーブルを追跡するには、ディクショナリーレコードの [監査] チェックボックスをオンにして、値を true に設定します。デフォルトでは、更新セットテーブルなどのシステムテーブルのレコードは監査されません。

注: パフォーマンスの問題や無限ループを防ぐために、システム監査テーブルへ挿入することでトリガーされるビジネスルールやワークフローはスキップされます。

システム監査テーブルの列

ナビゲーションフィルターに 「sys_audit.list 」と入力して、インスタンスのシステム監査テーブルにアクセスします。

sys_audit テーブルレコードに次の列が表示されます。

フィールド	説明
テーブル名	監査レコードの対象となるテーブル (「incident」など)
フィールド名	監査レコードの対象となるテーブルの列 (「assigned_to」など)

フィールド	説明
ドキュメントキー	監査レコードに関連付けられている元のレコードの Sys_id (一意のレコード識別子)。
ユーザー	変更を作成したユーザーの名前。 注: 一部の自動プロセスでは、システムまたはゲストユーザーを使用して、レコードへの変更を適用および追跡します。詳細については、「システムユーザーとゲストユーザー」を参照してください。
以前の値	この sys_audit レコードで表されるフィールド変更の以前の値。 • 参照フィールド：変更されたレコードの一意の sys_id 値。 • 日付/時刻のフィールド：データベースに格納されている協定世界時 (UTC) の値。
新しい値	この sys_audit レコードで表されるフィールド変更の新しい値。 • 参照フィールド：変更されたレコードの一意の sys_id 値。 • 日付/時刻のフィールド：データベースに格納されている協定世界時 (UTC) の値。

リレーションシップ変更の監査 (sys_audit_relation) テーブルの仕組み

システム監査 [sys_audit] テーブルは、監査対象としてフラグが付けられたテーブルの参照フィールドの変更を追跡します。このアクティビティには、ジャーナルフィールドエントリと履歴セットが含まれます。リレーションシップ変更の監査 [sys_audit_relation] テーブルは、sys_audit テーブルレコードと監査されるレコードの作成元であるソーステーブルとの関係の変更を追跡します。また、レコードが削除された可能性がある場合も追跡します。

•

- テーブル内のレコードを監査するたびに、データを記録するストアに対して、さまざまな元のテーブル間の関係が作成されます。この関係性の情報は、sys_history_set、sys_history_line、および sys_journal テーブルに保存されます。
- 監査対象テーブルレコードに関連するフィールドを削除すると、sys_audit_relation テーブルに削除が記録されます。つまり、監査済みレコードを変更すると、まず過去の要素を削除してから、sys_audit_relation テーブルに新しいドキュメント ID が作成されます。

履歴セットについて

ユーザーがレコードを作成するか、その履歴を表示すると、必要に応じて監査テーブルから履歴セットレコードが自動的に生成されます。

レコードが監査対象テーブルにある場合、レコードが挿入されたとき、またはユーザーがレコードを表示したときに、その履歴セットが生成されます。

注: 履歴セットを使用してレポートを生成しないでください。

いくつかの情報のフィールドが履歴セットレコードにキャプチャされ、リストビューに表示されます。

リストビューレコードフィールド

フィールド	入力値
ID	履歴が記録されているレコードのドキュメント ID。
テーブル	履歴が記録されているレコードの監査対象テーブル。
ロード時間	履歴セットの生成に要した時間。

監査履歴レコードフィールド

フィールド	入力値
ラベル	変更されたフィールドのラベル。
以前 (Old)	変更前の値。
新規	変更後の値。
タイプ	エントリが通常のフィールド用か、メールレコード用か、関係変更レコード用かを示します。
更新回数	このフィールドが変更された回数。-1 の値は、レコードが作成または削除された時点を示します。
更新時間	変更した日時。 **注:** 自動生成された履歴行の更新時間は、特定の処理状況でのレコードの作成時刻または更新時間と一致しません。レコードの履歴セットを初めて表示すると、履歴行のレコードの最初のセットが自動的に生成されます。アップグレードでのファイルの変更は監査されないため、この日付の不一致は次の場合に発生します。 • レコードに変更が加えられた後に履歴セットを表示した場合。 • ただし、将来のアップグレードで別の変更が行われる前。
ユーザー名	変更を作成したユーザーの名前。

カレンダービューの履歴セット

履歴セットがアクティブになると、履歴コンテキストメニューの選択肢に、sys_audit テーブルからの情報ではなく、履歴セットの情報が入力されます。ユーザーの側から捉えた場合、同じ履歴データを同じユーザーインターフェイスで使用できますが、情報の保存方法は異なります。

履歴ビューにはカレンダービューが含まれていますが、通常のリストインターフェイスを使用して履歴レコードのフィルタリングと操作は行われないため、次の操作は可能になります。

- 履歴データの検索とフィルタリング
- 履歴データのエクスポート

履歴セットの表示

履歴セットを表示するには 2 つの方法があり、コンテキストメニューアクション履歴からアクセスできます。

監査セットと履歴セットの違い

監査 [sys_audit]、履歴セット [sys_history_set]、および履歴 [sys_history_line] テーブルは同じデータを格納しますが、目的やデータの管理方法は異なります。

監査 **[sys_audit]** テーブル

監査 [sys_audit] テーブルには、すべてのレコードの履歴情報が格納されます。これらのレコードは、アドミニストレーターが監査済みレコードの履歴を常に追跡できるように、永久に保持されるようになっています。監査レコードの数は時間の経過とともに増加するため、履歴情報について監査テーブルを直接クエリーするのは非効率的です。実際に履歴情報を表示する必要がある、より少量のサブセットレコードでのみクエリーを実行する方がはるかに効率的です。

履歴セット **[sys_history_set]** テーブル

履歴セット [sys_history_set] テーブルは、監査対象テーブルのどのレコードに履歴情報があるかを識別します。履歴 [sys_history_line] テーブルには、発生したフィールド値に加えた実際の変更が保存されます。

- ユーザーがレコードを作成するか、その履歴をリクエストすると、必要に応じて、監査テーブルから履歴セットと履歴レコードが自動的に生成されます。
- 履歴セットと履歴レコードには、システム内のすべての変更の完全な履歴ではなく、ユーザーが履歴情報の作成やリクエストを行った、レコードの履歴情報の最近のサブセットのみが含まれます。
- 監査データに加えて、履歴セットには、レコードの挿入中に設定された情報 (ジャーナルフィールドエントリなど) も含まれます。レコードの作成前に作成したジャーナルフィールドエントリは、レコード作成時に作成されたジャーナルエントリと同じ方法で処理されます。これらのジャーナルエントリは、関連レコード自体と同じ作成時刻と作成者のデータで履歴セットに表示されます。

履歴セットと履歴レコードは次のように制限されます。

- テーブルクリーナーを使用して、30 日間更新されていない履歴セットレコードを削除できます。
- テーブルローテーションを使用して、4 つの履歴テーブルを 7 日ごとにローテーションします。28 日以上経過した履歴レコードは削除されます。

後日、履歴情報が再度必要になった場合は、監査ソースレコードから再生成することができます。

履歴セットレコードが生成された後は、コンテキストメニューの [履歴] で監査レコードではなく履歴セットが使用されます。ユーザーの側から捉えた場合、同じ履歴データを同じユーザーインターフェイスで使用できますが、情報の保存方法は異なります。

履歴へのアクセスを制御する

監査履歴を表示するためのアクセス権は、システムプロパティを設定することでロールに付与できます。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > システムプロパティ > システム.
2. テーブルから *glide.history.role* プロパティを選択します。
3. *List of roles (comma-separated) that can access the history of a record*で、履歴にアクセスするユーザーロールを選択します。
4. [保存] を選択します。

結果
読み取りアクセス権のないユーザーがレコードの履歴を表示する場合、フィールドへの変更は省略されます。

履歴エントリ数を変更する

デフォルトでは、履歴には最大 250 件の履歴エントリが表示されますが、この値を変更することができます。

始める前に
必要なロール：admin

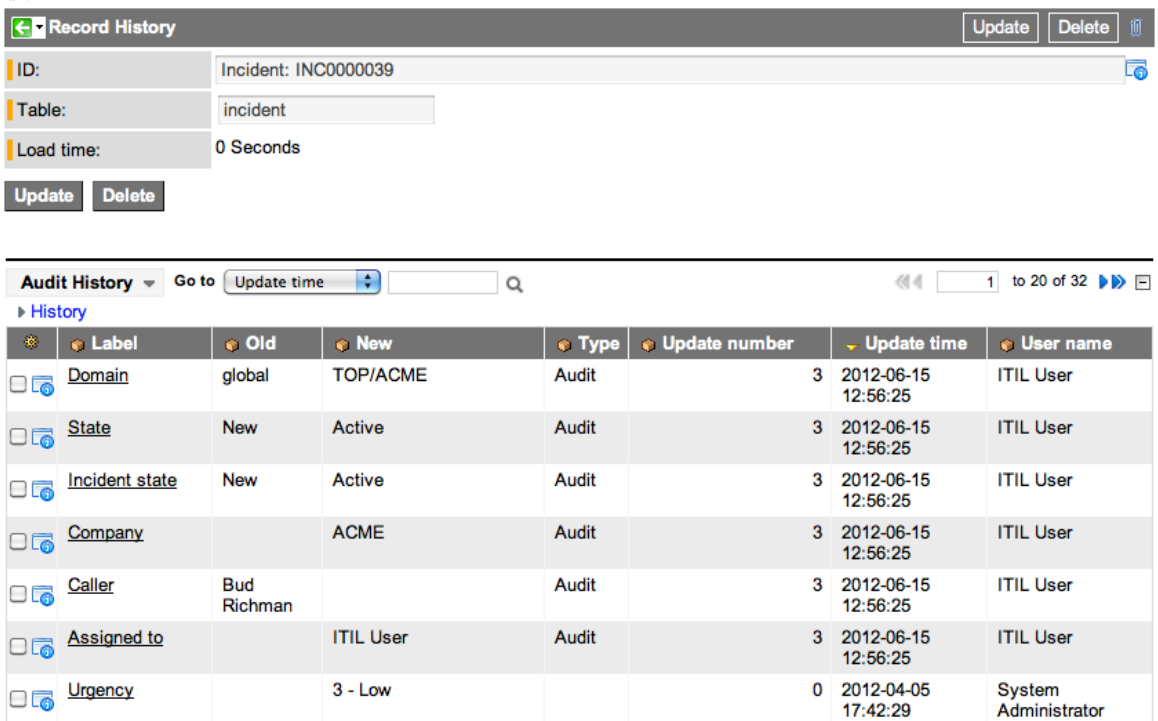
手順

1. 移動先 すべて > システムプロパティ > システム.
2. *glide.history.max_entries* プロパティを選択します。
3. *Maximum number of field entries displayed in record history, default is 250* 値を表示するエントリの新しい最大数を設定します。

履歴リスト

履歴リストには、各変更が変更リスト内の独自の行として表示されます。

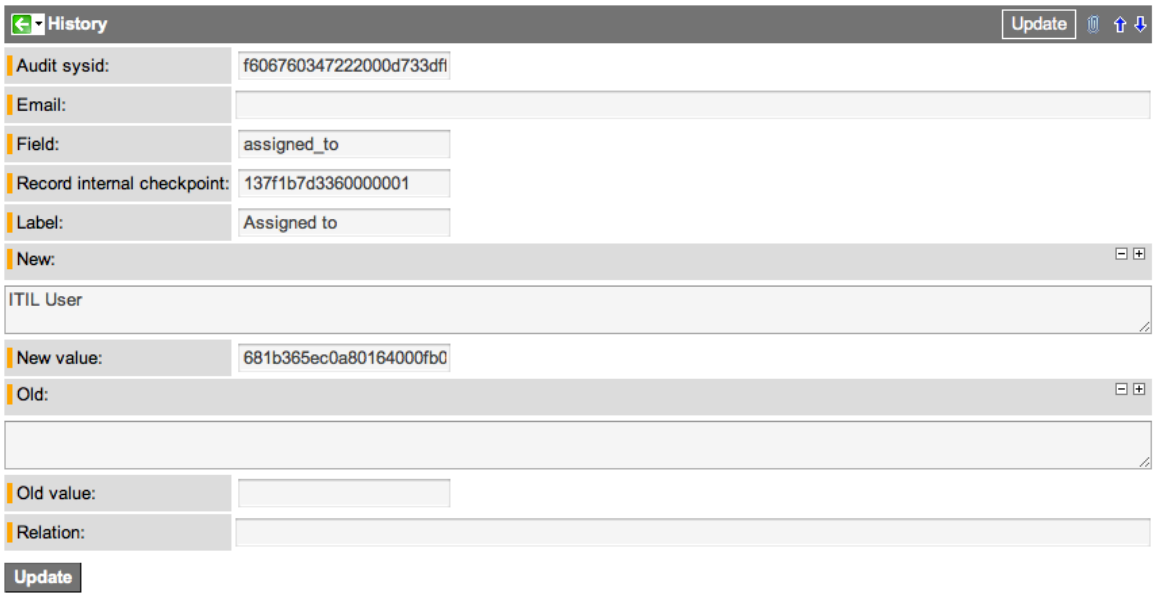
履歴リストを表示する



自動翻訳

行アイテムをクリックすると、変更に関する詳細が表示されます。

リストの変更を表示する



要件

履歴リストを表示するには、次の要件を満たす必要があります。

監査

履歴リストを表示するには、テーブルの監査を有効にする必要があります。

ACL

デフォルトでは、アドミンユーザーロールのユーザーのみがリスト履歴オプションを使用できます。アドミン以外のユーザーに対してこのオプションを有効にするには、カスタム ACL ルールを作成して、レコード履歴 [sys_history_set] テーブルへの読み取りアクセス権を付与します。

ロール

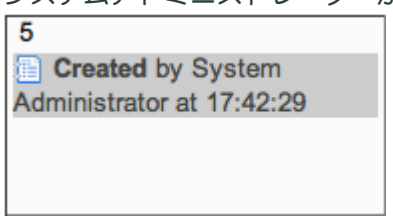
ユーザーが持つロールの少なくとも 1 つを *glide.history.role* プロパティに含める必要があります。このプロパティにはデフォルトで itil ロールが含まれます。

履歴カレンダー

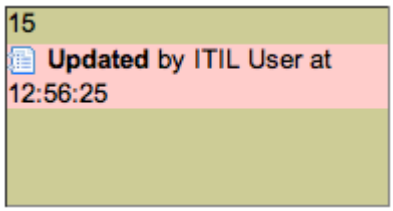
履歴カレンダーには、レコードの変更日、変更者、および変更時間が表示されます。

履歴カレンダーは更新番号でソートされます。各ユーザーには色が割り当てられているため、特定のユーザーがレコードを変更した回数を一目で把握できます。例：

システムアドミニストレーターが行った変更

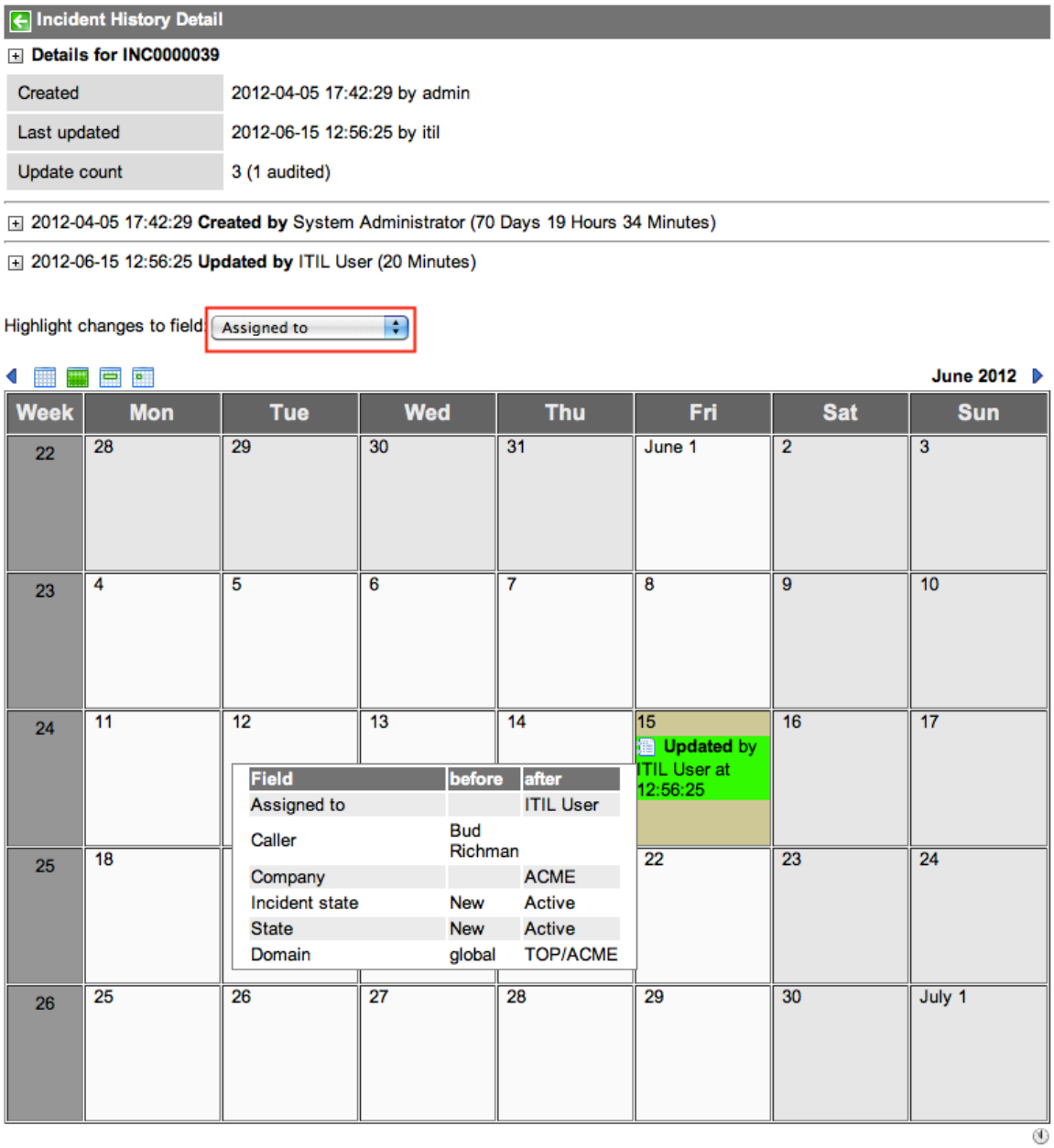


ITIL ユーザーが行った変更



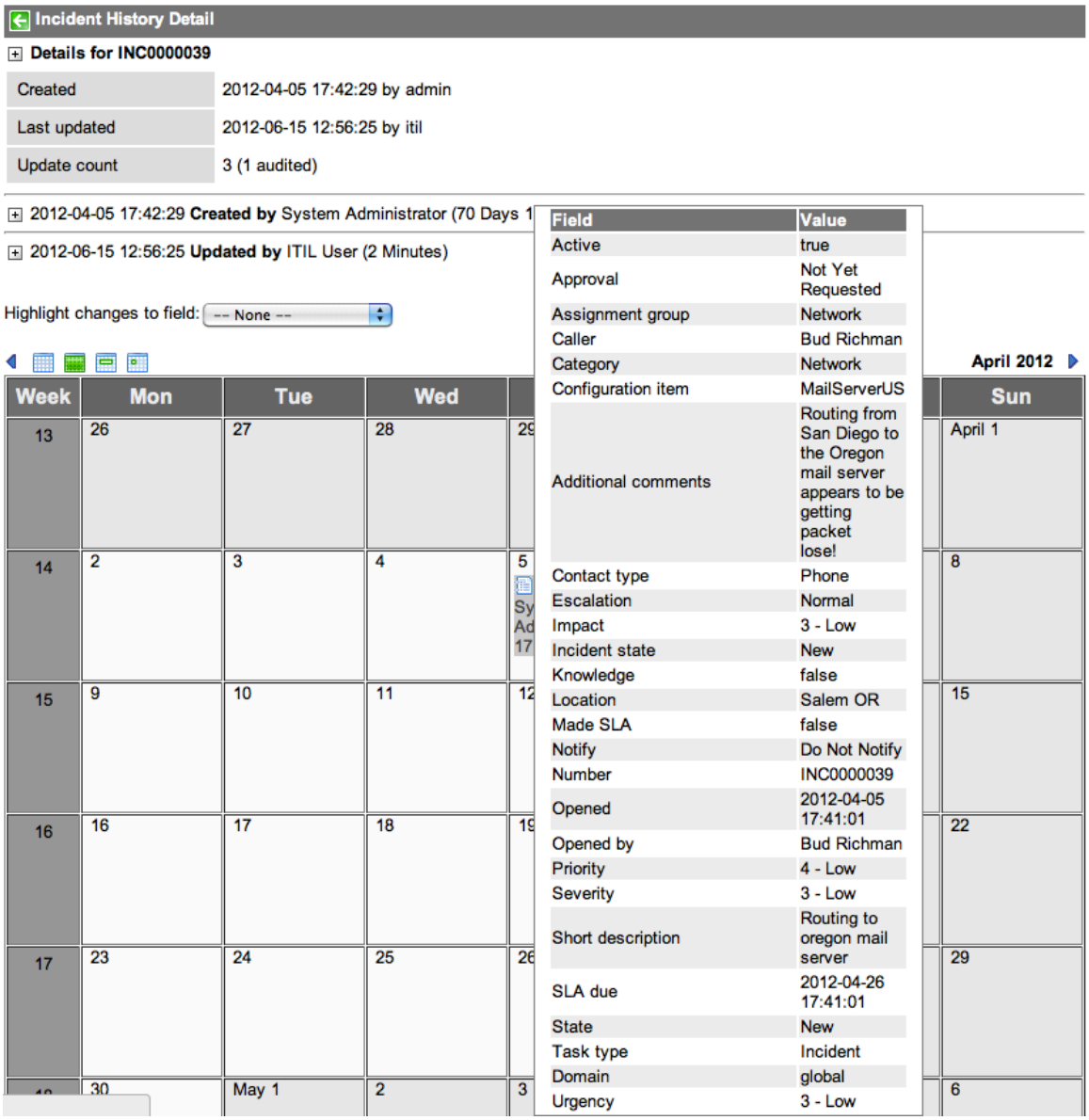
特定のフィールドの変更をハイライト表示するには、[フィールドへの変更をハイライト] 選択ボックスのフィールドを選択します。この選択ボックスのフィールドを選択すると、カレンダーが変更され、そのフィールドが変更された時間が強調表示されます。ハイライト表示された変更のいずれかのテキストにカーソルを合わせると、値の変更が表示されます。

ハイライト表示された変更を表示する



エントリ内のアイコンにカーソルを合わせると、すべての値の変更がポップアップに表示されます。これは、フォームの上部に表示される情報と同じです。

カレンダーの変更を表示する



日数をクリックすると、その日の変更が表示されます。左側の週番号をクリックして週次ビューを表示することもできます。月をスクロールして変更を確認できます。

履歴タイムライン

CI とその関連レコード、関係、ベースライン、および CI に対して提案された変更のタイムラインを表示できます。タイムラインは、構成アイテム [cmdb_ci] テーブルまたはこのテーブルの子孫 (テーブルの監査が有効になっている場合) の CI で使用できます。

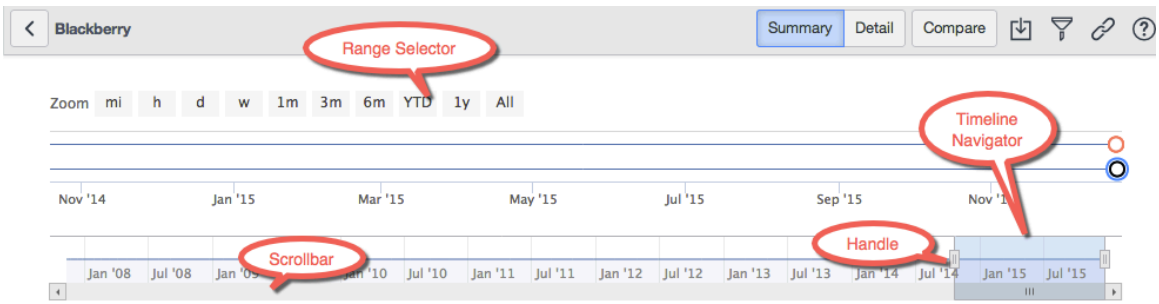
必要なロール：このビューの ACL は、*glide.history.role* システムプロパティで定義されたロールに基づいており、デフォルトでは itil に設定されています。また、ユーザーは、デフォルトでアドミニストレーターに付与されている履歴セット [sys_history_set] テーブルへの読み取りアクセス権も持っている必要があります。

CI の履歴を表示すると、タイムラインを開くことができます。タイムラインに表示される期間、時間範囲、およびプロパティを指定できます。特定の変更セットの変更内容を表示することも、CI 全体を表示して問題のトラブルシューティングを行うこともできます。CI の関連レコードに対する変更のタイムラインを表示し、任意の時点での CI のスナップショットをエクスポートして比較することもできます。

CI の変更は、タイムラインに沿ってさまざまな形や色のバブルで表されます。各バブルの形状はさまざまなタイプの変更を表し、各バブルの色は変更が有効か無効かを示します。CI のベースラインは、カーソルを合わせると詳細を表示できる黒い円で表されます。**[?]** アイコンをクリックしてバブルの形状と色の定義を表示し、バブルをポイントして変更セットの詳細を表示します。

関係への変更は、変更管理によって適用された場合にのみ有効と見なされます。提案された変更フレームワークを介して変更が適用された場合、その変更は有効です。追加の検証手順については、「予定されている検証スクリプトの作成または編集」を参照してください。

履歴タイムライン表示



タイムラインのバブル

注: 開始予定日がない提案された変更は、将来の時点に配置されます。

タイムラインナビゲーター

タイムラインナビゲーターの両端にあるハンドルを使用して、表示される期間を延長または短縮します。

タイムラインナビゲーターの下部をクリックし、ナビゲーターを左右にドラッグすると、別の期間にスクロールできます。

Zoom

デフォルトでは、先月のタイムラインが表示されます。タイムラインの上にある [ズーム] ラベルの横で、別の時間間隔を選択できます。間隔は 1 分から全期間まで選択できます。

期間中に多くの CI 変更があると、表示されるバブルが密集しすぎる可能性があります。次のいずれかの方法で、バブルを拡大または縮小できます。

- タイムラインの時間間隔を変更します。時間間隔を短くするとズームインし、間隔を長くするとズームアウトします。
- ズームインするタイムラインのセクションを選択します。

プロパティフィルター

表示されるバブルをフィルタリングできます。デフォルトでは、CI のすべてのプロパティの変更を表すバブルがすべて表示されます。選択したプロパティが変更されたバブルのみを表示し、選択されていないプロパティのみが変更されたバブルを除外するようにビューを制限できます。

[詳細] ビューと [サマリー] ビューでは、変更されたフィルタースコープ内のプロパティがハイライト表示されます。変更されたプロパティはライトブルーでハイライト表示されます。

[サマリー] ビューでは、CI のすべてのプロパティを含めるか、変更されたプロパティのみを含めるかを選択できます。[サマリー] ビューですべてのプロパティを表示する場合、変更されたプロパティは未変更のプロパティの前に表示されます。

サマリービュー

[サマリー] ビューには、各バブルで表される CI のスナップショットが表示されます。各スナップショットには、変更セットに従って CI のフィールドと関係への変更が表示されます。変更前後の以前の値と新しい値、および追加または削除された関係が表示されます。

スナップショット表示の両側にある **[>]** および **[<]** ボタンを使用して、次の変更セットレコードと前の変更セットレコードを時系列にスクロールします。

詳細ビュー

[詳細] ビューでは、バブルに対応する CI のスナップショットが表示されます。各スナップショットには、プロパティフィルターのスコープ内にあるフィールドが含まれており、変更されたプロパティがライトブルーの背景で表示されます。バブルをクリックすると、対応する CI のスナップショットが表示されます。表示されるデータは読み取り専用です。

両側の **[>]** と **[<]** ボタンを使用して、次や前の変更セットレコードを時系列にスクロールします。

関連レコードの変更のタイムラインを表示する

CI レコードの変更のタイムラインでは、CI の関連レコードの変更のタイムラインを表示することもできます。

始める前に

必要なロール：admin

- ターゲットテーブル：CI レコードは構成アイテム [cmdb_ci] テーブルまたはこのテーブルの子孫に存在している必要があります。
- 監査：CI を含むテーブルに対して有効にする必要があります。

手順

1. CI のタイムラインを開きます。
2. [関連レコード] アイコンを選択して、[関連レコードリスト] から表示する関連レコードを選択します。
   [関連レコード] アイコンを再度クリックすると、関連レコードのタイムラインが表示されます。

結果

CI の関連レコードに対する変更のタイムラインは、CI のタイムラインのすぐ上に表示されます。すべての関連レコードをオフにすると、関連レコードのタイムラインは非表示になります。

次のタスク

関連レコードタイムラインの変更バブルにカーソルを合わせると、変更されたプロパティの日付や数などの詳細が表示されます。フォーカスされる時間間隔を変更したり、ズームインまたはズームアウトすると、CI タイムラインと関連レコードのタイムラインの両方に同時に影響します。

構成アイテム (CI) のスナップショットのエクスポート

タイムラインから構成アイテムのスナップショットをエクスポートできます。

始める前に
構成アイテムは、構成アイテム [cmdb_ci] テーブルまたはその子孫に存在している必要があります。監査は CI を含むテーブルに対して有効にする必要があります。
必要なロール：admin

このタスクについて
CI のスナップショットは XML、PDF (ポータル)、または PDF (横向き) 形式でエクスポートすることができます。

手順

1. 移動先 すべて > 設定 > ベースアイテム > すべて をクリックして構成アイテムリストを開きます。
2. 構成アイテムレコードを開きます。
3. レコードのタイムラインを開きます。
4. CI のスナップショットをエクスポートする時間を表すバブルを選択します。
5. エクスポートアイコン ( ) をクリックします。
6. エクスポートに使用するファイル形式を選択します。
   ファイルをシステムにダウンロードして表示することができます。

CI スナップショットを比較する

タイムライン内の 2 つの異なる時点の CI のプロパティと関係を比較できます。

始める前に
CI が構成アイテム [cmdb_ci] テーブルまたはそのデーブルの子孫に存在している必要があります。監査は CI を含むテーブルに対して有効にする必要があります。
必要なロール：admin

手順

1. CI のタイムラインを開きます。
2. [比較] をクリックします。
3. 開始日と終了日を選択します。
4. [比較] をクリックします。

参照フィールドの変更の追跡

アドミニストレーターは、参照フィールドの表示値の変更を追跡できます。

参照フィールドには ID 値のみが格納されるため、通常は ID 値が変更された場合にのみ変更を監査できます。デフォルトでは、参照フィールドの表示値が変更されても、変更は監査されません。

次の状況を考えてみましょう。ユーザーが名前を「Jane Smith」から「Jane Miller」に変更します。ユーザー名はユーザーテーブルの表示値であるため、Jane Smith への以前の参照は Jane Miller を参照します。アドミニストレーターが既存のユーザーレコードの名前を更新したばかりの場合は、監査レコードと履歴レコードには新しい名前である Jane Miller のみが表示されます。デフォルトでは、元のユーザー名で行われた変更と新しいユーザー名で行われた変更を区別する方法は提供されていません。

監査ポリシーでユーザー名の変更を追跡するよう求められている場合は、次のことができます。

- 新しい名前の新しいユーザーレコードを作成し、以前のユーザーレコードを無効にします。以前のユーザー名の監査レコードは保持され、新しいユーザー名で将来の監査レコードが作成されます。
- 以前の名前と名前変更の日付を保存するカスタムフィールドとビジネスルールを作成します。この情報を使用して、監査レコードと履歴レコードで適切な名前が作成されます。

挿入の追跡

デフォルトでは、挿入が監査テーブルのサイズの 80% を超える可能性があるため、挿入の監査レコードは作成されません。

挿入を追跡しないと、パフォーマンスが向上し、監査テーブルが大幅に小さくなります。アドミニストレーターは、*glide.sys.audit_inserts* プロパティを *true* に設定することで、挿入の監査を有効にできます。

CI 関係の追跡

CI 関係 (CI 関係、CI/ユーザー関係、または CI/グループ関係) への変更は、変更が手動か ディスカバリー の結果であるかにかかわらず、変更された関係の両側のアイテムの履歴に表示されます。

たとえば、コンピューターアルファにコンピューターベータから使用されているという CI 関係がある場合、アルファの履歴にはベータとの関係がいつ確立されたかが記録され、同様に、ベータの履歴にはアルファとの関係がいつ確立されたかが記録されます。この例は、一部の CI 関係が確立され、その後、いずれかの関係が削除されたときに表示される履歴を示しています。

CI 関係履歴

⊟ 2008-12-03 10:49:37 **Updated by** Guest (19 Hours 54 minutes ago) - CI Relationship Change

- created 2 Days 2 Hours 51 minutes earlier

Relationship	Before	After
Runs	(relationship added)	Tomcat@tomdmac
Runs	(relationship added)	MySQL Server@tomdmac

⊞ 2008-12-03 10:49:38 **Updated by** Guest (19 Hours 54 minutes ago) - Mac OS X - Disks

⊞ 2008-12-03 10:49:43 **Updated by** Guest (19 Hours 54 minutes ago) - Mac OS X - Active Connections

⊞ 2008-12-03 10:50:02 **Updated by** Guest (19 Hours 53 minutes ago)

⊟ 2008-12-04 06:43:39 **Updated by** Glide Maintenance (just now) - CI Relationship Change

- last activity was 19 Hours 53 minutes earlier
- created 2 Days 22 Hours 45 minutes earlier

Relationship	Before	After
Runs	MySQL Server@tomdmac	(relationship removed)

作成された黒い丸は、CI、ユーザー、またはグループが作成された日付を示します。最後のアクティビティの黒い丸は、関係が最後に変更された日時を示します。いずれかまたはすべての CI 関係タイプの CI 関係履歴を表示しない場合は、CI 関係テーブル (CI 関係 [cmdb_rel_ci]、CI/ユーザー関係タイプ [cmdb_rel_user_type]、またはグループ関係 [cmdb_rel_group]) の監査を無効にしてオフにすることができます。

高セキュリティ設定

高セキュリティ設定とは、インスタンスで利用可能な複数のセキュリティオプションを指します。

高セキュリティ設定の詳細	高セキュリティ設定をアクティブ化
	
高セキュリティ設定の機能とビジネス価値について説明します。	高セキュリティ設定を有効にします。

高セキュリティ設定の詳細

高セキュリティ設定とは、インスタンスで利用可能な複数のセキュリティオプションを指します。

高セキュリティ設定モジュールは、High Security Settings プラグインで有効になり、新しいインスタンスではデフォルトで有効になります。インスタンスで高セキュリティ設定がアクティブでない場合は、「高セキュリティ設定のア有効化を要求する」を参照してください。このプラグインの詳細については、「インスタンスセキュリティ強化設定」の「高セキュリティプラグインを有効にする (Security Center 1.3 で更新)」を参照してください。次のタイプの高セキュリティ設定のプロパティを使用できます。

- デフォルトのプロパティ値：すべての重要なセキュリティ設定を 1 つの場所で管理および監査することで、プラットフォームのセキュリティを強化します。
- デフォルトの拒否プロパティ：テーブルアクセスのデフォルトのセキュリティ動作を制御するセキュリティマネージャープロパティを提供します。
- セキュリティアドミニストレーターロール：主要なセキュリティ設定とリソースの変更を防止するロールを提供します。セキュリティアドミンロールは admin ロールによって継承されないため、明示的にアサインする必要があります。
- 昇格された権限：security admin ロールを持つユーザーが、通常のユーザーのコンテキストで操作し、必要に応じてより高いセキュリティロールに昇格できるようにします。
- プロパティアクセス制御：セキュリティアドミニストレーターは、プロパティの読み取りと書き込みに必要なロールを設定できます。

- システムログ：読み取り専用です。
- アクセス制御ルール：ユーザーがアクセスできるデータとアクセス方法を制御します。

注:

- 高セキュリティ設定では、コンテキストセキュリティプラグインも自動的に有効になります (まだ有効になっていない場合)。さらに、プラットフォームセキュリティ設定 - 高は、インスタンスのセキュリティを強化するための設定と機能を提供します。
- [インスタンスセキュリティ強化設定] コンテンツには、ServiceNow AI Platform のセキュリティ関連のシステムプロパティとプラグインの詳細な説明とコンプライアンス値が含まれています。
- これらのプロパティの詳細については、「ハードニング設定」を参照してください。

高セキュリティ設定プロパティを設定または変更するには 2 つの方法があります。

- 移動先 システムセキュリティ > 高セキュリティ設定.

  [高セキュリティプロパティ] ページのオプションは [はい] または [いいえ] です。

- **sys_properties.list** に移動し、設定または変更するプロパティを検索します。

  システムプロパティテーブル [sys_properties.list] のオプションは **true** または **false** です。

プロパティアクセス制御

高セキュリティ設定がアクティブな場合、プロパティ [sys_properties] テーブルに 2 つの追加列が作成されます。

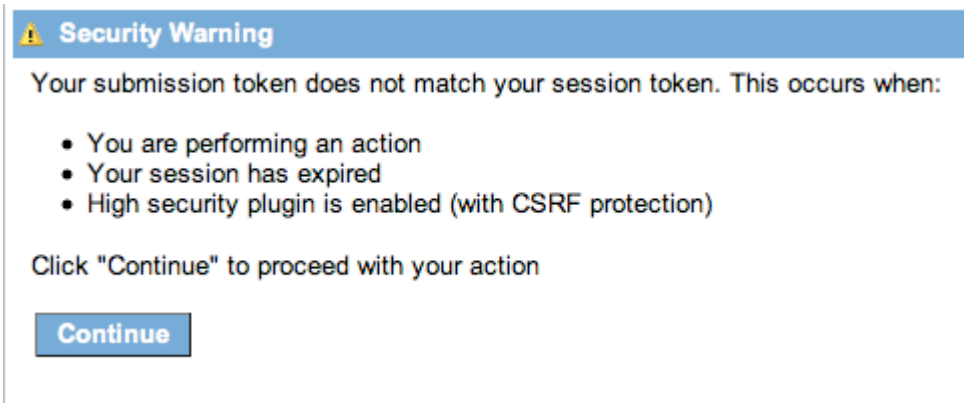
- **read_roles**：このプロパティのすべてのフィールドを読み込むことができるロール名のカンマ区切りリスト。
- **write_roles**：このプロパティのすべてのフィールドに対して書き込み/変更ができるロール名のカンマ区切りリスト。

プロパティテーブルにリストされているプロパティには、admin の **read_roles** と security_admin の **write_roles** があります。admin ロールを持つユーザーはプロパティ値を表示して読み取ることができますが、変更するには security_admin ロールに昇格する必要があります。

通知

高度なセキュリティ設定を有効にすると、セキュリティ警告メッセージも有効になります。承認後に表示されるメッセージの例を次に示します。

セキュリティ警告通知



高セキュリティ設定プロパティ

プロパティ	説明	デフォルト値	インスタンスセキュリティ強化設定
glide.ui.escape_text	ユーザーインターフェイスのパーサーレベルで XML 値をエスケープします。反映および格納されたクロスサイトの防止スクリプティング攻撃を防止します。このプロパティは、サービスポータルには適用されません。 注: Vancouver 以降のリリースでは、このプロパティはデフォルトで **true** に設定されており、アドミニストレーターが変更することはできません。プロパティを変更する必要があるユースケースについては、カスタマーサポートにお問い合わせください。	はい	XML マークアップをエスケープ (セキュリティセンター 1.3 での更新)
glide.ui.escape_all_script	デフォルトで Jelly JavaScript <![CDATA[<script type="text/javascript">]]> タグ内のすべての式を強制的にエスケープ処理します。<![CDATA[<script>]]> タグの type 属性が空の場合、または値が text/javascript、text/ecmascript、application/javascript、application/ecmascript、または application/x-javascript の場合にのみ、エスケープを適用します。	はい (新しいインスタンスの場合)	Jelly スクリプトをエスケープ (セキュリティセンター 1.3 および 1.5 で更新)

プロパティ	説明	デフォルト値	インスタンスセキュリティ強化設定
glide.ui.rotate_sessions	HTTP セッション識別子をローテーションしてセキュリティの脆弱性を低減します。参照：http://www.owasp.org/index.php/Session_Management#	はい 注: シングルサインオン認証に SAML 2.0 プラグインを使用している場合、このプロパティを [いいえ] に設定します。設定しない場合、インスタンスと ID プロバイダーの間で行われるセッション情報の共有が妨げられます。	HTTP セッション識別子をローテーションする ifiers
glide.ui.secure_cookies	有効化セキュアセッションの cookie：追加の cookie セキュリティを有効にします。「はい」の場合、厳格なセッション cookie 検証が適用されます。	はい	厳格なセッション Cookie のセキュリティを強制する (セキュリティセンター 1.3 で更新)
glide.security.password_reset.uri	モバイルの パスワードリセット では、ユーザーが [パスワードを忘れた場合] ボタンをクリックしたときに移動する URL を指定できます。		なし
glide.security.strict.updates	フォーム送信時に受信トランザクションのセキュリティを再確認します (フォーム生成時に常に権限が確認されます)。	はい	受信トランザクションを再確認する (Security Center 1.3 で更新)

プロパティ	説明	デフォルト値	インスタンスセキュリティ強化設定
	注: Vancouver 以降のリリースでは、このプロパティはデフォルトで **true** に設定されており、アドミニストレーターが変更することはできません。プロパティを変更する必要があるユースケースについては、カスタマーサポートにお問い合わせください。		
glide.security.strict.actions	実行前に UI アクションの条件をチェックします。通常、条件はフォームのレンダリング中にのみチェックされます。	はい	実行前の UI アクションの条件のチェック
glide.security.use_csrf_token	セキュアなトークンの使用を有効にして、受信要求を識別して検証します。このトークンはクロスサイトリクエストフォージェリ攻撃を防ぐために使用されます。	はい	Anti-CSRF トークンを有効にする (Security Center 1.3 の新機能、1.5 で更新、2.0 で削除)
glide.ui.escape_html_list_field	リストビュー内の HTML フィールドに対して HTML をエスケープします。	はい	リストビューでの HTML をエスケープ (セキュリティセンター 1.3 および 1.5 で更新)
glide.html.escape_script	HTML フィールドの javascript タグをエスケープします。	はい	JavaScript をエスケープ (Security Center 1.3 で更新)
glide.ui.forgetme	ログインページから [記憶する] チェックボックスを削除します。	はい	[記憶する] の削除
glide.smtp.auth	ユーザー名とパスワードのプロパティで SMTP サーバーを認証します。	はい	

プロパティ	説明	デフォルト値	インスタンスセキュリティ強化設定
	注: このプロパティは廃止されました。		
glide.soap.strict_security	着信 SOAP 要求に厳格なセキュリティを適用します。着信 SOAP 要求は、セキュリティマネージャーを経由してテーブルとフィールドにアクセスし、SOAP ユーザーに Web サービスを使用するための正しいロールがあるかをチェックする必要があります。	はい	SOAP 要求の厳格なセキュリティを強制する (Security Center 1.3 で更新)
glide.basicauth.required.wsdl	着信 WSDL 要求に認証を要求します。 注: 着信 WSDL 要求に認証を要求することを選択しない場合、アクセス制御 (ACL) ルールを変更してゲストユーザーが WSDL コンテンツにアクセスできるようにする必要があります。	はい	WSDL 要求に認証を必須とする (セキュリティセンター 1.3 および 1.5 で更新)
glide.basicauth.required.csv	着信 CSV 要求に基本認証を要求します 。	はい	csv 要求に認証を必須とする (Security Center 1.3 で更新)
glide.basicauth.required.excel	着信 Excel 要求に基本認証を要求します。	はい	Excel 要求に認証を必須とする (Security Center 1.3 で更新)
glide.basicauth.required.importprocessor	着信インポート要求に基本認証を要求します。	はい	インポート要求に認証を必須とする (Security Center 1.3 で更新)
glide.basicauth.required.pdf	着信 PDF 要求に基本認証を要求します。	はい	PDF 要求に認証を必須とする (セキュリティセンター 1.3 で更新)

自動翻訳

プロパティ	説明	デフォルト値	インスタンスセキュリティ強化設定
glide.basicauth.required.rss	着信 RSS 要求に基本認証を要求します。	はい	RSS 要求に認証を必須とする (Security Center 1.3 で更新)
glide.basicauth.required.scriptedprocessor	着信スクリプト要求に基本認証を要求します。	はい	スクリプト要求に認証を必須とする (Security Center 1.3 で更新)
glide.basicauth.required.soap	着信 SOAP 要求に基本認証を要求します。	はい	SOAP 要求に認証を必須とする (Security Center 1.3、1.5、および 2.0 で更新)
glide.basicauth.required.unl	着信アンロード要求に基本認証を要求します。	はい	アンロード要求に認証を必須とする (Security Center 1.3 で更新)
glide.basicauth.required.xml	着信 XML 要求に基本認証を要求します。	はい	XML 要求に対する認証を必須とする (セキュリティセンター 1.3 で更新)
glide.basicauth.required.xsd	着信 XSD 要求に基本認証を要求します。	はい	XSD 要求に認証を必須とする (Security Center 1.3 で更新)
glide.cms.catalog_uri_relative	/ess/catalog.do の URL パラメーターからの相対リンクを適用します。「はい」の場合、*uri* パラメーターを使用して、/ess/catalog.do ページから相対 URL のみが許可されます。「いいえ」の場合、すべての URL が許可され、外部の許可されていないコンテンツへのリンクが許可される可能性があります。	はい	相対リンクを強制する (セキュリティセンター 1.3 および 1.5 で更新)
glide.set_x_frame_options	このプロパティを有効にして、すべての UI ページの SAMEORIGIN に対して X-Frame-Options 応答ヘッダーを設定します。X-Frame-Options HTTP 応答ヘッダーは、<frame> または <iframe> 内の	はい	X-Frame-Options : SAMEORIGIN セキュリティヘッダーを実装 (セキュリティセンター 1.3 で更新)

プロパティ	説明	デフォルト値	インスタンスセキュリティ強化設定
	ページの表示をブラウザに許可するかどうかを示すために使用されます。サイトでは、このプロパティを使用して、このコンテンツがその他のサイトに組み込みできないようにすることで、クリックジャッキング攻撃を回避することができます。 https://developer.mozilla.org/en/the_x-frame-options_response_header		
glide.ui.attachment.download_mime_types	ブラウザにインラインで表示されない、カンマ区切りの添付 MIME タイプのリスト。クロスサイトの防止スクリプティング攻撃を防ぎます。たとえば、text/html は、HTML ファイルをブラウザでインライン表示するのではなく、添付ファイルとしてクライアントに強制的にダウンロードします。	text/html,image/svg,image/svg+xml	ダウンロード可能な MIME タイプを制限する (セキュリティセンター 1.3 および 2.0 で更新)
glide.security.groupby_acl_check	このプロパティが有効である場合、GroupBy 操作について、グループの実際のデータに基づきグループ名の ACL チェックが実行されます。	はい	なし
glide.security.diag_txns_admin	「はい」の場合、admin ユーザーまたは許可されている IP アドレスからのユーザーのみが stats.do、threads.do、および replication.do にアクセスできます。	いいえ	パフォーマンス監視のアクセスを制限する (Security Center 1.3 で更新)
glide.ui.security.codetag.allow_script	埋め込み HTML ([code] タグを使用) に JavaScript タグを含めることができます。	いいえ	埋め込み HTML コードを無効化する (セキュリティセンター 1.3 で更新)

自動翻訳

プロパティ	説明	デフォルト値	インスタンスセキュリティ強化設定
	注: Vancouver 以降のリリースでは、このプロパティはデフォルトで **true** に設定されており、アドミニストレーターが変更することはできません。プロパティを変更する必要があるユースケースについては、カスタマーサポートにお問い合わせください。		
glide.script.allow.ajaxevaluate	AJAXEvaluate プロセッサを有効にします。*AJAXEvaluate* API 呼び出しにより、クライアントはサーバー上で任意のスクリプトを送信して実行できます。	いいえ	AJAXEvaluate を無効にする

次のプロパティは sys_properties テーブルで定義されますが、[高セキュリティ設定] ページには表示されません。

プロパティ	説明	デフォルト値	インスタンスセキュリティ強化設定
com.glide.communications.httpclient.verify_hostname	リモート SSL ホストによって提示されたホスト名と証明書チェーンを確認します。中間者 (MITM) 攻撃から保護します。 詳細については、「Kubernetes Spoke を設定」を参照してください。 注: このプロパティは、com.glide.communications.trustmanager_trust_all プロパティを上書きします。	true	なし

プロパティ	説明	デフォルト値	インスタンスセキュリティ強化設定
glide.basicauth.required.schema	受信テーブルスキーマ要求に対してベーシック認証を要求します。	true	なし
glide.security.csrf_previous.allow	期限切れのセキュアトークンの使用を許可して、受信要求を識別して検証します。このトークンはクロスサイトリクエストフォージェリ攻撃を防ぐために使用されます。	false	なし
glide.security.csrf_previous.time_limit	保護トークンが期限切れになるまでの時間 (秒)。以前の CSRF トークンが有効である期間の長さを制御できます。ユーザーセッションが期限切れになると、*glide.security.csrf_previous.allow* プロパティが有効になっていて、このプロパティで指定された期間内にない限り、セキュアトークンも期限切れになります。このトークンはクロスサイトリクエストフォージェリ攻撃を防ぐために使用されます。	86400 **注:** 値は秒単位です。1 日に相当します。	なし
glide.security.csrf.strict.validation.mode	CSRF トークンの厳格な妥当性確認が実施されるため、CSRF トークンが一致しない場合、ユーザーは要求を再送信できません。	false	CSRF 検証をバイパスする警告をユーザーが受け入れるのを防ぐ (Security Center 1.3 および 1.5 で更新)
com.glide.security.check_unsanitized_html	クロスサイトスクリプティング攻撃に対処するために、グローバルレベルで translated_html フィールドのサニタイズ動作を強制します。	enforce	なし

高セキュリティ設定の有効化

High Security Settings プラグインは、すべての新しいインスタンスでデフォルトで有効です。インスタンスで有効にされていない場合は、プラグインを要求できます。

始める前に

必要なロール：なし

既存のインスタンスで高セキュリティ設定を有効にする前に次のことを実行します。

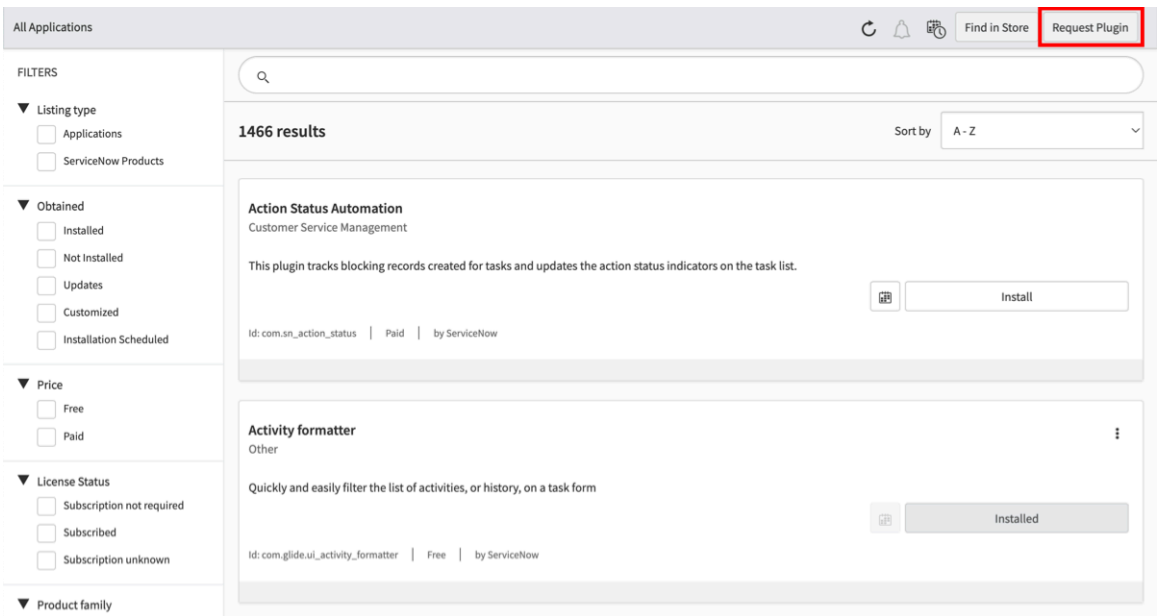
1. 新しい動作を理解するには、次の情報を確認してください。
   - アクセス制御リストのルール
   - 高セキュリティ設定
   - デフォルトの拒否プロパティ
2. 非本番インスタンスでプラグインを有効にします。最新の本番クローンが推奨されます。
3. 変更された機能、特に追加された ACL とデフォルト拒否機能をテストします。システムが期待どおりに動作するまでテストを続行します。ユーザーが予定されているリソースにアクセスできない場合は、アクセス権を付与するための適切なロールと ACL ルールがあることを確認してください。
4. 本番環境に適用できるように、必要な変更の更新セットを作成します。

> **注:** このプラグインの詳細については、「インスタンスセキュリティ強化設定」の「高セキュリティプラグインを有効にする (Security Center 1.3 で更新)」を参照してください。

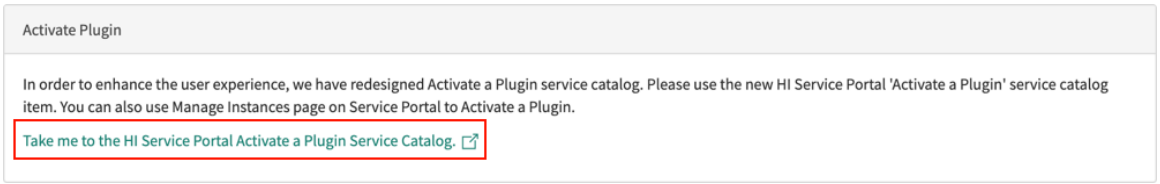
必要なロール：admin

手順

1. 移動先 すべて > システムアプリケーション > 利用可能なすべてのアプリケーション > すべて.
2. [すべてのアプリケーション] ページで [プラグインの要求] をクリックして、Now Support で [プラグインをアクティブ化] フォームを開きます。



3. Now Support で、Now Support サービスポータル サービスカタログ にアクセスするリンクを選択します。



4. インスタンスを選択します。
5. [アクション] > [プラグインのアクティブ化] を選択します。
6. [プラグインのアクティブ化] フォームで、次の情報を入力します。

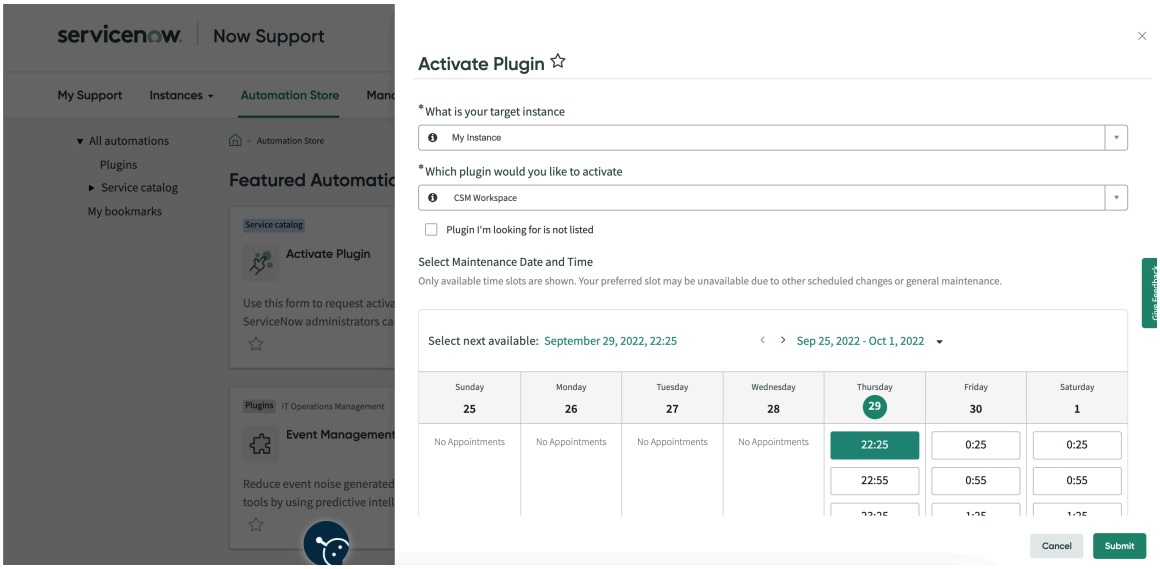
[プラグインのアクティブ化] フォーム

フィールド	説明
ターゲットインスタンスは何ですか	プラグインをアクティブ化するインスタンス。
どのプラグインをアクティブ化しますか	アクティブ化するプラグインの名前です。 注: 必要なプラグインが表示されない場合、または OEM またはオンプレミスのインスタンスでプラグインをアクティブ化している場合は、[探しているプラグインが表示されていません **(Plugin I'm looking for is not listed)]** チェックボックスをオンにして、プラグインの名前を入力します。
メンテナンスの日時を選択 (Select Maintenance Date and Time)	プラグインをアクティブ化する日時。 注: プラグインは、米国太平洋標準時で、毎営業日の朝と夕方の 2 回のバッチでアクティブ化されます。特定の時刻にプラグインをアクティブ化する必要がある場合は、[理由/コメント **(Reason/ Comments)]** フィールドに要求を入力します。

Example

たとえば、 [自分のインスタンス (My Instance)] という名前のインスタンスで CSM Workspace プラグインをアクティブ化するには、次のフォームを参照してください。

[プラグインのアクティブ化] フォーム



7. **[Submit** (送信)] を選択します。
   プラグインの要求の詳細については、次を参照してください。 のサービスカタログ [KB0751715] 記事からのプラグインの要求 Now Support ナレッジベース.

仮想プライベートネットワーク (VPN)

仮想プライベートネットワーク (VPN) を使用して、インターネット経由でインスタンスを外部データソースと統合します。

探索

仮想プライベートネットワークの機能とビジネス価値について説明します。

アクティブ化

仮想プライベートネットワークを有効にします。

構成

自動翻訳

仮想プライベートネットワークの構成方法について説明します。

仮想プライベートネットワーク (VPN) の詳細

仮想プライベートネットワーク (VPN) を使用して、インターネット経由でインスタンスを外部データソースと統合します。

自分の IP 情報

ライトウェイトディレクトリアクセスプロトコル (LDAP) や HTTPS などの暗号化されたプロトコルを使用する統合を設定する場合は、インターネットを転送メカニズムとして使用することをお勧めします。

ただし、データセンターとビジネスネットワーク間でサイト間インターネットプロトコルセキュリティ (IPSEC) 仮想プライベートネットワーク (VPN) 接続の使用を指示するセキュリティまたはネットワークアーキテクチャ要件がある場合があります。VPN は、インスタンスとネットワーク間で必要な暗号化通信をサポートします。

⚠ 警告:

VPN トンネルが開始されると、サイト間接続として動作します。これは、インフラストラクチャ上のエンドポイントが暗号化ドメインと呼ばれる IP アドレスを受け取ることを意味します。このパブリック IP には、同じデータセンター内の任意のインスタンスからアクセスできます。

たとえば、内部 Web サービスがあり、VPN トンネルを確立している場合、インスタンスは内部エンドポイントだけでなく、同じデータセンター内の他のすべてのインスタンスにも到達できます。

VPN 接続

ServiceNowVPN インフラストラクチャは、VPN の終了ポイントとして機能する Cisco 適応型セキュリティアプライアンス (ASA) デバイスのペアを使用します。

インスタンスとネットワーク間の VPN は、既存のネットワークハードウェアを使用して通信をサポートします。ハードウェアをインストールする必要はありません。構成は顧客ごとに異なるため、インスタンスには柔軟な VPN ソリューションが提供されています。インスタンスには、チェックポイント、Juniper、Nortel、およびその他の IPSEC VPN 対応デバイスへのトンネルがビルドされています。

インスタンスとネットワーク間の VPN 接続は、ネットワークへの暗号化されたトラフィックフローをサポートするために作成されます。多くの場合、VPN を使用する統合では、基礎となるプロトコルの一部として暗号化は行われません。たとえば、VPN 経由の LDAP とインターネット経由の LDAPS、VPN 経由の HTTP とインターネット経由の HTTPS などです。

ネットワークでは、受信から ServiceNow への統合またはエンドユーザーから ServiceNow へのトラフィックが VPN 接続を走査することは許可されていません。この制限付き通信には、プラットフォームへのエンドユーザーアクセス、プラットフォームの管理、Web サービスの統合、および MID サーバー を使用するように設定されたその他の統合が含まれます。インスタンスへのこのような受信通信はすべて、HTTPS を使用してインターネット経由で実行する必要があります。この設定では、暗号化された通信チャネルが提供されます。暗号化チャネルは、IP アクセス制御とともに、このトラフィックフローのセキュリティ要件を満たしています。

自動翻訳

VPN 通信用のアドレス

内部 ServiceNow ネットワークまたはネットワーク内の別の内部 IP アドレススキームとの競合または重複を防ぐために、暗号化ドメイン内のすべてのトンネルトラフィックは、トンネルの両側で RFC-1918 以外のアドレスを使用する必要があります。

ServiceNow は、ネットワークに対するクエリのソースの単一 IP アドレスを提供します。インスタンスと統合している各ホストのネットワークアドレス変換 (NAT)、非 RFC-1918 アドレスを指定する必要があります。このようなパブリックアドレスは組織が所有する必要があります。サードパーティのアドレスはトンネル内で使用できません。また、暗号化ドメインに VPN ピアの IP アドレスを含めることはできません。

冗長トンネル

次の 2 つの方法でトンネルの冗長性をビルドすることができます。

- 両方のピアの背後で同じ暗号化ドメインを使用する。これが推奨される方法です。
- 各ピアの背後で異なる暗号化ドメインを使用する。

最初の方法では、各ピアの背後に同じ NAT アドレスを指定し、そのアドレスを使用してサーバーへの接続パスを作成する必要があります。サーバーへのパスは、同じ物理マシンにすることも、同じサービスを提供するミラーにすることもできます。この方法では、プライマリトンネルとセカンダリトンネルのどちらがアクティブであるかに関係なく、インスタンスは同じ IP アドレスを使用してサーバーに接続します。複数のサーバーがある場合は、追加のサーバーでもこの同じスキームに従います。この方法はユーザーにとって最も透明性が高いため、推奨されます。

2 番目の方法では、インスタンスを構成して冗長性を持たせる必要があります。たとえば、LDAP にトンネルを使用する場合は、インスタンスに冗長化した LDAP サーバーをビルドすることができます。この方法では、インスタンスがセカンダリサーバーに接続しようとする前に、最初に設定された LDAP サーバーへの接続がタイムアウトする必要があります。このような追加の時間遅延があるため、最初のオプションが達成できない場合にのみこのソリューションを実装してください。また、インスタンスですべてのサービスに冗長性を提供できるわけではないためご注意ください。LDAP 以外のものに VPN トンネルを使用しているために冗長性が必要な場合は、複数のアドレスに対応できる構成であることを確認するか、上記の最初のオプションを参照してください。

VPN の使用の代替

この代替手段は、インスタンスを ServiceNow データセンターのリソースに接続するより簡単な方法とより優れた暗号化を提供します。また、VPN トンネルに問題がある場合にユーザーがインスタンスを使用できなくなるといった、VPN のダウンタイムによって発生する可能性のある問題を回避できます。

シングルサインオンと MID サーバー

VPN を使用して LDAP サーバーをインスタンスに接続するのではなく、認証用のシングルサインオン (SSO) とユーザーデータの同期用の MID サーバーを組み合わせて使用することを検討してください。LDAP 以外の統合の場合は、証明書ベースの暗号化を使用することを検討してください。

MID サーバーで LDAP リスナーを使用すると、ユーザーテーブルをほぼリアルタイムで同期できます。

このアプローチの利点は、ファイアウォールの穴、ルート、VPN トンネル、またはその他の特別なネットワーク設定を構成して維持する必要がないことです。SSO/MID サーバーソリューションは、完全な LDAP 統合を実現するための最も柔軟かつ安全でコスト効率の高い方法です。

LDAP over SSL

VPN トンネルを使用する別の方法として、インターネット経由で直接 LDAP Over SSL (LDAPS) を設定する方法があります。選択したインスタンスのソースアドレスと宛先ポートのみを使用して、読み取り専用のドメインコントローラーを構成し、DMZ 内のインスタンスをロックすることができます。LDAP のポートはインスタンスで設定できるため、必要に応じてポートアドレス変換 (PAT) を実行できます。LDAPS では、暗号化されたチャネルを介してインスタンスにアップロードされる証明書を制御します (「インスタンスへの証明書のアップロード」を参照)。証明書がないとパケットを暗号化または復号化できません。

このアプローチの利点は、より強力な暗号化と復号化のメカニズムが提供されることです。パスワードと同様に、VPN は調整された事前共有キーを使用して、インターネット上の 2 つのピア間のトラフィックを暗号化および復号化することができます。LDAPS は、アプリケーションレイヤーでエンドツーエンドの長い暗号化パスを提供し、IPSec トンネルが使用する事前共有キーよりもはるかに複雑な証明書を提供します。

VPN セットアップ

VPN 要求が送信されてから、通常、1 週間以内に VPN ビルドが完了します。インスタンスと組織の冗長性の要件に対応するために、2 ～ 4 つの VPN がプロビジョニングされます (アクティブサイトからアクティブサイトへ、またはアクティブサイトから DR サイトへなど)。

暗号化ドメインをできるだけ限定することをお勧めします。暗号化ドメインには、統合に必要な特定のホストのみを含めることが理想的です。大きな暗号化ドメインでは、ルーティングの不一致が発生する可能性があります (VPN とインターネット)。

VPN を作成するために、インスタンスでは次のことが行われます。

1. 各データセンターから VPN ピアとホストアドレスが提供されます。
2. 2 つのデータセンターからネットワークへの必要な VPN 接続がビルドされます。冗長性と災害復旧 (DR) の要件に対応するために、VPN を 2 つのデータセンターから 2 つのネットワークにプロビジョニングすることができます。

インスタンスでは、複数の地理的リージョンまたは子会社に接続する目的で、顧客のネットワークに複数の VPN トンネルをビルドすることはできません。複数の VPN トンネルを持つのではなく、独自の内部ネットワーク内でサイト間ルーティング、トラフィック分散、またはトラフィックシェーピングを実行する必要があります。

VPN サービスのアクティブ化

プロビジョニング、変更、または一般的な質問を含むすべての VPN 要求に サービスカタログ VPN 要求フォームを使用します。

始める前に
必要なロール：admin

手順

1. https://support.servicenow.com/now?draw=case に移動します。
2. [オートメーションストア] タブを選択します。
3. 左側のツリーを使用して移動先 すべての自動化 > サービスカタログ > クラウドインフラストラクチャ.
4. **[VPN** 要求] を選択します。
5. 適切な VPN 要求タイプを選択します。

6. 質問に回答します。
   質問は、選択した要求タイプによって変わります。
7. [送信] をクリックします。

結果
要求が送信されると、ServiceNow はネットワークエンジニアと協力して、VPN が正常にトラフィックを通過していることをテストし、検証します。質問に適切なタイミングで回答できるように、このプロセス中に VPN に関する質問に対処してください。

VPN 通信用のアドレスの構成

内部 ServiceNow ネットワークまたは別の顧客の内部 IP アドレススキームとの競合または重複を防ぐために、インスタンスでは、暗号化ドメインのすべてのトンネルトラフィックにおいてトンネルの両側で非 RFC 1918 アドレスを使用する必要があります。

始める前に
必要なロール：admin

このタスクについて
インスタンスは、ネットワークに対するクエリのソースの単一 IP アドレスを提供します。

手順

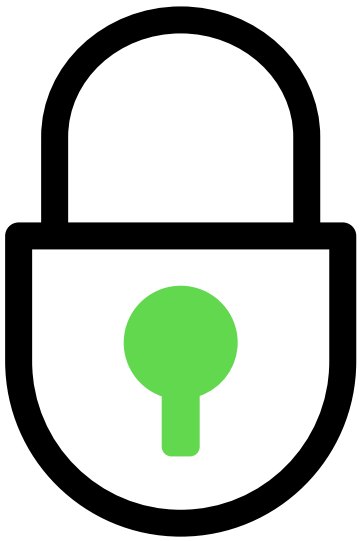
インスタンスと統合している各ホストのネットワークアドレス変換 (NAT)、非 RFC-1918 アドレスを指定します。

プラットフォームプライバシー

プライバシーを使用すると、インスタンスの機密データをマスクできます。

ストアアプリケーション

データプライバシー



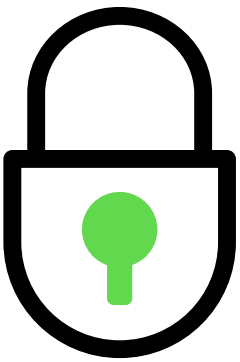
データプライバシーを使用して、機密データを分類し、本番インスタンスのユーザーデータから個人識別可能情報 (PII) を削除して、非本番インスタンスのデータを匿名化します。

データディスカバリー

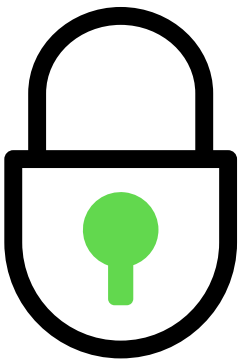
データディスカバリーを使用してインスタンス内の機密データを識別し、分類、保護、またはレポートします。

プラグイン

データプライバシー (従来)

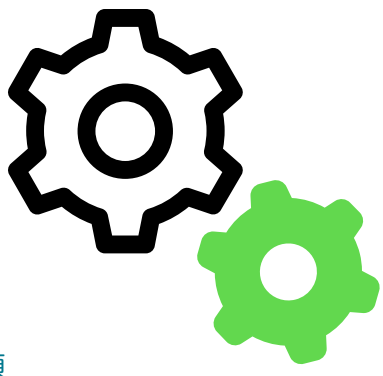


データの匿名化



データの匿名化は、データプライバシー規制に厳格に準拠する

データ分類



事前定義またはユーザー定義のデータ分類を使用して、データ

自動翻訳

従来のデータプライバシー (従来) プラグインを使用します。	ように、データを簡単に変換して識別不可能にする方法を提供します。	をタイプ別にグループ化します。ユーザーがデータ分類アドミニストレーターまたは監査人ロールを持っている場合は、さまざまなデータクラスを管理したり、インスタンス内のさまざまなタイプのデータの現在のステータスを視覚的に分析したりすることができます。

データプライバシーの概要

Data Privacy を使用して、機密データを分類し、本番インスタンスのユーザーデータから個人識別可能情報 (PII) を削除して、非本番インスタンスのデータを匿名化します。匿名化されると、ユーザーデータは規制対象の個人情報と見なされなくなります。

データプライバシーの重要性と、分類および匿名化によって機密情報を保護するために企業が実行できる手順について説明します。データプライバシーストアアプリケーションを使用して、メインダッシュボードと機能の概要を説明します。

開発者は、非本番インスタンスのデータを操作して、実装が期待どおりに機能していることを確認する必要があります。本番インスタンスからのデータのインポートは、本番環境をシミュレートするための便利な方法ですが、セキュリティ上のリスクがあります。データプライバシーを使用すれば、アドミニストレーターは非本番環境で安全に作業するための、個人情報を含まないデータを開発者に提供できます。

データ分類

インスタンス内のデータタイプの機密性レベルによって決定される事前定義された基準に従って、機密データを識別して分類します。データの機密性レベルは、分類された各データタイプの処理方法を決定するのに役立ちます。基本レベルのデータプライバシーを備えた事前定義されたクラスがいくつか提供されています。データプライバシーの分類セクションを使用して、インスタンス内のデータにラベルを付けてグループ化します。クラスを追加し、データクラス構造を表示して、データを分類します。事前定義またはユーザー定義のデータ分類を使用して、データをタイプ別にグループ化します。

データプライバシーストアアプリケーションを使用してデータを分類する方法について説明します。

ユーザーデータの匿名化

アドミニストレーターは、すべてのユーザーまたは一部のユーザーの情報をすべて匿名化するかどうかを定義します。匿名化すると、選択したユーザーレコードのデータが、ランダムな値または定義した値に置き換えられます。値を置き換える際は、さまざまな手法でデータ構造を保持できます。

データプライバシーストアアプリケーションを使用してデータを匿名化する方法について説明します。

データプライバシー

データプライバシーを使用して、機密データを分類し、本番インスタンスのユーザーデータから個人識別可能情報 (PII) を削除して、非本番インスタンスのデータを匿名化します。

注: 従来のプラグイン Data Privacy (Classic) を使用することもできます。

自動翻訳

インストールの詳細

インスタンスに次のアプリケーションがインストールされている必要があります。

- データプライバシー (従来) [com.glide.data_privacy]
- データプライバシー [sn_dp_store_app]
- データディスカバリー [sn_data_discovery]
- データディスカバリー API [com.glide.data_discovery]

インストールの仕組みは次のとおりです。

- データプライバシーストアアプリをインストールすると、データディスカバリーストアアプリ、データプライバシー (従来) プラグイン、およびデータ分類プラグインが自動的にインストールされます。
- データディスカバリーストアアプリをインストールすると、データディスカバリー API プラグインが自動的にインストールされます

考慮事項

- 匿名化できるのは、分類されたデータのみです。データのクラスと分類の詳細については、「データ分類」(Classic) または データ分類 ストアアプリを参照してください。
- ログ内の PII およびその他の監査データは匿名化されません。
- 匿名化できるのは、構造化されたデータのみです。ジャーナルフィールド、コメント、添付ファイル、および部分的なテキストが PII を表す可能性があるその他のフィールドなどの非構造化データは匿名化されません。詳細については、「匿名化でサポートされているフィールドタイプ」を参照してください。
- シングルサインオン (SSO) システムとの統合により、信頼できる情報源システムからのユーザー情報を再同期できます。sys_user データの匿名化の永続性を保証するメカニズムはありません。ユーザー管理および sys_users の詳細については、「ユーザー管理 」を参照してください。

データプライバシー

データプライバシーを使用して、機密データを分類し、本番インスタンスのユーザーデータから個人識別可能情報 (PII) を削除して、非本番インスタンスのデータを匿名化します。

データプライバシーの詳細

データプライバシーについて学習します。

データプライバシーの構成

データプライバシーの構成方法について説明します。

データプライバシーのロール

データプライバシーのロールの詳細を取得します。

データプライバシーの高度な機能

データプライバシーの高度な機能について説明します。

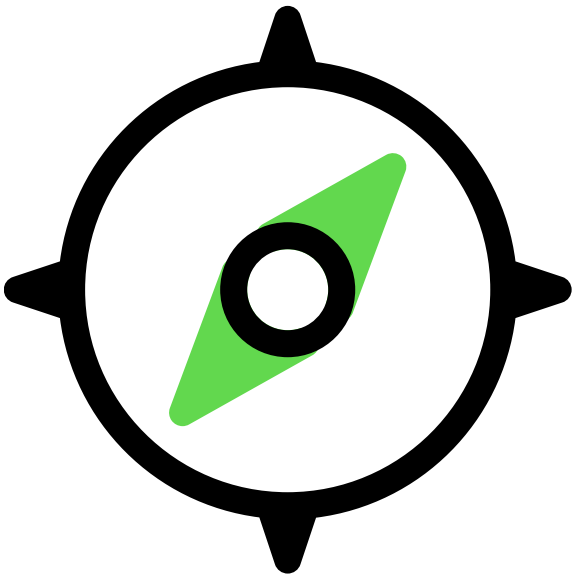
自動翻訳

Now Assist のデータプライバシー

生成 AI プロンプトから機密データを検出して匿名化する方法を設定および構成します。

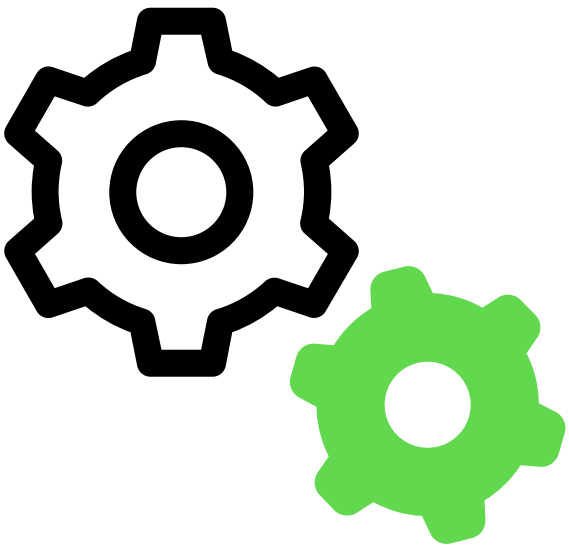
開始するには

探索



Now Assist のデータプライバシーの詳細

構成



Now Assist のデータプライバシーの構成

注: Now Assist のデータプライバシーは、正規表現パターンに基づいて機密データを検出してマスクし、コンテキスト (モデルタイプ) データパターンはサポートしていません。

Now Assist のデータプライバシーの詳細

データプライバシー for Now Assist で機密データの保護がどのように強化されるかについて詳しく説明します。

Now Assist の データプライバシー の概要

年齢、電話番号、その他の個人識別可能情報 (PII) などの機密データは、生成 AI プロンプトから処理されないようにマスクできます。プレースホルダーテキストと匿名化データは代わりにプロンプトとともに送信され、大規模言語学習モジュール (LLM) 応答の受信後にこれらの値が元のテキストに置き換えられます。この双方向マスキングにより、エンドユーザーは正確な応答を受け取ることができますが、機密データが LLM に公開されることはありません。

Now Assistのデータプライバシーを構成する際には、いくつかの考慮事項があります。構成詳細については、「Now Assist 向けデータプライバシーの設定」を参照してください。

重要:

データプライバシー for Now Assist は正規表現パターンに基づいて機密データを検出してマスクし、コンテキスト (モデルタイプ) データパターンをサポートしていません

自動翻訳

Now Assist 向けデータプライバシーの設定

生成 AI アプリケーションで個人識別可能情報 (PII) を匿名化するようにデータプライバシーの詳細構成を構成します。

始める前に

インスタンスに次のアプリケーションがインストールされている必要があります。

- データプライバシー (クラシック版)[com.glide.data_privacy]
- データプライバシー [sn_dp_store_app]
- データディスカバリー [sn_data_discovery]
- データディスカバリー API [com.glide.data_discovery]

最新バージョンの生成 AI コントローラーをインストールすると、データプライバシーストアアプリ (sn_dp_store_app) が自動的にインストールされます。データプライバシーストアアプリは、データディスカバリーストアアプリ [sn_data_discovery]、データプライバシー (従来) (com.glide.data_privacy) プラグイン、および データディスカバリー API [com.glide.data_discovery] を自動インストールします。

必要なロール:now_assist_data_privacy_admin

重要: データプライバシー for Now Assist を設定するために、フル アクティブ ライセンスは必要ありません。

手順

1. 移動先 すべて > データプライバシー (従来) > プライバシーポリシーの詳細構成.
   以前に機密データハンドラーを使用して生成 AI のデータを匿名化したことがある場合は、既にプライバシーポリシーが構成されている可能性があります。以前に設定した正規表現は、アップグレードの一部として移行されました。Now Assistのデータポリシーが既にある場合は、手順 6 に進みます。
2. **[New** (新規)] を選択します。
3. プライバシーポリシーの名前を入力します。
4. [データチャネル] フィールドで、使用するデータチャネルを選択します。

チャネル	説明
データキット	AI モデルが評価に使用しているデータは、機密データを検出して匿名化することでサニタイズされます
データ抽出	データはモデルトレーニング用に送信される前にサニタイズされる
Now Assist	データは GenAI コントローラーに送信される前にサニタイズされます

5. [アクティブ] を `true` に設定し、[送信] を選択してポリシーの詳細設定を作成します。
   データチャネルごとに一度にアクティブにできるポリシー構成は 1 つだけです。新しいポリシー詳細構成を有効にするには、そのデータチャネルの他のすべてのポリシー構成で [アクティブ] を [false] に設定する必要があります。
6. ポリシー詳細構成のリストにリダイレクトされたら、作成したレコードを開きます。
   既存のレコードを Now Assist データチャネルで開きます (すでに存在する場合)。
7. 匿名化するデータパターンを追加するには、[データパターンの選択] を選択します。

8. オプション: 独自のデータパターンを作成するには、「 データディスカバリー パターンの構成」を参照してください。
9. データ パターンを選択し、 [保存] を選択します。

結果

データパターンで選択された正規表現によってキャッチされたデータは、生成 AI アプリケーションで匿名化されます。このポリシー例では、一連のアクティブなデータパターンをキャッチしてから、データ抽出データチャネルを使用するように構成されています。

データプライバシー

データプライバシーストアアプリは、最新の外観と操作性を備えたデータ分類とデータプライバシーのための Next Experience 更新です。Data Privacy ストアアプリは、Utah 以降でサポートされています。

データプライバシーは、概要、分類、および匿名化という複数のコンポーネントで構成されています。

概要

[概要] セクションを開始点として使用して、データとデータプライバシーのコンプライアンスを管理します。詳細については、「データプライバシーの概要」を参照してください。

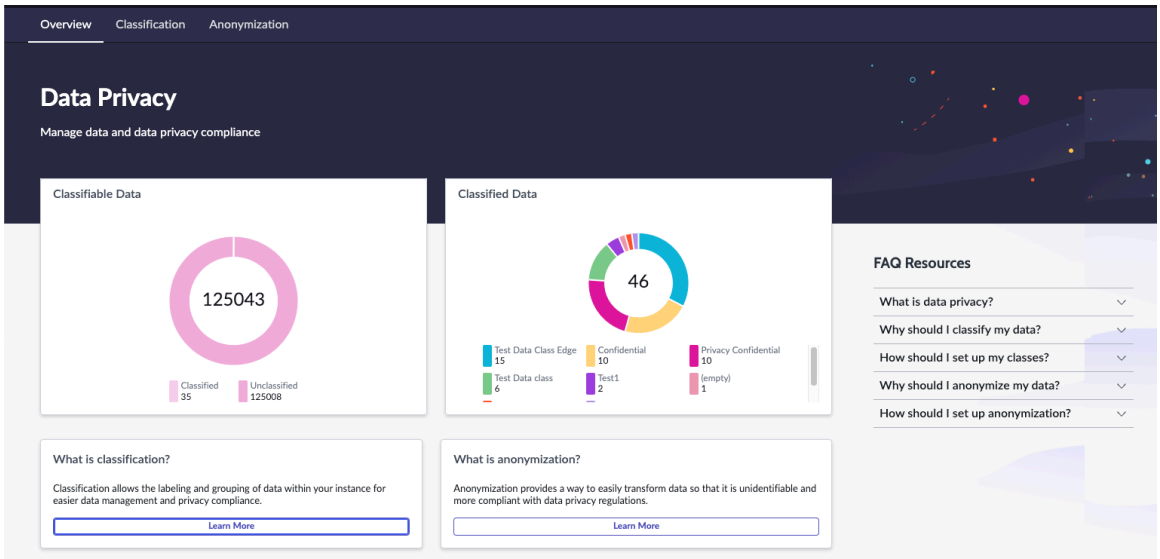
分類

データ分類は、データをカテゴリに整理して、後で使用するときに簡単に取得、ソート、および保存できるようにするプロセスです。分類システムを使用すると、機密性およびセキュリティポリシーの要件に焦点を当てることができます。データを分類して匿名化に使用できるようにします。データクラスと分類の詳細については、「データ分類」を参照してください。

データプライバシーの概要

[概要] ホームページは、データとデータプライバシーのコンプライアンスを管理するための開始点です。

概要ダッシュボードは、インスタンス内のデータ分類ジョブおよび匿名化ジョブの現在のステータスをレポートします。概要ダッシュボードでデータを表示し、分類可能なデータの量、利用可能なデータクラス、各クラスのデータ量、および匿名化ジョブの全体的なステータスを確認できます。分類されたデータはさまざまなカテゴリに分類されます。いずれかのチャートでサブカテゴリを選択すると、チャート全体でカテゴリが追加または削除され、数が調整されます。



分類可能なデータ (Classifiable Data)

インスタンス内の分類可能なデータレコードの合計数を表示します。この数は、分類されたデータの合計数と未分類のデータの合計に分類されます。これにより、アドミニストレーターは使用可能な分類の機会をすばやく把握できます。

分類済みデータ (Classified Data)

インスタンス内の分類済みのデータレコードの合計数を表示します。 この数は、割り当てられた各データクラスの分類済みデータの合計数に分類されます。これにより、各エリアで分類されたデータの量をすばやく把握できます。

匿名化ジョブ

ユーザーおよびデータクラスジョブについて、非本番インスタンスで完了したジョブの現在処理中であるデータプライバシージョブを表示します。

詳細

[詳細] セクションでは、分類と匿名化を簡単に表示できるほか、プロセスを簡単に開始できます。

FAQ リソース (FAQ Resources)

データプライバシーを使用してインスタンスを保護する方法に関する追加情報については、分類と匿名化に関する製品学習リソースにアクセスしてください。

データ分類

事前定義またはユーザー定義のデータ分類を使用して、データをタイプ別にグループ化します。ユーザーがデータ分類アドミニストレーターまたは監査人ロールを持っている場合は、さまざまなデータクラスを管理したり、インスタンス内のさまざまなタイプのデータの現在のステータスを視覚的に分析したりすることができます。

データ分類を使用して次のサポートを有効にできます。

- ServiceNow AI Platform インスタンスでホストされているデータのタイプの可視化。
- プライバシー法を遵守し、金融サービスや医療機器の製造など、業界の規制要件を満たします。

データ分類

データ分類は、任意のテーブルの既存のディクショナリーエントリにデータ分類を手動で適用するスタンドアロンプロセスです。詳細については、「データディクショナリテーブル」を参照してください。

- ビジネスに適したデータ分類を行い、必要に応じて利用可能なデータクラスを変更できます。
- データを分類するときは、事前定義されたデータ分類を使用することも、独自に作成することもできます。事前定義されたデータ分類の使用はオプションですが、開始点として使用することをお勧めします。これらの事前定義されたデータ分類は、インスタンスにインストールできるデモデータに含まれています。
- 独自のデータ分類を作成する場合は、親と子のデータ分類を使用して階層型の階層システムを設計することもできます。

注: データ分類 はドメインセパレーションをサポートしており、data_classification テーブル自体がプロセス分離されています。詳細については、「ドメインセパレーションとデータ分類」を参照してください。

ユースケース

一般データ保護規則 (GDPR) は、個人が自分の個人データを制御できるようにすることを目的とした欧州連合の規制です。個人を特定できる情報などのデータ分類を使用して、個人データがインスタンス内のどこに保存されているかを特定できます。適切なセキュリティメカニズムを適用して個人データの漏洩を防ぐことで、組織は GDPR 要件を満たすことができます。

顧客情報を ServiceNow AI Platform に保存する場合は、現地のプライバシー法の規制の対象となるデータを追跡するために必要な個人情報 (PII) 分類コードを使用します。

制限付きデータ分類は、社会保障番号 (SSN) などの従業員の機密情報を格納する従業員テーブル列に適用できます。アドミニストレーターと監査人は、概要ダッシュボードを使用して、データ分類が正しい列に割り当てられていることを確認できます。また、制限されたタイプの情報の分類の詳細を表示することもできます。

データ分類を作成する

[data_classification] テーブルに独自のユーザー定義のデータ分類を作成し、特定のテーブルの特定の列に割り当てることができます。新しいデータクラスを作成して、分類プロセスを開始します。

始める前に

必要なロール：data_classification_admin、admin

手順

1. 移動先 すべて > システムセキュリティ > データプライバシー > 分類.
2. [データクラスを追加 **(+Add data class)**] を選択します。

   注: ベースシステムには複数のデータクラスが含まれています。

3. フォームのフィールドに入力します。

フィールド	説明
クラス名	データ分類名。

フィールド	説明
親クラス	このデータ分類が従属している親データ分類の名前。このデータ分類が親データ分類の子ではない場合は、フィールドを空のままにします。
説明	データ分類の説明。

4. [送信] をクリックします。
   新しいデータクラスが追加されます。データクラスが子の場合、左側のナビゲーションバーの親の下に表示されます。

分類済みデータ

事前定義またはユーザー定義のデータ分類を使用して、データをタイプ別にグループ化します。辞書 [sys_dictionary] テーブルの特定のテーブル列にデータ分類を割り当てます。データ分類を割り当てると、辞書データクラス [m2m_dictionary_dataclass] テーブルにエントリが作成されます。作成されたエントリは概要ダッシュボードで確認できます。

始める前に

必要なロール：data_classification_admin、admin

手順

1. [新規] を選択し、分類するテーブルからデータを割り当てます。
2. ドロップダウンからデータクラスを選択します。
3. 分類するテーブルと列を反映するレコードを選択します。
   特定のテーブルを簡単に見つけることができるように、テーブルのページごとに表示する追加の行を選択します。
4. [データを分類] を選択します。
   データは、[分類済みデータ] テーブルで [分類名] で分類されます。
5. データを表示するか、Excel、CSV、JSON、または PDF にエクスポートします。
   選択した形式でデータをダウンロードするか、メールで受信するかを選択します。

データの匿名化

匿名化は、データプライバシー規制に厳格に準拠するように、データを簡単に変換して識別不可能にする方法を提供します。

アドミニストレーターは、本番インスタンスと非本番インスタンスの両方で匿名化プロセスを実行できます。データプライバシーを本番インスタンスで使用して、ユーザーを匿名化できます。データクラスの匿名化は、非本番インスタンスでのみ使用する必要があります。データ構造を保持すると、メールアドレスや物理アドレスなどのデータが、同様の形式で匿名化されたバージョンに置き換えられます。

アドミニストレーターは、一般データ保護規則 (GDPR) の忘れられる権利 (RTBF) プロセスの一部として、匿名化を使用してユーザー情報を匿名化することもできます。詳細については、「https://gdpr-info.eu/art-17-gdpr/」を参照してください。

データプライバシーの匿名化セクションを使用して、プライバシーポリシーと手法を作成および表示し、プライバシーの一括割り当てを実行します。すべてのジョブと、その説明、使用されたプライベートポリシー、およびステータスを表示します。匿名化セクションにアクセスするには、admin は最初に data_privacy_admin および data_privacy_processor ロールに昇格される必要があります。

匿名化手法

匿名化手法は、データの匿名化方法を決定するために選択するオプションです。匿名化ジョブで参照する匿名化手法を作成する必要があります。プライバシー手法を関連する匿名化手法構成に関連付けるには、「匿名化手法の作成」を参照してください。

匿名化ポリシー

匿名化ポリシーを構成して、データを匿名化するときに使用するデータプライバシー手法を指定します。詳細については、「匿名化ポリシーの作成」を参照してください。

匿名化ジョブ

匿名化ジョブは、これらすべてのコンポーネントを使用してデータを匿名化します。これらのジョブの詳細については、「匿名化ジョブの作成」を参照してください。

匿名化手法の作成

Data Privacy 手法構成を作成して、Data Privacy でデータを匿名化する方法をカスタマイズします。

始める前に
必要なロール：data_privacy_admin および admin

手順

1. **data_privacy_admin** ロールに昇格させます。
   ロールの昇格の詳細については、「特権ロールへの昇格」を参照してください。
2. 移動先 システムセキュリティ > データプライバシー > 匿名化.
3. [手法を表示 **(View techniques)]** を選択します。
   いくつかの事前定義された手法を選択できます。

手法	説明
選択的置換 (Selective Replace)	この手法では、文字列データを選択的に置換します。入力の開始インデックスと終了インデックスの間のすべての文字が、選択した文字に置き換えられます。マスクから除外する文字を指定できます。 ○ start_index：手法は指定された文字以降のデータをマスクします。空白のままにすると、最初の文字からマスクが開始されます。 ○ end_index：手法は文字列の先頭から指定された文字までのデータをマスクします。空白のままにすると、最後の文字でマスクが終了します。 ○ exclude_char：マスクから除外する文字を定義します。 ○ replacement_char：マスクに使用する文字を定義します。何も指定しない場合、デフォルトでアスタリスク (*) が使用されます。
静的置換 (Static Replace)	この手法では、値を静的な値に置換します。文字列、数値、および日付データでこの手法を使用できます。 ○ date_time_value：日付値をこの日付に置き換えます。yyyy-MM-dd HH:mm:ss 形式を使用します。 ○ date_value：日付値をこの日付に置き換えます。 yyyy-MM-dd 形式を使用します。 ○ number_value：数値をこの数値に置き換えます。

手法	説明
	○ string_value：文字列値をこのテキストに置き換えます。 ○ number_type：整数を受け入れます。
ランダム置換	この手法では、値をランダムに生成された値に置換します。文字列および数値データでこの手法を使用できます。
削除	この手法では、値を削除して空の (null) 値に置き換えます。
アクションなし	この手法はプレースホルダーです。選択されている場合、フィールドは変更されません。
X との選択的置換	文字列データを変換し、機密性の高い文字を選択的に文字 X に置き換えます。 注: 「データディスカバリー (クラシック) の詳細」のデータパターンに対するデフォルトのテクニック。
データパターンの匿名化	基礎となるコンテキストを損なわずに、非構造化データフィールド内で検出されたデータパターンのみを匿名化します。 注: この手法の設定については、「データディスカバリー (クラシック) の詳細」のデータパターン匿名化手法の設定を参照してください。

4. 事前定義された手法を使用しない場合は、[カスタム手法を追加 **(Add custom technique)]** を選択します。
5. [手法をカスタマイズ **(Customize technique)]** フォームのフィールドに入力します。

フィールド 1	説明
基本手法	カスタム手法は事前定義された手法に基づいているため、事前定義された手法を選択します。
手法名	手法の名前を入力します。
手法の説明	手法の説明を入力します。

6. [次へ] を選択します。
7. 手法のパラメーターを入力します。
使用可能なパラメーター化された値は、選択したプライバシー手法によって異なります。[アクションなし] および [削除] 手法にはパラメーター化された値はありません。

基本手法のプライバシーのパラメーター化された値

基本手法	プライバシー手法のパラメーターの値	説明	デフォルト値
選択的置換 (Selective Replace)	end_index	手法は文字列の先頭から指定された文字までのデータをマスクします。空白のままにする	(空)

基本手法	プライバシー手法のパラメーターの値	説明	デフォルト値
		と、最後の文字でマスクが終了します。	
選択的置換 (Selective Replace)	exclude_char	マスクをスキップする文字。この値に使用できるのは 1 文字のみです。複数入力した場合は、最初の文字が使用されます。	(空)
選択的置換 (Selective Replace)	[replacement_char]	選択的置換を使用して値を置換するときに使用する文字。	他の値が入力されていない場合は、アスタリスク (*) が使用されます。
選択的置換 (Selective Replace)	start_index	手法は指定された文字以降のデータをマスクします。	空白のままにすると、最初の文字からマスクが開始されます。
静的置換 (Static Replace)	date_time_value	日時値をこの日付に置き換えます。yyyy-MM-dd HH:mm:ss 形式を使用します。	1988-11-11 10:10:10
静的置換 (Static Replace)	date_value	日付値をこの日付に置き換えます。 yyyy-MM-dd 形式を使用します。	1988-11-11
静的置換 (Static Replace)	number_value	[数値] の値をこの数値に置き換えます。	1234567
静的置換 (Static Replace)	string_value	文字列値をこのテキストに置き換えます。	TEXT123
ランダム置換	preserve_data_length	データ長を保持する場合は **true** に設定します。匿名化されたデータは、元のデータと同じ長さになります。	true

8. [カスタム手法を作成 **(Create Custom Technique)]** を選択します。
   カスタム手法が匿名化手法に追加されます。

次のタスク
匿名化ポリシーを構成して、データを匿名化するときに使用する手法を指定するには、「匿名化ポリシーの作成」を参照してください。

匿名化ポリシーの作成

匿名化ポリシーを構成して、データを匿名化するときに使用する手法を指定します。

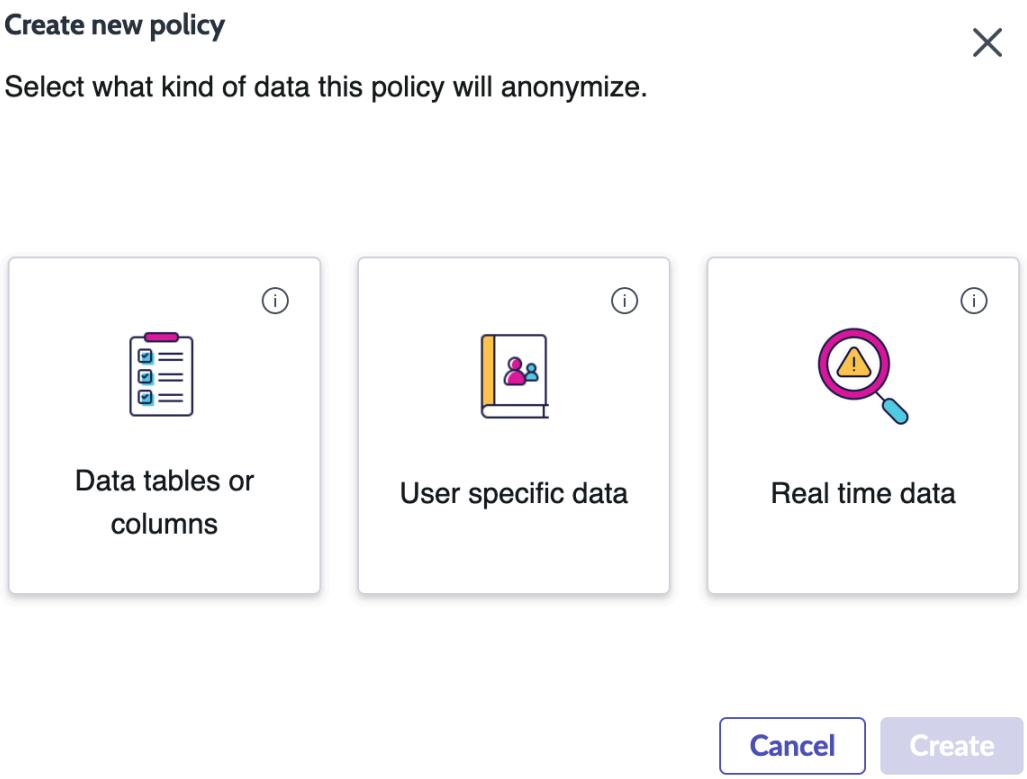
始める前に
Data Privacy 構成では、ユースケースに応じてテーブル、sys_user など、および列を匿名化するように定義し、データの匿名化中に使用されるパラメーター化される手法のタイプを指定します。

> **注:** プライバシー構成を完了するには、最初にデータプライバシー手法構成を設定する必要があります。詳細については、「匿名化手法の作成」を参照してください。

必要なロール：data_privacy_admin および admin

手順

1. **data_privacy_admin** ロールに昇格させます。
   ロールの昇格の詳細については、「特権ロールへの昇格」を参照してください。
2. 移動先 システムセキュリティ > データプライバシー > 匿名化.
   すべての匿名化ポリシーが表示されます。公開されたポリシーは、匿名化ジョブをスケジュールするために使用できます。
3. [新しいポリシーを作成] を選択します。
4. データテーブルまたは列、ユーザー固有のデータ、リアルタイムデータのどれを匿名化するかを選択します。



データ タイプ	説明
データテーブルまたは列	データポリシーに一致するレコードは匿名化されます。
ユーザー固有のデータ	匿名化するユーザーまたはユーザーグループのセットを選択します。
リアルタイムデータ	列のセットのリアルタイムエントリを匿名化します。

Data Privacy ポリシーは分類されたデータにのみ適用できます。データ分類の詳細については、「データ分類」を参照してください。

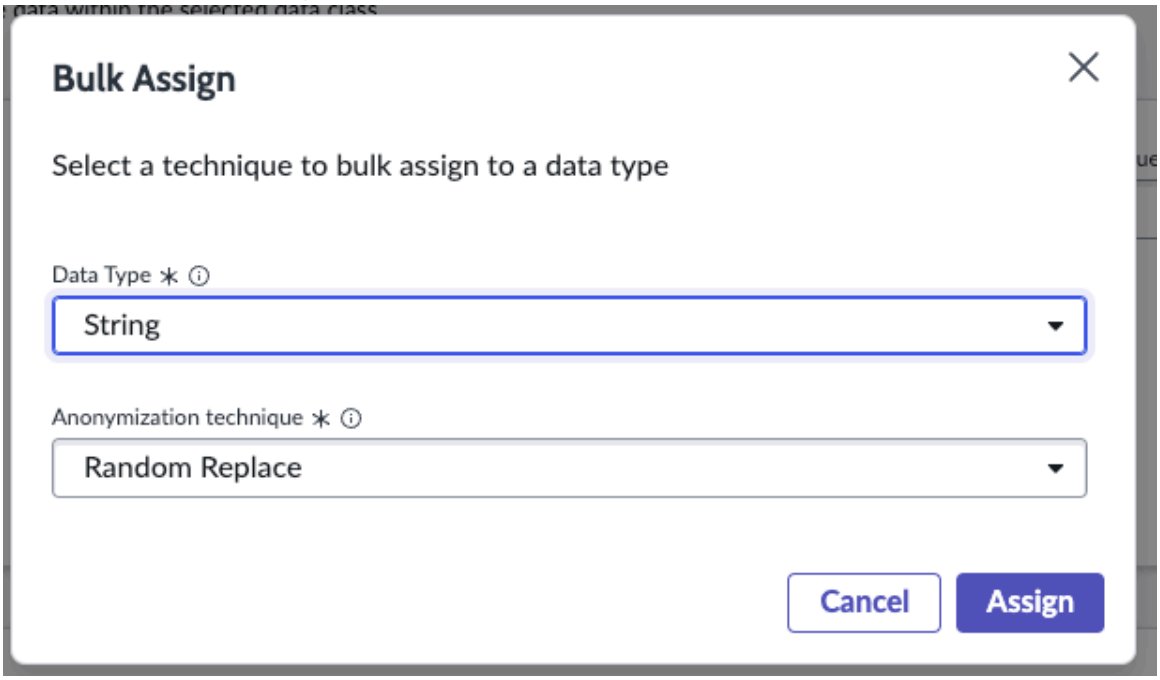
5. [作成] を選択します。
   ポリシー、[詳細を定義]、および [手法を割り当て **(Assign techniques)**]を完了するために必要な一連のステップがあります。ユーザー固有のデータのポリシーを定義する場合は、ユーザー参照の選択も必要です。

6. 新しい匿名化ポリシーの詳細を定義します。
   - [名前] フィールドにポリシー名を入力し、[説明] フィールドにポリシーの説明を入力します。
   - [有効化チャネル] でポリシーとチャネル優先度を自動的に有効にするチャネルを定義します
   - [データクラス] フィールドで、このポリシーで使用するデータクラスを選択します。
   - リアルタイムデータの匿名化をオンまたはオフにします。リアルタイムデータの匿名化がオンになっている場合は、ステップ 8 を参照してください

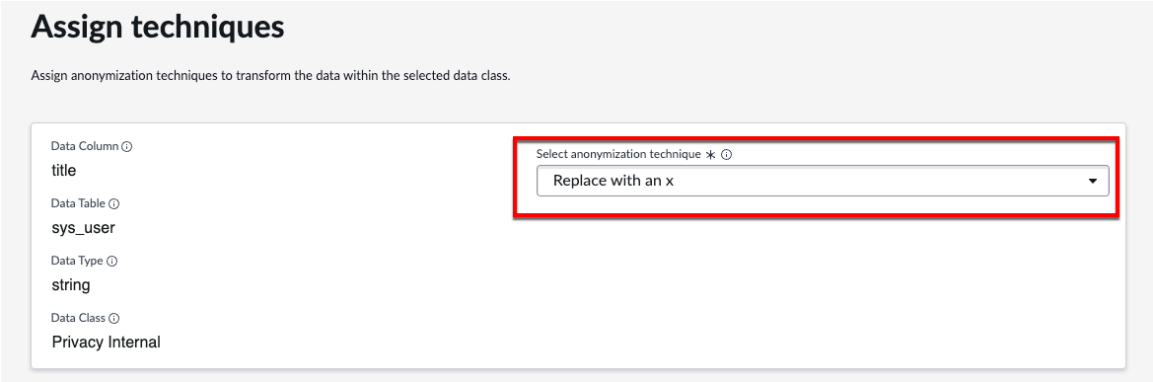
   注: エントリを匿名化しない場合は、エントリを空のままにするのではなく、**[DoNothing]** 手法を選択します。[プライバシー手法構成] フィールドの値が空になっているポリシーは、Data Privacy ジョブで使用されるときに実行できません。

データクラスを選択した後に、定義されたデータクラスに対して返された各レコードの [手法を割り当て (Assign techniques)] フォームが表示されます。

7. 選択したデータクラスの匿名化手法を割り当てます。



オプション	説明
[手法の一括割り当て **(Bulk Assign Techniques)]** を選択します。	選択したデータクラスのすべてのデータレコードに匿名化を適用します。選択したデータタイプのすべてのエントリに適用するデータタイプと匿名化手法を選択します。さまざまなデータタイプの追加の一括割り当てについて、この手順を繰り返します。 データタイプのリストについては、「匿名化でサポートされているフィールドタイプ」を参照してください。
各データ列レコードの [匿名化手法 **(Anonymization technique)]** を選択します。	データプライバシープロセッサーユーザーが、データプライバシージョブの作成時に匿名化するレコードを選択できます。選択したデータクラスの各データレコードに匿名化を個別に適用します。



8. オプション: スキャンする子テーブルを入力します。
   親の子テーブルは匿名化されます。テーブルに子がない場合、このオプションは使用できません。

   ⚠ 警告: 子ジョブが失敗すると、親ジョブも失敗します。

9. オプション: [データパターンの匿名化] を選択した場合は、使用する匿名化手法を選択します。
10. オプション: データパターンの順序を設定します。
11. オプション:

    💡 ヒント: [テスト] 機能を使用して、サンプル入力をテストします。スキャン時間、結果、検出されたパターンなどの測定基準を結果からレビューできます。

    [テスト] ボタンを選択してポリシーをテストします。

12. ℹ 重要: すべてのテーブルに正しい sys_dictionary エントリが必要です。

    [保存] を選択します。
13. [公開] を選択すると、スケジュール設定の匿名化ポリシーが更新され、匿名化ポリシーに戻ります。

    ℹ 注: 匿名化ジョブのスケジュール設定には、公開されたポリシーのみを使用できます。

次のタスク
匿名化ジョブの作成。

データ匿名化クローン要求の設定

Data Privacy クローン統合は、PostClone スクリプトを使用して構成され、ターゲット上で構成されたポリシー用の Data Privacy ジョブを作成して実行します。スクリプトが実行された後は、ユーザーに匿名化されたデータが表示され、元のデータにアクセスできなくなります。

始める前に
データプライバシー PostClone スクリプトは、データプライバシープラグイン (sn_dp_store_app) のアクティブ化とともにインストールされます。詳細については、「データプライバシーのアクティブ化」を参照してください。

必要なロール：data_privacy_clone_processor、data_privacy_admin、および admin

手順

1. ソースインスタンスでデータプライバシープラグイン (sn_dp_store_app) をアクティブ化します。
   データプライバシー PostClone スクリプトがインストールされます。

2. **data_privacy_admin** ロールに昇格させます。
   ロールの昇格の詳細については、「特権ロールへの昇格」を参照してください。
3. 移動先 システムセキュリティ > データプライバシー > 匿名化.
4. [新しいポリシーを作成] を選択します。
   詳細については、「匿名化ポリシーの作成」を参照してください。
5. データテーブルまたは列を選択します。
6. [作成] を選択します。
7. 名前を入力し、データクラスを選択します。
8. [クローン作成中にポリシーを有効にする **(Activate the policy during cloning)**] を選択します。
9. 複数のクローンポリシーがある場合に実行するポリシーの順序を選択します。
   アプリケーション順序が上位になっている Postclone 構成用のデータプライバシージョブは、上位のジョブが他の下位のジョブに関連するテーブルを含んでいない場合、下位のジョブの前に開始されることがあります。
10. [続行] を選択します。
11. ポリシー構成を完了し、ポリシーを公開します。
12. データプライバシー構成をバックアップします。
13. 匿名化ジョブをスケジュールします。
    詳細については、「匿名化ジョブの作成」を参照してください。
14. Data Privacy アドミニストレーターとして、クローン要求を送信します。

結果
データプライバシー PostClone スクリプトはターゲットインスタンスで実行され、PostClone スクリプトはターゲットインスタンス上にデータプライバシー連携ジョブレコードを作成します。連携ジョブは、ターゲットインスタンスで、各クローン後ポリシー用のデータプライバシージョブを作成してアプリケーションの順序で実行します。 バックアップソースはターゲットインスタンスにクローン作成されます。データプライバシー PostClone スクリプトは、ターゲット上で構成されたポリシー用のデータプライバシージョブを作成して実行します。

昇格されたデータプライバシークローンプロセッサーは、ターゲットインスタンスにログオンし、dp_federated_job.list および dp_job.list でクローン後の連携ジョブのステータスを監視できます。

匿名化ジョブの作成

非本番インスタンスでユーザーおよびデータクラスのジョブに匿名化されたデータを使用するように、本番インスタンスで Data Privacy ジョブを構成します。

始める前に
データプライバシージョブは、次の 2 つの匿名化ユースケースをサポートしています。

- 特定の sys_users の機密データ
- 特定のデータクラスの機密データ。

必要なロール：data_privacy_processor および admin

手順

1. **data_privacy_processor** ロールに昇格させます。
   ロールの昇格の詳細については、「特権ロールへの昇格」を参照してください。
2. 移動先 システムセキュリティ > データプライバシー > 匿名化.

3. 匿名化ポリシーで、ジョブで使用するポリシーの [ジョブをスケジュール] を選択します。
匿名化ジョブをスケジュールするには、ポリシーが公開済みステータスである必要があります。

> ⚠ **警告:** 匿名化ジョブは非常に破壊的であり、ロールバック時にのみ取り消すことができます。ジョブをスケジュールする前に、処理されたレコードやテーブルなどのすべての情報を再確認してください。

4. フォームのフィールドに入力します。

データプライバシージョブのフィールド

フィールド	説明
使用されるポリシー (Policy used)	このジョブに使用する選択されたプライバシーポリシー構成の読み取り専用の名前。ポリシーに関する補足情報を表示するには、ポリシーを編集します。 プライバシーポリシー構成の詳細については、「匿名化ポリシーの作成」を参照してください。
ジョブの説明	ジョブの説明。
開始時間	このジョブを実行する期間の開始時間 (**HH:MM:SS** 形式)。
終了時間	このジョブを実行する期間の終了時間 (**HH:MM:SS** 形式)。終了時間は開始時間よりも後でなければなりません ジョブは、このフィールドに入力された時間まで実行されます。ジョブがまだ完了していない場合、ジョブは一時停止し、次の期間の開始時に再開されます。
予行演習	ジョブをテストとして実行します。このジョブの実行時に影響を受けるレコードはありません。結果は、ジョブが実行された場合のように [ジョブ] リストに表示されます。 **注:** ロールバックを使用してデータプライバシージョブを構成するときは、[予行演習] をオフにする必要があります。詳細については、「Data Privacy ジョブのロールバック」を参照してください。
ユーザ選択のタイプ	匿名化するユーザーまたはグループを選択します。
ユーザー/グループを選択	このジョブで匿名化する特定のユーザーまたはグループのセットを選択します。最大 1000 人のユーザーをサポートします。

フィールド	説明
	注: この必須フィールドは、選択したプライバシーポリシーの条件でユーザーレコードを選択する必要がある場合にのみ表示されます。

5. **重要:** スケジューリング前およびジョブ中に、すべてのテーブルに正しい sys_dictionary エントリが必要です。

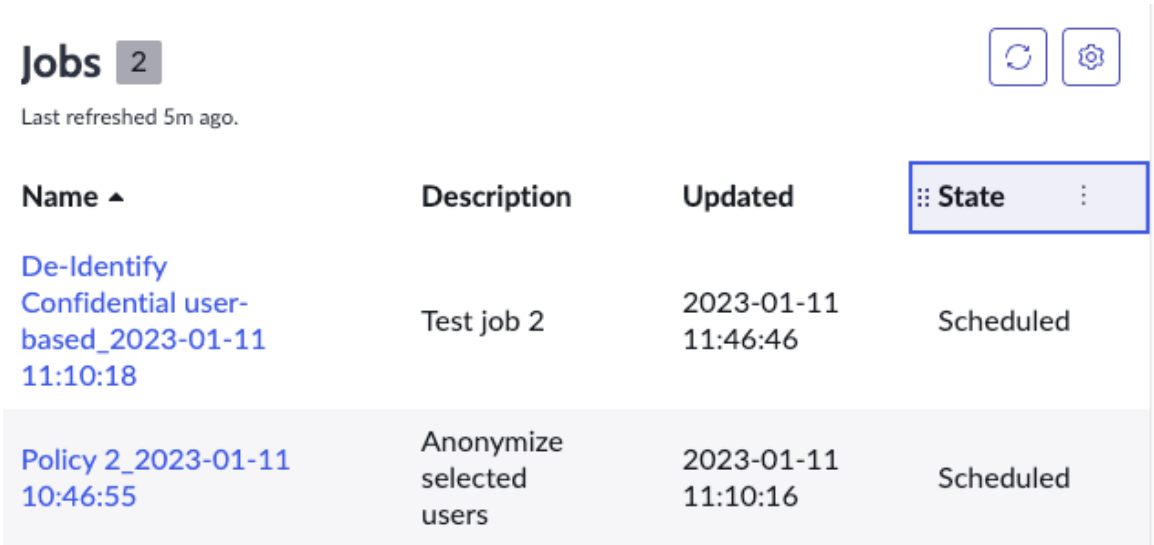
フォームで [ジョブをスケジュール] を選択して、匿名化をジョブキューに配置します。

[開始時間] と [終了時間] フィールドで選択した時間の間でジョブが実行されます。ジョブが期間開始と期間終了の間に完了しなかった場合、ジョブは次の期間の開始時に続行されます。

[予行演習] が選択されている場合でも、ジョブは 1 回のみ実行できます。同じポリシーに基づいてジョブを再度実行するには、[ジョブをスケジュール] を選択し、同じフィールド値を使用してフォームに入力します。

警告: 暗号化された列の匿名化ジョブでは、ジョブの対象となるすべての暗号化された列を復号化して再暗号化します。これを防ぐには、ポリシー手法として [アクションなし] を選択します。

ジョブが [ジョブ] ペインに一覧表示されま



す。

フィールド	説明
名前	匿名化ジョブの名前。
説明	匿名化ジョブの説明。
更新日時	ジョブが最後に更新された日時。
状況	データプライバシージョブのステータス：

フィールド	説明
	○ スケジュール済み：新しいジョブのデフォルトステータス。 ○ 完了：ジョブが選択したデータを正常に匿名化しました。 ○ キャンセル：ジョブは手動でキャンセルされました。 ○ エラー：ジョブの保存中に問題が発生しました。ジョブを再スケジュールするか、新しいジョブを作成します。エラーが引き続き発生する場合は、構成に問題がある可能性があります。 ○ ロールバック中：ジョブは匿名化をロールバックするように設定されています。 ○ ロールバック完了：匿名化ジョブのロールバックが正常に完了しました。 読み取り専用フィールド。

6. [ジョブ] ペインからジョブを選択して、ジョブサマリーを開きます。
ジョブがスケジュールされた後、[ジョブのキャンセル] および [一時停止] ボタンがジョブサマリーに表示されます。

スケジュール済みジョブの追加フィールド

フィールド	説明
推定レコード数	この予行演習ジョブが実行される前に影響するレコードの推定数。読み取り専用フィールド
処理されるデータレコードの合計数 (Total data records processed)	このジョブによって影響を受ける個々のデータレコードの合計数。読み取り専用フィールド。
処理されるデータテーブルの合計数 (Total data tables processed)	このジョブによって処理されたデータテーブルの合計数。読み取り専用フィールド。
ロールバックの残り時間 (Time remaining to rollback)	完了したデータ匿名化ジョブをロールバックしてデータを匿名化解除できる残り時間。読み取り専用フィールド。
処理されたユーザーの合計数 (Total users processed)	このジョブによって影響を受ける個々のユーザーレコードの合計数。読み取り専用フィールド。
ジョブのキャンセル	匿名化ジョブをキャンセルする場合に選択します。これはジョブの開始時間の前に選択する必要があります。 選択すると、ジョブステータスが [キャンセル] に更新されます。
一時停止	ロールバックが選択されている場合、選択するとジョブが一時停止し、記録がロールバックされます。ロールバックコンテキストの 3 日間の有効期限が過ぎると、警告メッセージが表示されます。 これは、ジョブの開始時間からジョブの終了時間までの間に選択する必要があります。

フィールド	説明
	選択すると、ジョブステータスが [一時停止] に更新されます。
再開	一時停止したジョブを再開します。一時停止した場合、再開されたジョブのロールバックはサポートされません。ジョブをキャンセルし、Data Privacy ジョブを作成します。記録では、期限切れになっていないロールバックコンテキストが使用されます。 選択すると、ジョブステータスが [スケジュール済み] に更新されます。
エクスポート	データプライバシージョブの詳細の .PDF ファイルをダウンロードします。

データの匿名化の並列ジョブのアクティブ化

並列ジョブを使用して、匿名化ジョブの実行時間を短縮します。

始める前に
必要なロール：admin

手順
デフォルトでは、データクラスおよびユーザーベースの匿名化ジョブは、シングルスレッドを使用して実行されます。データクラスおよびユーザーベースのジョブに対して複数のスレッドをアクティブ化できます。また、クローン作成中に使用される連携ジョブでは、デフォルトで 3 ワーカーの並列ジョブが使用されます。

1. ナビゲーションフィルターに、sys_properties.list と入力します。
2. プロパティ com.glide.data_privacy.max_parallel_workers を作成します。
3. オプション: 連携ジョブの並列ワーカーの数を調整するには、dp.max_concurrent_clone_item_workers プロパティを編集します。
4. [タイプ] が integer であることを確認します。
5. **重要:** ノードあたりのワーカー数は、少なめ (2 または 3) から始めることをお勧めします。

   [値] フィールドに並列ジョブの数を 5 までの範囲で入力します。

リアルタイム匿名化

リアルタイム匿名化 (RTA) ポリシーを使用して、データエントリをリアルタイムで匿名化します。

リアルタイム匿名化の概要

ユーザーは、[匿名化ポリシー] ページで [リアルタイム匿名化] を選択し、適切なデータチャネルを選択して、RTA ポリシーを作成します。たとえば、リアルタイム匿名化で仮想エージェントを使用できます。データチャネルとして選択した仮想エージェントを使用して匿名化ポリシーを作成します。

ターゲットテーブルから RTA の列を選択すると、アクティブなデータパターンが使用され、そのポリシーが RTA のターゲット列の任意の有効なレコードエントリに適用されます。エントリがアクティブなデータパターンに一致すると、関連付けられた匿名化手法を使用して匿名化されます。

ヒント: 匿名化手法を変更する必要がある場合は、「データディスカバリー パターンの構成」を参照してください。

リアルタイム匿名化の失敗

RTA ポリシーが失敗した場合は、リアルタイム匿名化の失敗 テーブルでそのステータスを確認できます。

リアルタイム匿名化の失敗

リアルタイム匿名化 (RTA) の失敗について説明します。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > システムセキュリティ > データプライバシー (従来).
2. テーブルの RTA エラーを確認します。

フィールドタイトル	説明
テーブルの列	障害が発生したテーブル列。
障害の原因	障害の原因のカテゴリ。
障害の原因の詳細	障害の原因を要約した簡単な説明。
プライバシー構成	失敗した RTA プライバシー構成。
レコード	失敗した特定のレコード。
テーブル名	障害が発生したテーブル。
タイムスタンプ	障害の時刻。

データプライバシーのアクティブ化

データプライバシーにはデータ分類と匿名化が含まれ、ServiceNow Store からインストールされます。

始める前に
データの匿名化を使用するには、まず ServiceNow Vault エンタイトルメントを使用してデータプライバシー (従来) を有効にする必要があります。詳細については、「Data Privacy (Classic) のアクティブ化」を参照してください。

> **注:** データプライバシーストアアプリをインストールすると、データディスカバリーストアアプリ、データプライバシー (従来) プラグイン、およびデータ分類プラグインが自動的にインストールされます。

必要なロール：admin

手順

1. 移動先 すべて > システムアプリケーション > 利用可能なすべてのアプリケーション > すべて > データプライバシー.
2. フィルター基準と検索バーを使用してアプリケーションを検索します。

   名前または ID でアプリケーションを検索できます。アプリケーションが見つからない場合は、ServiceNow Store へのリクエストが必要になることもあります。

   ServiceNow Store Web サイトにアクセスして利用可能なすべてのアプリを表示し、ストアにリクエストを送信する方法について確認してください。リリースされたすべてのアプリのリリース

ノート情報については、「ServiceNow Storeバージョン履歴のリリースノート」を参照してください。

3. リストからバージョンを選択し、[インストール] を選択します。

   表示されるインストールダイアログには、アプリケーションとともにインストールされている依存関係が一覧表示されます。

4. プロンプトが表示された場合は、ServiceNow Store へのリンクに従って、依存関係のエンタイトルメントを取得します。
5. オプション: 利用できるデモデータをインストールするには、[デモデータのロード] チェックボックスを選択します。

   (Optional) デモデータには、一般的なユースケース向けのアプリケーション機能を説明するサンプルレコードが含まれています。開発またはテストインスタンスで初めてアプリケーションをインストールする場合は、デモデータを読み込みます。

   重要: インストール時にデモデータを読み込んでおかないと、後から読み込むことはできません。

6. [インストール] を選択します。

仮想エージェントのデータプライバシー

データプライバシーを使用して、仮想エージェントの会話中に機密データと PII を検出してマスクできます。

始める前に

必要なロール:virtual_agent_data_privacy_admin

このタスクについて

データプライバシーを使用して、仮想エージェントの会話中に機密データと PII を検出してマスクします。データプライバシーは、減価償却された機密データハンドラーに代わるものです。詳細については、「 廃止プロセス [KB0867184] 」を参照してください。セットアッププロセスは、機密データハンドラーが以前にインストールおよび構成されていたかどうかによって異なる場合があります。

注: 以前に機密データハンドラーを構成した場合、設定はポリシーとしてデータプライバシーに移行されます。詳細については、「匿名化ポリシーの作成」を参照してください。

手順

1. 移動先 すべて > 対話型インターフェース > 設定.
2. [機密データの検出] で、[すべて表示] を選択します。
   ボタンが [データプライバシーを取得 **(Get data privacy)]** の場合、続行するにはデータプライバシープラグインをインストールする必要があります。詳細については、「Data Privacy (Classic) のアクティブ化」を参照してください。
3. スライダーを [アクティブ] に設定します。
4. オプション: データプライバシーがまだインストールされていない場合は、続行するためにデータプライバシープラグインをインストールするように求められます。詳細については、 Data Privacy (Classic) のアクティブ化 を参照してください。
5. 適切な会話フローを選択します。

a. 要求者からエージェントへ：エージェントチャット 会話で要求者が入力した機密データを検出してマスクします。

b. エージェントから要求者へ：エージェントチャット 会話でエージェントが入力した機密データを検出してマスクします。

c. 要求者から 仮想エージェント へ：仮想エージェント 会話で要求者が入力した機密データを検出してマスクします。

注: 少なくとも 1 つの会話フローを選択する必要があります。

6. [データプライバシーで管理] を選択します。
以前に機密データハンドラーを構成したことがある場合、設定はポリシーとしてデータプライバシーに移行されます

7. 仮想エージェントの新しいデータプライバシーポリシーを作成します。データプライバシーポリシーの詳細については、 匿名化ポリシーの作成 を参照してください。

ドメインセパレーションと Data Privacy

データプライバシー ではドメインセパレーションはサポートされていません。 ドメインセパレーションでは、データ、プロセス、および管理タスクをドメインと呼ばれる論理的なグループに分けることができます。どのユーザーがデータを表示できるか、データにアクセスできるかなど、このアプリケーションのいくつかの側面を制御できます。

サポートレベル: サポートなし

- ドメインフィールドがデータテーブル に存在している可能性がありますが、データを管理するロジックがありません。
- このレベルでは、ドメイン分割は考慮されません。

サポートレベルの詳細については、「アプリケーションでのドメインセパレーションのサポート)」を参照してください。

関連トピック

サービスプロバイダーのドメインセパレーション

匿名化でサポートされているフィールドタイプ

データを匿名化するときにサポートされているフィールドタイプを確認します。

注: 分類されたすべてのフィールドタイプが匿名化に使用できるわけではありません。

次の表に示すように、一部の高リスクフィールドタイプはデフォルトでオフになっています。フィールドの詳細については、「フィールドタイプ」を参照してください。

匿名化でサポートされているフィールドタイプ

フィールドタイプ	デフォルトで利用可能
audio	いいえ
状態	いいえ
condition_string	いいえ
currency	はい

フィールドタイプ	デフォルトで利用可能
decimal	はい
due_date	はい
浮動小数点数	はい
glide_date	はい
glide_date_time	はい
glide_duration	いいえ
glide_time	はい
HTML	いいえ
アイコン	いいえ
integer	はい
ip_addr	いいえ
ip_address	いいえ
journal	はい
journal_input	はい
journal_list	はい
longint	はい
name_values	いいえ
percent_complete	いいえ
phone_number_e164	はい
price	はい
string	はい
string_full_utf8	はい
Translation_html	いいえ
translated_text	いいえ
url	いいえ
user_image	いいえ
video	いいえ
wiki_text	いいえ

データプライバシーロール

データプライバシー は次のロールを追加します。

データプライバシー アドミニストレーター

Data Privacy アドミンロールは、Data Privacy のテクニックとポリシーを作成するために使用されるアドミンロールです。

データプライバシー アドミニストレーター **[data_privacy_admin]**

ロールを含む

なし

グループにアサイン

なし

登録

いいえ

昇格

あり

考慮事項

より限定的なロールを使用できる場合は、このロールをユーザーにアサインしないでください。

「データプライバシーアドミニストレーター」ロールをアサインするには、ユーザーをセキュリティアドミンロールに昇格させてロールを追加します。

Data Privacy 監査人

Data Privacy 監査人は、Data Privacy レコードを表示するために使用される読み取り専用のロールです。

Data Privacy 監査人 **[data_privacy_auditor]**

ロールを含む

なし

グループにアサイン

なし

登録

いいえ

昇格

いいえ

考慮事項

なし

Data Privacy クローンプロセッサー

Data Privacy クローンプロセッサーロールを持つユーザーは、データクラスの Data Privacy ジョブを作成して実行できます。

Data Privacy クローンプロセッサー **[data_privacy_clone_processor]**

ロールを含む

なし

グループにアサイン

なし

登録

いいえ

昇格

あり

考慮事項

なし

データプライバシープロセッサー

Data Privacy プロセッサーロールを持つユーザーは、ユーザー [sys_user] テーブルで Data Privacy ジョブを作成して実行できます。

Data Privacy プロセッサー [data_privacy_processor]

ロールを含む

なし

グループにアサイン

なし

登録

いいえ

昇格

あり

考慮事項

なし

Data Privacy (Classic)

データプライバシー (従来) は、PII を匿名化するためのデータ分類、テクニック、およびジョブを提供する従来のアプリケーションです。

データプライバシークラシック

注: このセクションは、データプライバシー (従来) 用です。最新のストアバージョンについては、 データプライバシー を参照してください。

Data Privacy (Classic) のアクティブ化

admin ロールを持っている場合は、プラットフォームセキュリティ のデータプライバシープラグイン (com.glide.data_privacy) をアクティブ化できます。このアプリケーションにはデモデータが含まれています。まだインストールされていない場合は、関連する ServiceNow Store アプリケーションとプラグインをインストールします。

始める前に

重要: このセクションは、データプライバシー (従来) 用です。最新のストアバージョンについては、 データプライバシー を参照してください。

データプライバシーでは、他の ServiceNow AI Platform® とは別のサブスクリプションが必要です。

サブスクリプションを購入するには、ServiceNow アカウントマネージャーにお問い合わせください。サブスクリプションを購入すると、特定のプラグインが自動的にアクティブになります。有料プラグインが自動的にアクティブになっていない場合は、インスタンスの [すべてのアプリケーション] リストから手動でアクティブ化できます。

注: サブスクリプションを購入する前に、30 日間の試用期間でデータプライバシー (従来) とデータディスカバリーを評価できます。試用期間が終了すると、ライセンスなしでデータを検出したり、匿名化ジョブを実行したりすることができなくなります。

必要なロール：admin

このタスクについて
次のアイテムがデータプライバシーとともにインストールされます。

- プラグイン
- ロール
- テーブル

詳細については、「データプライバシー (従来) とともにインストールされるもの」を参照してください。

手順

1. 移動先 すべて > システムアプリケーション > 利用可能なすべてのアプリケーション > すべて.
2. フィルター基準と検索バーを使用して Data Privacy プラグイン (com.glide.data_privacy) を検索します。

   名前または ID でプラグインを検索できます。プラグインが見つからない場合は、ServiceNow 担当者から要求する必要があります。
3. [インストール] を選択して、インストールプロセスを開始します。

   注: ドメインセパレーションと代理アドミンがインスタンスで有効になっている場合、管理ユーザーはグローバルドメインに含まれている必要があります。それ以外の場合、次のエラーが表示されます： 「別の操作が実行されているため、アプリケーションのインストールは利用できません： <プラグイン名> のプラグインの有効化 (Application installation is unavailable because another operation is running: Plugin Activation for <plugin name>)」

   インストールが完了するとメッセージが表示されます。プラグインとともにインストールされるコンポーネントの詳細については、「 アプリケーションとともにインストールされているコンポーネントの検索 」を参照してください。

データプライバシー (従来) とともにインストールされるもの

データプライバシープラグイン (com.glide.data_privacy) とともにインストールされるコンポーネントについて説明します。

インストールされるテーブル

テーブル	説明
データプライバシー連携ジョブ [dp_federated_job]	データプライバシーの連携ジョブ
Data Privacy ジョブ [dp_job]	データプライバシージョブ

テーブル	説明
データプライバシージョブ [dp_job_summary]	Data Privacy ジョブのサマリー
Data Privacy 手法 [dp_technique]	Data Privacy 手法
プライマリ参照リンク [dp_primary_reference]	プライマリ参照リンク
プライバシーに分類されたフィールドの手法 [dp_field_technique]	データプライバシーに分類されたフィールドの手法
プライバシー構成 [dp_configuration]	データプライバシー構成
プライバシー手法構成 [dp_technique_with_params]	データプライバシー手法
プライバシー手法パラメーター [dp_technique_with_parameter]	データプライバシー手法とパラメーター
プライバシー手法のパラメーターの値 [dp_technique_with_parameter_value]	Data Privacy 手法とパラメーターで使用されるパラメーター値

Data Privacy (Classic) 構成

Data Privacy 手法とポリシーを作成する方法、および Data Privacy ジョブを作成して実行する方法について説明します。

重要: このセクションは、データプライバシー (従来) 用です。最新のストアバージョンについては、 データプライバシー を参照してください。

データプライバシー手法

データプライバシー手法は、データの匿名化方法を決定するために選択するオプションです。データプライバシージョブで参照するデータプライバシー手法を作成する必要があります。プライバシー手法を関連するプライバシー手法構成に関連付けるには、「データプライバシー 手法構成の作成」を参照してください。

データプライバシーポリシー

データプライバシーポリシーを構成して、データを匿名化するときに使用するデータプライバシー手法を指定します。詳細については、「Data Privacy ポリシーの作成」を参照してください。

データプライバシージョブ

データプライバシージョブは、これらすべてのコンポーネントを使用してデータを匿名化します。これらのジョブの詳細については、「データプライバシージョブの構成」を参照してください。

データプライバシー 手法構成の作成

Data Privacy 手法構成を作成して、Data Privacy でデータを匿名化する方法をカスタマイズします。

始める前に
必要なロール：data_privacy_admin および admin

手順

1. **data_privacy_admin** ロールに昇格させます。
   ロールの昇格の詳細については、「特権ロールへの昇格」を参照してください。
2. 移動先 システムセキュリティ > データプライバシー > プライバシー手法構成.
3. [**New**] をクリックします。
4. [名前] フィールドに、プライバシー手法構成の名前を入力します。
5. [プライバシー手法] フィールドで、プライバシー手法を選択します。

データプライバシー手法

テクニック	説明
アクションなし	この手法はプレースホルダーです。選択されている場合、フィールドは変更されません。
ランダム置換	この手法では、値をランダムに生成された値に置換します。文字列および数値データでこの手法を使用できます。
選択的置換 (Selective Replace)	この手法では、文字列データを選択的に置換します。入力の開始インデックスと終了インデックスの間のすべての文字が、選択した文字に置き換えられます。マスクから除外する文字を指定できます。 ○ start_index：手法は指定された文字以降のデータをマスクします。空白のままにすると、最初の文字からマスクが開始されます。 ○ end_index：手法は文字列の先頭から指定された文字までのデータをマスクします。空白のままにすると、最後の文字でマスクが終了します。 ○ exclude_char：マスクから除外する文字を定義します。 ○ replacement_char：マスクに使用する文字を定義します。何も指定しない場合、デフォルトでアスタリスク (*) が使用されます。

テクニック	説明
静的置換 (Static Replace)	この手法では、値を静的な値に置換します。文字列、数値、および日付データでこの手法を使用できます。 ○ date_time_value：日付値をこの日付に置き換えます。yyyy-MM-dd HH:mm:ss 形式を使用します。 ○ date_value：日付値をこの日付に置き換えます。 yyyy-MM-dd 形式を使用します。 ○ number_value：数値をこの数値に置き換えます。 ○ string_value：文字列値をこのテキストに置き換えます。
削除	この手法では、値を削除して空の (null) 値に置き換えます。

注: 以前にサポートされていた値 **Replace** は廃止されたので使用しないでください。これは名前が **Replace-Deprecated** に変更されています。

6. ヘッダーを右クリックし、コンテキストメニューの [保存] を選択します。
レコードが保存されると、[プライバシーのパラメーター化された値] リストが表示されます。
7. [プライバシーのパラメーター化された値] リストのレコードを使用して、データプライバシー手法の構成をカスタマイズします。
使用可能なパラメーター化された値は、選択したプライバシー手法によって異なります。[アクションなし] および [削除] 手法にはパラメーター化された値はありません。

選択的置換のプライバシーのパラメーター化された値

プライバシー手法のパラメーターの値	説明	デフォルト値
char_to_replace	選択的置換を使用して値を置換するときに使用する文字。	*
end_index	手法は文字列の先頭から指定された文字までのデータをマスクします。空白のままにすると、最後の文字でマスクが終了します。	(空)
exclude_char	マスクをスキップする文字。この値に使用できるのは 1 文字のみです。複数入力した場合は、最初の文字が使用されます。	(空)
start_index	手法は指定された文字以降のデータをマスクします。	1

置換のプライバシーのパラメーター化された値

プライバシー手法のパラメーターの値	説明	デフォルト値
date_time_value	日時値をこの日付に置き換えます。yyyy-MM-dd HH:mm:ss 形式を使用します。	1988-11-11 10:10:10
date_value	日付値をこの日付に置き換えます。 yyyy-MM-dd 形式を使用します。	1988-11-11
number_value	[数値] の値をこの数値に置き換えます。	1234567
preserve_data_length	データ長を保持する場合は **true** に設定します。匿名化されたデータは、元のデータと同じ長さになります。	true
string_value	文字列値をこのテキストに置き換えます。	TEXT123
use_random_generated_value	データをランダムに生成された値に置き換えるには **true** に設定します。ランダム値に置き換えることができるのは文字列と数値のデータのみです。このオプションは、静的な値を上書きします。	false

8. [保存] をクリックします。

Data Privacy ポリシーの作成

Data Privacy ポリシーを構成して、データを匿名化するときに使用する Data Privacy 手法を指定します。

始める前に

Data Privacy 構成では、ユースケースに応じてテーブル、sys_user など、および列を匿名化するように定義し、データの匿名化中に使用されるパラメーター化される手法のタイプを指定します。

> **注:** プライバシー構成を完了するには、最初にデータプライバシー手法構成を設定する必要があります。詳細については、「データプライバシー 手法構成の作成」を参照してください。

必要なロール：data_privacy_admin および admin

手順

1. **data_privacy_admin** ロールに昇格させます。
   ロールの昇格の詳細については、「特権ロールへの昇格」を参照してください。
2. 移動先 システムセキュリティ > データプライバシー (従来) > プライバシーポリシー構成.
3. **[New** (新規)] を選択します。
4. [名前] フィールドに、プライバシーポリシー構成の名前を入力します。

5. [データクラス] フィールドで、このポリシーで使用するデータクラスを選択します。
データプライバシーポリシーは分類されたデータにのみ適用できます。データ分類の詳細については、「データ分類」を参照してください。
データクラスを選択すると、[プライバシーに分類されたフィールドの手法] と [プライバシーのプライマリ参照リンク] のリストがフォームに表示されます。
6. オプション: [クラスのすべてのデータに適用 **(Apply to All Data in Class)**] を選択すると、選択したデータクラスのすべてのデータに匿名化が適用されます。
このフィールドを選択しない場合、データプライバシープロセッサユーザーが、データプライバシージョブの作成時に匿名化するユーザーを選択できます。このフィールドを選択した場合、そのオプションは使用できません。
   - [クローン作成時に適用]：このオプションが利用可能になります。選択すると、データプライバシークローン作成中にプライバシー構成が実行されます。
   - [アプリケーションの順序]：上位のアプリケーション順序を持つクローン後構成のデータプライバシージョブが、下位のジョブの前に開始される場合があります。

   重要: 同じアプリケーション順序で複数のデータプライバシーポリシーを作成することは避けてください。結果として同じ順序の処理順序の一貫性がなくなります。

7. オプション: データプライバシージョブからデータを匿名化解除する機能を有効にするには、[サポートロールバック] を選択します。
詳細については、「データプライバシージョブのロールバック」を参照してください。
データプライバシージョブの作成時に [サポートロールバック] を選択すると、ジョブをロールバックするオプションが利用可能になります。
8. [プライバシーに分類されたフィールドの手法] タブを選択して、[プライバシーに分類されたフィールドの手法] リストを表示します。
9. [テーブル] フィールドのエントリを選択して、各リストエントリの [プライバシー手法構成] フィールドを開きます。
[プライバシーに分類されたフィールドの手法] リストには、選択したデータクラスで匿名化されるすべてのデータが表示されます。これらのエントリごとに、適用するプライバシー手法を選択する必要があります。
10. 適用するプライバシー手法構成を選択します。

    重要: エントリを匿名化しない場合は、エントリを空のままにするのではなく、**[DoNothing]** 手法を選択します。[プライバシー手法構成] フィールドの値が空になっているポリシーは、データプライバシージョブで使用されるときに実行できません。

11. [送信] または [保存] をクリックしてレコードを保存します。

次のタスク
データプライバシージョブの構成。

データプライバシージョブの構成

非本番インスタンスでユーザーおよびデータクラスのジョブに匿名化されたデータを使用するように、本番インスタンスで Data Privacy ジョブを構成します。

始める前に
データプライバシージョブは、次の 2 つの匿名化ユースケースをサポートしています。

- 特定の sys_users の機密データ
- 特定のデータクラスの機密データ。

自動翻訳

必要なロール：data_privacy_processor および admin

手順

1. **data_privacy_processor** ロールに昇格させます。
   ロールの昇格の詳細については、「特権ロールへの昇格」を参照してください。
2. 移動先 システムセキュリティ > データプライバシー > データプライバシージョブ.
3. [Data Privacy ジョブ] リストで、[新規] をクリックします。
4. フォームのフィールドに入力します。

データプライバシージョブのフィールド

フィールド	説明
名前	ジョブの名前。
説明	ジョブの説明。
プライバシー構成	このジョブに使用するプライバシーポリシー構成。プライバシーポリシー構成の詳細については、「Data Privacy ポリシーの作成」を参照してください。
周期	○ **1**回実行:ジョブは、指定された期間内に 1 回実行されます。 ○ 週次:ジョブは、指定された時間枠内で [曜日] フィールドの各日に実行されます。 ○ 月次:ジョブは、指定された時間枠内の [日付] フィールドに入力された毎月の毎日実行されます。
ユーザー	このジョブで匿名化するユーザーまたはユーザーグループを選択します。最大 1000 人のユーザーグループを処理できます。 **注:** このフィールドは、選択したプライバシーポリシーの条件でユーザーレコードを選択する必要がある場合にのみ表示されます。
予行演習	ジョブをテストとして実行します。このジョブの実行時に影響を受けるレコードはありません。結果は、ジョブが実行された場合のように [サマリー] フィールドに表示されます。 **注:** ロールバックを使用してデータプライバシージョブを構成するときは、[予行演習] をオフにする必要があります。詳細については、「データプライバシージョブのロールバック」を参照してください。
状態	データプライバシージョブのステータス：

自動翻訳

フィールド	説明
	○ 完了:ジョブが正常に完了しました。 ○ スケジュールの準備ができました：新しいジョブのデフォルトステータス。 ○ ロールバック中：ジョブは匿名化をロールバックするように設定されています。 ○ ロールバック完了：匿名化ジョブのロールバックが正常に完了しました。 ○ 完了 (エラーあり):ジョブは完了しましたが、エラーがあります。詳細については、「データプライバシージョブログ」を参照してください。 ○ エラー:ジョブが完了しず、エラーが発生しました。詳細については、「データプライバシージョブログ」を参照してください。 読み取り専用フィールド。
推定レコード数	このジョブが影響するレコードの推定数。読み取り専用フィールド。
サマリー	実行時にジョブの結果を表示する読み取り専用フィールド。
期間開始	このジョブを実行する期間の開始時間。ジョブは、このフィールドに入力された時間の後に実行されます。 有効な時間値は、24 時間表記に基づく世界標準時です。
期間終了	このジョブを実行する期間の終了時間。ジョブは、このフィールドに入力された時間まで実行されます。ジョブがまだ完了していない場合、ジョブは一時停止し、次の期間の開始時に再開されます。終了時間は開始時間よりも後でなければなりません 有効な時間値は、24 時間表記に基づく世界標準時です。

5. フォームヘッダーを右クリックし、コンテキストメニューから [保存] を選択します。
レコードを保存すると、[ジョブをスケジュール] ボタンと [ジョブを削除] ボタンが表示されます。
6. [ジョブをスケジュール] をクリックしてジョブを実行します。
[期間開始] と [期間終了] フィールドで選択した時間の間でジョブが実行されます。ジョブが期間開始と期間終了の間に完了しなかった場合、ジョブは次の期間の開始時に続行されます。

   注: [予行演習] が選択されている場合でも、ジョブは 1 回のみ実行できます。同じジョブを再度実行するには、同じフィールド値を使用してデータプライバシージョブを作成します。

7. オプション: 次のいずれかの機能を選択します。

- ジョブのキャンセル：データプライバシージョブをキャンセルします。
- 一時停止：ロールバックが選択されている場合、ジョブを一時停止し、記録をロールバックします。ロールバックコンテキストの 3 日間の有効期限が過ぎると、警告メッセージが表示されます。詳細については、「データプライバシージョブのロールバック」を参照してください。
- 再開：一時停止したジョブを再開します。一時停止した場合、再開されたジョブのロールバックはサポートされません。ジョブをキャンセルし、Data Privacy ジョブを作成します。記録では、期限切れになっていないロールバックコンテキストが使用されます。

Data Privacy ジョブのロールバック

変更をロールバックできるように、ジョブやスクリプトなどのアクションでのデータベースの変更が取得されます。人為的なミスによって誤ったユーザー情報が匿名化された場合に、Data Privacy ジョブをロールバックします。ロールバックでは、データプライバシージョブのデータの匿名化解除が行われます。

概要

- ロールバックは、新しい RollbackType *REDACT* の RollbackContext の構成済み有効期間ごとに数日間に制限されています。Data Privacy ジョブに関連付けられた RollbackContext の有効期限が切れると、そのジョブではロールバック機能を使用できなくなります。
  - デフォルトでは、匿名化解除からのロールバックコンテキストは 3 日間保存されます。
  - デフォルトの有効期限は、新しい **RollbackType** の **RollbackContext** *REDACT.* において、データプライバシーアドミニストレーターが 1 より大きい値に設定できます。Glide システムプロパティ `glide.rollback.expiration_days_redact` で値を設定します。「ロールバックと削除の復旧」を参照してください。

    システムプロパティの追加または作成の詳細については、「システムプロパティを追加する」を参照してください。

    参照してください。
- ロールバックは、[完了]、[キャンセル]、または [エラー] ステータスのデータプライバシージョブで使用できます。
- ロールバックのサポートがオンになっているデータプライバシーポリシーで構成された sys_user 匿名化解除ジョブが成功するたびに、ロールバックコンテキストが作成されます。Data Privacy ジョブごとに設定できるロールバックコンテキストは 1 つだけです。

データプライバシージョブのロールバック

本番インスタンスの匿名化されたデータを使用する非本番インスタンスのデータプライバシージョブを、データクラスまたはユーザージョブの匿名化前の状態にロールバックします。

始める前に
必要なロール：data_privacy_admin または data_privacy_processor、および admin

手順

1. **data_privacy_processor** または **data_privacy_admin** ロールに昇格させます。
   ロールの昇格の詳細については、「特権ロールへの昇格」を参照してください。
2. 移動先 すべて > システムセキュリティ > データプライバシー > データプライバシージョブ.

   注: 事前に、ロールバックをサポートするデータプライバシーポリシー構成を作成する必要があります。Data Privacy ポリシーの作成。
3. データプライバシージョブを作成し、ロールバックをサポートするプライバシー構成を選択します。

「データプライバシージョブの構成」を参照してください。

4. データ匿名化のためのジョブをスケジュールします。
   - ジョブが実行されると、選択した構成でデータが匿名化されます。
   - 有効期限を知らせるメッセージがジョブに表示されます。有効期限内であれば、ジョブをロールバックできます。

5. **data_privacy_processor** ロールに昇格させます。
6. ロールバックするデータプライバシージョブを開きます。
7. [ロールバック] を選択して、データの匿名化を解除します。

データプライバシークローン

顧客データはソースからターゲットインスタンスに (通常は本番環境から非本番環境に) クローン作成されるため、機密データはターゲットインスタンスで匿名化されます。

重要: このセクションは、データプライバシー (従来) 用です。最新のストアバージョンについては、 データプライバシー を参照してください。

データプライバシーアドミニストレーターがクローン作成後のポリシーを構成します。ターゲットインスタンスでクローン後スクリプトが完了した後は、ユーザーには匿名化されたデータが表示され、元のデータにアクセスできなくなります。データプライバシーアドミニストレーターは、クローン作成時にターゲットインスタンスに適用する匿名化ポリシーを構成して、ターゲットインスタンスに元の機密データが含まれないようにすることができます。 実行される他のポリシーの中でのこのポリシーの順序を指定します。

注: データプライバシーのクローン作成は、自己ホストインスタンスでは使用できません。

データプライバシーのクローンには、次の追加の属性があります。

- データプライバシープラグインは、ターゲットインスタンスで実行されるクローン後スクリプトを作成します。
- PostClone 構成用に作成されたデータプライバシージョブは、同じテーブルを含んでいない場合、並行して実行できます。
- アプリケーション順序が上位になっている Postclone 構成用の Data Privacy ジョブは、上位のジョブが他の下位のジョブに関連するテーブルを含んでいない場合、下位のジョブの前に開始されることがあります。
- データプライバシープラグインを使用する場合、データプライバシーテーブルはデフォルトでクローンデータプリザーバーテーブルセットに含まれます。
- Data Privacy テーブル (dp_[table]) をクローン除外テーブルに追加しようとすると、そのテーブルを除外すべきではないことを示す警告が表示されます。

データプライバシークローン要求の構成

データプライバシークローン統合は、PostClone スクリプトを使用して構成され、ターゲット上で構成されたポリシー用のデータプライバシージョブを作成して実行します。スクリプトが実行された後は、ユーザーに匿名化されたデータが表示され、元のデータにアクセスできなくなります。

始める前に

必要なロール：data_privacy_clone_processor、data_privacy_admin、および admin

手順

1. ソースインスタンスでデータプライバシープラグイン (sn_dp_store_app) をアクティブ化します。
   プラグインは、カスタマーサービス & サポートのみがインストールできます。
   データプライバシー PostClone スクリプトがインストールされます。
2. **data_privacy_admin** ロールに昇格させます。
   ロールの昇格の詳細については、「特権ロールへの昇格」を参照してください。
3. 移動先 システムセキュリティ > データプライバシー > プライバシーポリシー構成.
4. プライバシーポリシー構成を作成します。
   [データクラスのすべてに適用] 選択し、[クローン作成時に適用] を選択します。詳細については、「Data Privacy ポリシーの作成」を参照してください。
5. データプライバシー構成をバックアップします。
6. Data Privacy アドミニストレーターとして、クローン要求を送信します。

結果
データプライバシー PostClone スクリプトはターゲットインスタンスで実行され、PostClone スクリプトはターゲットインスタンス上にデータプライバシー連携ジョブレコードを作成します。連携ジョブは、ターゲットインスタンスで、各クローン後ポリシー用のデータプライバシージョブを作成してアプリケーションの順序で実行します。 バックアップソースはターゲットインスタンスにクローン作成されます。データプライバシー PostClone スクリプトは、ターゲット上で構成されたポリシー用のデータプライバシージョブを作成して実行します。

昇格されたデータプライバシークローンプロセッサーは、ターゲットインスタンスにログオンし、dp_federated_job.list および dp_job.list でクローン後の連携ジョブのステータスを監視できます。

データプライバシージョブログ

データプライバシージョブのエラーを確認します。

始める前に
重要: このセクションは、データプライバシー (従来) 用です。最新のストアバージョンについては、 データプライバシー を参照してください。

必要なロール:data_privacy_admin、data_privacy_clone_processor、data_privacy_processor、data_privacy_auditor

このタスクについて
データプライバシージョブログを使用して、失敗したディスカバリー、分類、および匿名化ジョブを確認します。

手順

1. 移動先 システムセキュリティ > データプライバシー (従来) > データプライバシージョブログ.
2. テーブルをレビュー

データプライバシージョブログテーブル

ラベル	説明
作成日時	ログエントリの作成者または対象
レベル	エラーの重大度

ラベル	説明
メッセージ	エラーの原因の説明
ソース	エラーのプラグインソース
タイプ	発生したエラーのタイプ
コード	発生したエラーコード
ジョブテーブル名	ジョブが配置されているテーブル
ジョブ ID	ジョブ ID
ターゲットテーブル名	ジョブを実行したテーブル
ターゲットテーブル列	ジョブが実行された列
ターゲットテーブルレコード	実行されたジョブを記録

データディスカバリー

データディスカバリー を使用して、クレジットカード情報、メール、社会保障番号などのインスタンス内の機密データを識別します。

データディスカバリーの詳細

データディスカバリーについて学習します。

データディスカバリーの構成

データディスカバリーの構成に関するヘルプを取得します。

自動翻訳

データタディスカバリーのロール

データタディスカバリーの
ロールについて説明します。

データタディスカバリー結果

データタディスカバリーの結果を確認します。

自動翻訳

データタディスカバリー (クラシック) の詳細

データタディスカバリー を使用して、クレジットカード情報、メール、社会保障番号などのインスタンス内の機密データを識別します。

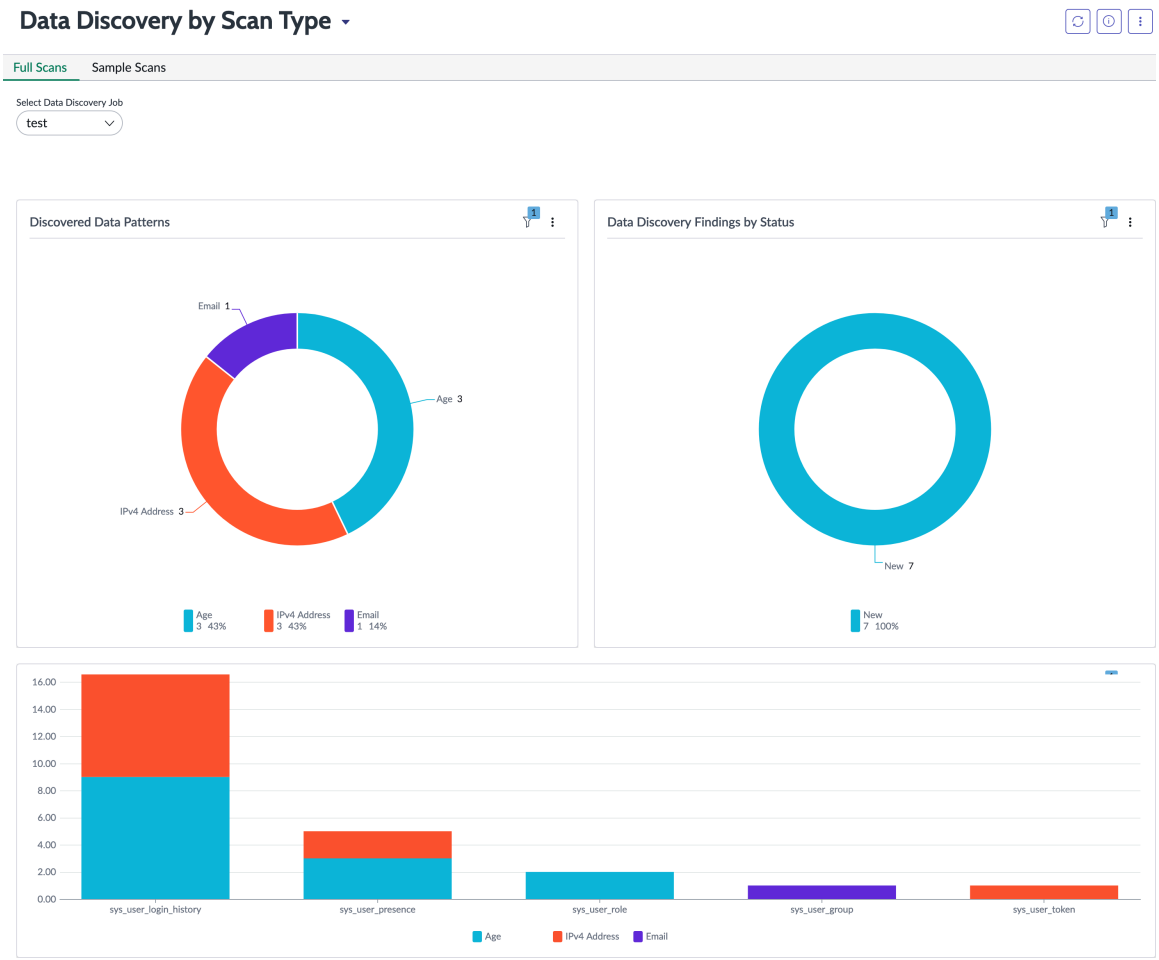
データタディスカバリー は、一連のテーブルでユーザー定義のジョブセットを実行します。ジョブはデータタディスカバリー ダッシュボードでレビューするために機密情報を検索してレポートします。スケジュール済みジョブは、実行時にすべてのアクティブなデータパターンとターゲットテーブルを自動的に使用します。

データタディスカバリー には、データへのさまざまなアクセスレベルで事前に割り当てられたロールも含まれています。

データタディスカバリー ダッシュボードへのアクセス

移動先 すべて > システムセキュリティ > データタディスカバリー (クラシック) > ダッシュボード をクリックして データタディスカバリー ダッシュボードを表示し、現在のジョブ結果をレビューします。

データディスカバリー ダッシュボード



検出されたデータパターン **(Discovered Data Patterns)**

ヒント: レコードを選択して詳細情報を表示し、検果を元にさらにアクションを実行します。詳細については、「データディスカバリー のジョブ結果」を参照してください。

[検出されたデータパターン (Discovered Data Patterns)] チャートには、現在検出されているデータパターンの数と割合が表示されます。さらにアクションを実行するセクションを選択します。

ステータス別のデータディスカバリー結果

[データディスカバリー結果 (Data Discovery Findings)] チャートでは、現在検出されているデータパターンの数と割合がステータス別に表示されます。

検出されたテーブル **(Discovered Table)**

[検出されたテーブル (Discovered tables)] では、現在検出されているデータが含まれているテーブルとそのデータパターンが強調表示されます。

検出されたデータを含む列

[検出されたデータを含む列 (Columns with Discovered Data)] セクションでは、現在検出されているデータが列別に表示されます。

データディスカバリー のアクティブ化

このアプリケーションは、データディスカバリー および関連する ServiceNow® Store アプリケーションとプラグインをインストールします (まだインストールされていない場合)。

始める前に

データディスカバリーでは、他の ServiceNow AI Platform とは別のサブスクリプションが必要です。

サブスクリプションを購入するには、ServiceNow アカウントマネージャーにお問い合わせください。サブスクリプションを購入すると、特定のプラグインが自動的にアクティブになります。有料プラグインが自動的にアクティブになっていない場合は、インスタンスの [すべてのアプリケーション] リストから手動でアクティブ化できます。

注:

サブスクリプションを購入する前に、Now Support サービスカタログ から非本番インスタンスを要求することによって、その機能を課金なしに評価できます。

データディスカバリーストアアプリをインストールすると、データディスカバリー API プラグインが自動的にインストールされます

必要なロール：admin

このタスクについて

次のアイテムがデータディスカバリーとともにインストールされます。

- ロール
- テーブル

インストールされるロールとテーブルの詳細については、「データディスカバリー ロール」および「デフォルトデータパターン」を参照してください。

手順

1. 移動先 すべて > システムアプリケーション > 利用可能なすべてのアプリケーション > すべて.
2. フィルター条件と検索バーを使用して、データディスカバリープラグイン (sn_data_discovery) を検索します。

   名前または ID でプラグインを検索できます。プラグインが見つからない場合は、ServiceNow 担当者から要求する必要があります。

3. [インストール] を選択して、インストールプロセスを開始します。

   注: ドメインセパレーションと代理アドミンがインスタンスで有効になっている場合、管理ユーザーはグローバルドメインに含まれている必要があります。それ以外の場合、次のエラーが表示されます： 「別の操作が実行されているため、アプリケーションのインストールは利用できません： <プラグイン名> のプラグインの有効化 (Application installation is unavailable because another operation is running: Plugin Activation for <plugin name>)」

   インストールが完了するとメッセージが表示されます。プラグインとともにインストールされるコンポーネントの詳細については、「 アプリケーションとともにインストールされているコンポーネントの検索 」を参照してください。

データディスカバリー の結果ページでデータを分類する

データディスカバリー の結果を通じて成功したすべてのジョブで見つかった機密データを分類します。

始める前に
必要なロール：data_discovery_admin

結果データを分類された状態で表示するには、ジョブの実行が正常に完了する必要があります。

このタスクについて
データディスカバリー の結果ページには、ジョブによって検出されたすべてのデータエントリが表示されます。結果ページでは、エントリを確認して分類に関連付けることができます。

手順

1. 移動先 システムセキュリティ > データディスカバリー (クラシック) > データディスカバリー結果.
2. 分類するエントリを選択します。
3. データディスカバリー の結果リストで、[分類] を選択します。
4. テーブルエントリに関連付けるデータクラスを選択します。
5. [分類] を選択します。

データディスカバリー ジョブ

データディスカバリー では、ユーザー定義のデータパターンとターゲットテーブルを使用して、ターゲット情報をレビューします。

データディスカバリー では、まず [データディスカバリージョブ] セクションでジョブをスケジュールします。スケジュール済みジョブが実行されると、現在のターゲットテーブルですべてのアクティブなデータパターンが検索されます。データディスカバリー ジョブの作成と構成の詳細については、「データディスカバリー ジョブの構成」を参照してください。

注: サンプルスキャンジョブは、パターンごとにテーブルあたり最大 10,000 レコードをスキャンします。

警告: ジョブが短時間に 2 回実行されると、2 回目の実行でスキャンされた行数として 0 が返されることがあります。

データディスカバリー データパターン

[データパターン **(Data Patterns)**] セクションには、アクティブと非アクティブの両方の現在のすべてのデータパターンが表示されます。データパターンは、機密データの照合に使用される正規表現です。データパターンにアクセスして構成する方法の詳細については、「データディスカバリー パターンの構成」を参照してください。[アクティブなデータパターン **(Active Data Patterns)**] セクションには、アクティブなパターンのみが表示されます。非アクティブなパターンは表示されません。

ターゲットテーブル

[ターゲットテーブル] セクションには、ジョブで処理できるすべてのテーブルが一覧表示されます。ターゲットテーブルにアクセスするには、[ターゲットテーブル] を選択します。

次のテーブルはサポートされていません。

- sr
- sysx
- v
- sh
- syslog

- ua
- usageanalytics
- ecc
- clone
- jrobin
- pa
- sla_repair_log
- scan
- gcf
- fm_log
- log
- np$
- sn_data_discovery
- dp_configuration
- dp_federated_job
- dp_field_technique
- dp_job
- dp_job_summary
- dp_primary_reference
- dp_technique
- data_classification
- m2m_dictionary_dataclass

粒度の構成

[粒度の構成 (Granular Configuration)] ツールを使用すると、テーブル内の機密データを検出するための、より限定的で繊細な制御が可能になります。ただし、従来のデータディスカバリージョブと比較すると、その操作には重要な違いがあります。粒度の構成では、テーブルの指定された列のみを検出のためにスキャンし、検出結果をレコード別に報告します。粒度の構成のスキャンによる検出結果 (粒度の検出結果) は、レコードごとにアクションが指定され、data_privacy_admin ロールが匿名化されている必要があります。

データディスカバリー ジョブの構成

データディスカバリー ジョブを構成し、進行中のジョブのステータスを確認します。データディスカバリー ジョブは、ターゲットテーブルでパターンが実行されるタイミングを定義します。

始める前に
必要なロール：data_discovery_admin

手順

1. 移動先 システムセキュリティ > データディスカバリー (クラシック) > データディスカバリージョブ.
2. データディスカバリー ジョブリストで、[新規] を選択します。

3. データディスカバリー ジョブフォームで、フィールドに入力します。

データディスカバリー ジョブフィールド

フィールド	説明
名前	ジョブの名前。
説明	ジョブの説明。
スキャンタイプ	スキャンするエントリの数。可能なステータスは次のとおりです。 ○ サンプル：10,000 エントリをスキャンします。 ○ 増分:新規または変更されたレコードの機密データパターンをスキャンします。このスキャンタイプは繰り返し発生しています。 ○ 完全：すべてのエントリをスキャンします。
ポリシー	このジョブで使用するデータディスカバリーポリシー。詳細については、「データディスカバリー ポリシー」を参照してください。
添付ファイルをスキャン	機密データのパターンがないか添付ファイルをスキャンします。.pdfまたは.docファイルに埋め込まれた画像は、機密データについてスキャンされません。 ⚠ **警告:** この機能を使用するには、制御された ServiceNow 環境にデータを送信する必要があります。有効にする前に、この機能を使用するために必要な追加の契約条件に同意したことを確認してください。
コンテキスト	スキャンされたパターン、ヒットしたターゲットテーブルエントリ、および経過時間に関するジョブの詳細。
状況	データディスカバリー ジョブのステータス。可能なステータスは次のとおりです。 ○ スケジュールの準備ができました：新しいジョブのデフォルトステータス。 ○ スケジュール済み：ジョブの実行がスケジュールされています。 ○ 対応中：ジョブはアクティブに実行されています。 ○ 完了：ジョブの実行が正常に終了しました。 ○ エラー：エラーのため、ジョブの実行が停止しました。 ○ キャンセル済み：ジョブがキャンセルされました。 ○ 一時停止：ジョブは一時停止されています。

フィールド	説明
サマリ	ジョブを実行した後の結果を表示します。
開始日	ジョブの開始日を設定します。
期間開始	このジョブを実行する期間の開始時間。ジョブは、このフィールドに入力された時間の後に実行されます。[期間開始] フィールドに入力された時間は、[期間終了] フィールドに入力された時間より前である必要があります。 注: 有効な時間値は、24 時間表記に基づく世界標準時です。
期間終了	このジョブを実行する期間の終了時間。ジョブは、このフィールドに入力された時間まで実行されます。ジョブがまだ完了していない場合、ジョブは一時停止し、次の期間の開始時に再開されます。[期間終了] フィールドに入力された時間は、[期間開始] フィールドに入力された時間より後である必要があります。 注: 有効な時間値は、24 時間表記に基づく世界標準時です。

4. [送信] を選択します。
   [ジョブをスケジュール] ボタンと [更新] ボタンが表示されます。
5. [ジョブをスケジュール] を選択してジョブを実行します。
   [期間開始] と [期間終了] フィールドで選択した時間の間でジョブが実行されます。ジョブが期間開始と期間終了の間に完了しなかった場合、ジョブは次の期間の開始時に続行されます。
6. オプション: 次のいずれかの機能を選択します。
   - ジョブのキャンセル：ジョブをキャンセルします。
   - 一時停止：ジョブを一時停止します。
   - 再開：一時停止したジョブを再開します。

データディスカバリージョブでの添付ファイルのスキャン

データディスカバリー の添付ファイルスキャンを使用すると、添付ファイル内の機密データをスキャン、検出、およびレポートすることができます。

多くの場合、社会保障番号 (SSN)、クレジット カード番号、その他の個人識別可能情報 (PII) などの機密データが添付ファイルに含まれています。 データディスカバリー ジョブの添付ファイルスキャンを使用すると、特定の添付ファイル内の機密データを検出してレポートできます。添付ファイルのスキャン機能は、スケジュールされたスキャンでのみ使用できます。

⚠ 警告: この機能を使用するには、制御された ServiceNow 環境にデータを送信する必要があります。添付ファイルのスキャン機能を有効にするには、アカウントとサポートチームにお問い合わせください

制限事項

- PDFおよびDOCファイルに埋め込まれた画像と添付ファイルは、機密データについてスキャンされません。
- ファイルは 10 MB 未満である必要があります。
- 次のファイルタイプがサポートされています。
  - PDF
  - ドキュメント(X)
  - TXT
  - XLS
  - CSV

データディスカバリー パターンの構成

データディスカバリー パターンを構成し、現在のパターンを確認します。データディスカバリー パターンは、データをターゲットテーブルと照合するために使用する正規表現を定義します。

始める前に

必要なロール：data_discovery_admin

手順

1. 移動先 システムセキュリティ > データディスカバリー > すべてのデータパターン.
2. データディスカバリー パターンリストで、[新規] を選択します。
3. データディスカバリー ジョブフィールドフォームで、フィールドに入力します。

データディスカバリー ジョブフィールド

フィールド	説明
内部スコープ	パターンのスコープ
説明	ジョブの説明
名前	データパターンの名前
アプリケーション	パターンのアプリケーションスコープ
式	データパターンを検出するために使用される正規表現 **注:** 表現の長さは 1000 文字未満にする必要があります。
キーワード (オプション)	式について検索する特定の単語 (またはカンマ区切りの単語)。キーワードの近接性に基づいて使用する必要があります。 **注:** キーワードを使用して、パターンの追加コンテキストを検索できます。たとえば、キーワードを使用して、生年月日と雇用日を区別するのに役立ちます (MM/DD/YY 形式が同じ場合)。

フィールド	説明
キーワードの近接性 (オプション)	キーワード検索時の式との近接性。キーワードとともに使用する必要があります 注: デフォルトは 30、上限値は 64 です
プライバシー手法構成	パターンに使用されるプライバシー手法
合成値	パターンに代入された値のリスト
タイプ	パターンのタイプ ○ ローカル:パターンは正規表現ベースです ○ モデル:AI/ML サービスを使用

4. **[Submit** (送信)] を選択します。
   - [テスト] ボタンを使用すると、データパターンリストを送信する前に正規表現をテストできます。

スケジュール済みジョブで使用するには、データパターンをアクティブとして設定する必要があります。

5. 移動先 システムセキュリティ > データディスカバリー > アクティブなデータパターン.
6. データディスカバリー のアクティブなパターンリストで、[編集] を選択します。
7. [利用可能なリスト **(Available Lists)]** からパターンリストを選択し、[選択したリスト **(Selected Lists)]** に移動します。

デフォルトデータパターン

データディスカバリー に含まれるデフォルトデータパターンの正規表現を確認します。これらのデフォルトデータパターンを使用して、さらに分類するためにテーブルエントリをフィルタリングできます。

データ検出に使用できるデフォルトのパターンは次のとおりです。

名前	説明	正規表現	キーワード	例
経過時間	0 ～ 99 歳の人の年齢	\b([0-9]|[1-9][0-9]|1[012][0-9])\b	経過時間	一致 24 一致しない • 103 • -2
生年月日	DD/MM/YYYY 形式の生年月日	\b[0-3]?[0-9]/[0-3]?[0-9]/(?:[0-9]{2})?[0-9]{2}\b	dob、誕生日、生年月日	一致 • 06/18/2012 • 1/1/19
電子メール	標準メールアドレス	\b[\w!#$%&'*+/=?`{\|}~^-]+(?:\.[\w!#$%&'*+/=?`{|}~^-]+)*@(?:		一致 • johndoe@emailserver.com • historyprofessor@college

名前	説明	正規表現	キーワード	例
		[a-zA-Z0-9-]+\.)+[a-zA-Z]{2,6}\b		一致しない • notanemail.com • bademail@.org
車両識別番号	車両識別番号 (VIN)	\b[A-HJ-NPR-Z0-9]{17}\b		一致 AHUYA31581L000000
IP アドレス	標準 IP アドレス	4 桁の IP \b(?:(?:25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)\.){3}(?:25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)\b 6 桁の IP \b(((?:[0-9A-Fa-f]{1,4}(?::[0-9A-Fa-f]{1,4})*)?)::((?:[0-9A-Fa-f]{1,4}(?::[0-9A-Fa-f]{1,4})*)?))|((?:[0-9a-fA-F]{1,4}:){7}[0-9a-fA-F]{1,4})\b		一致する 4 桁の IP 102.28.46.103 一致する 6 桁の IP 914b:d45a:61ea:6346:59b

名前	説明	正規表現	キーワード	例
クレジットカード - Visa	Visa クレジットカード番号	\b4[0-9]{12}(?:[0-9]{3})?\b		一致 4444434342424242
クレジットカード - American Express	American Express クレジットカード番号	\b3[47][0-9]{13}\b		一致 378225246366005
クレジットカード - Mastercard	Mastercard クレジットカード番号	\b(?:5[1-5][0-9]{2}|222[1-9]|22[3-9][0-9]|2[3-6][0-9]{2}|27[01][0-9]|2720)[0-9]{12}\b		一致 5555444455554444
クレジットカード - Diners Club	Diners Club クレジットカード番号	\b3(?:0[0-5]|[68][0-9])[0-9]{11}\b		一致 3056930009020004
クレジットカード - Discover	Discover クレジットカード番号	\b6(?:011|5[0-9]{2})[0-9]{12}\b		一致 6011025690875424
クレジットカード - CCV	クレジットカードセキュリティ番号	\b[0-9]{3,4}\b	CVV、検証コード、セキュリティコード	一致 124
クレジットカード - 有効期限	MM/YYYY 形式のクレジットカードの有効期限	\b(([1-9])|(0[1-9]|1[0-2]))\/?([0-9]{4}|[0-9]{2})\b	有効期限、exp	一致 • 02/2027 • 04/23 一致しない 03/9
米国 - 社会保障番号	米国市民社会保障番号	\b(?!666|000|9\d{2})\d{3}-(?!00)\d{2}-(?!0{4})\d{4}\b		一致 001-22-111
米国 - 電話番号	米国電話番号 ⚠ 警告: 米国国番号は使用しません。	\b\(?([0-9]{3})\)?[-.]?([0-9]{3})[-.]?([0-9]{4})\b		一致 2065550199 一致しない 1 555 238 0199
米国 - パスポート番号	9 桁の米国パスポート番号	\b[a-zA-Z0-9]\\d{8}\b		一致 770022122
米国 - 納税者 ID	米国納税者 ID 番号	\b(9\d{2})([\-]?)([7]\d|8[0-8])([\-]?)(\d{4})\b		一致 927 70 5828

自動翻訳

名前	説明	正規表現	キーワード	例
米国 - カリフォルニア州運転免許証番号	米国カリフォルニア州の運転免許証番号	\b[a-zA-Z]\d{7}\b		一致 A0002144
米国 - 銀行ルーティング番号	米国銀行ルーティング (ABA) 番号	\b((0[0-9])|(1[0-2])|(2[1-9])|(3[0-2])|(6[1-9])|(7[0-2])|80)([0-9]{7})\b		一致 125210305

データディスカバリー ターゲットテーブルの構成

データディスカバリー ジョブで使用するターゲットテーブルを追加します。ターゲットテーブルでのみデータパターンがスキャンされます。

始める前に
必要なロール：data_discovery_admin

このタスクについて
データディスカバリー ジョブが実行されると、すべてのアクティブなパターンを持つすべてのターゲットテーブルに対して実行されます。

手順

1. 選択 システムセキュリティ > データディスカバリー (クラシック) > ターゲットテーブル.
2. **[New** (新規)] を選択します。
3. [テーブル名] フィールドでターゲットテーブルを選択します。
4. [送信] を選択します。

データディスカバリーの並列ジョブのアクティブ化

並列ジョブを使用して、データディスカバリージョブの実行時間を短縮します。

始める前に
必要なロール：admin

手順
デフォルトでは、データディスカバリーはシングルスレッドを使用してジョブを実行します。データディスカバリーのフルスキャンのために複数のスレッドをアクティブ化できます。

1. ナビゲーションフィルターに、sys_properties.list と入力します。
2. プロパティ com.glide.data_discovery.max_concurrent_item_workers を作成します。
3. [タイプ] が integer であることを確認します。
4. **重要:** ノードあたりのワーカー数は、少なめ (2 または 3) から始めることをお勧めします。

   [値] フィールドに並列ジョブの数を 5 までの範囲で入力します。

データディスカバリー ロール

データディスカバリー ロールをアサインして、特定のデータタイプへのユーザーアクセスを制限できます。

データディスカバリーアドミニストレーター **[sn_data_discovery.data_discovery_admin]**

データパターンと関連ジョブを表示、作成、変更します。

- データパターン：
  - 作成
  - 読み取り
  - 更新
  - 削除
- アクティブなデータパターン：
  - 削除
  - 読み取り
- ジョブ：
  - 作成
  - 読み取り
  - 更新
  - 削除
  - スケジュール
  - 一時停止
  - 再開
  - キャンセル
- ターゲットテーブル：
  - 作成
  - 読み取り
  - 書き込み
- 粒度の構成 (Granular configuration)：
  - 作成
  - 読み取り
  - 更新
  - 削除
- 粒度の検出結果 (Granular findings)：
  - 作成
  - 読み込み
  - アクション

  注: このロールを持つユーザーは、無視アクションのみ実行できます。

ロールを含む

ロール内に含まれるロールのリスト。

- data_classification_auditor
- data_classification_admin
- sn_data_discovery.data_discovery_api_processor

グループ

このロールがデフォルトでアサインされているグループのリスト。

なし。

特別な考慮事項

アドミンには、製品のインストール時にこのロールが自動的にアサインされます。

注: より分化したロールが利用可能な場合は、アドミンロールを付与しないでください。

データディスカバリー監査人 **[sn_data_discovery.data_discovery_auditor]**

データパターンとターゲットテーブルを読み取ります。

- データパターンの読み取り
- アクティブなデータパターンの読み取り
- データディスカバリー ジョブの読み取り
- ターゲットテーブルの読み取り
- 粒度の構成のレビュー
- 粒度の検出結果のレビュー

ロールを含む

ロール内に含まれるロールのリスト。

なし。

グループ

このロールがデフォルトでアサインされているグループのリスト。

なし。

特別な考慮事項

なし。

データ分類アドミニストレーター **[data_classification_admin]**

有効になっている特定のパターン結果が分類されたときに、データパターン、ディスカバリージョブ、data_classification テーブルを読み取ります。

ロールを含む

データ分類ロールには、data_classification_auditor ロール内に含まれるロールのリストが含まれています。

グループ

このロールがデフォルトでアサインされているグループのリスト。

なし。

特別な考慮事項

注: データ分類アドミンロールの詳細については、「データ分類 プラグインのデモデータをインストールする」を参照してください。

データディスカバリー のジョブ結果

[データディスカバリー の結果] ページには、ジョブによって検出されたデータの詳細が表示されます。[結果 (Findings)] ページを使用してジョブの結果を確認し、データの分類を開始できます。

[データディスカバリー の結果] ページには、完了したジョブの後に次の詳細が表示されます。

列	説明
辞書エントリ	データが格納されていたターゲットテーブルの列
テーブル	データが見つかったターゲットテーブル
データパターン (Data Pattern)	データの検索に使用されるデータパターン
データパターン一致数 (Data Pattern Match Count)	データパターンに一致するデータエントリの数
合計行スキャン数 (Total Row Scan Count)	ジョブ中にスキャンされた行数
一致する行の割合 (Percentage of Matching Rows)	データパターンに一致するターゲットテーブルの行の割合
データディスカバリー ジョブ	ターゲットテーブルで使用されるジョブ
ステータス	エントリのステータス

データディスカバリー ジョブを実行すると、結果のステータスは [新規] になります。アクションが必要ない場合は、データをそのままにしておくことができます。これにより、ステータスが自動的に [無視] に設定されます。それ以外の場合は、ユーザー定義データ分類の作成などによりデータを分類し、データ分類 ツールを使用してデータの匿名化を準備することができます。

粒度の検出結果

ユーザーは、[粒度の検出結果の追跡 **(Track Granular Findings)**] アクションを選択し、粒度の検出結果 ページを使用して、検出された特定のレコードに対してアクションを実行できます。詳細については、「粒度の検出結果」を参照してください。

利用可能な保護

[利用可能な保護 **(Available Protections)**] ボタンを選択すると、エントリ制御サービス、ステータス、メモ、および最後のチェックインを確認できます。Xanadu の時点では、この機能はフィールド暗号化のみをサポートしています。

保護サービスは、ダッシュボードからアクセスすることもできます。ダッシュボードの詳細については、「データディスカバリー」を参照してください。

結果を統合

ジョブ結果は、[結果を統合] ボタンを使用して統合できます。2 つの個別のジョブ間で共有されるジョブ結果は、最後に実行されたジョブに統合されます。

注: 終了したフルスキャンジョブのみを統合できます

データディスカバリー のジョブ結果ページでデータを分類する

ジョブの結果ページから直接 データディスカバリー のデータを分類します。

始める前に
必要なロール：data_discovery_admin および admin

結果データを分類された状態で表示するには、ジョブの実行が正常に完了する必要があります。

手順

1. 検索項目 システムセキュリティ > データディスカバリー > データディスカバリージョブ.
2. 分類するエントリを選択します。
3. データディスカバリー の結果リストで、[分類] を選択します。
4. テーブルエントリに関連付けるデータ分類を選択します。
5. [分類] を選択します。

データディスカバリー でサポートされているデータタイプ

データディスカバリーを使用するときにサポートされているフィールドタイプを確認します。

注: 分類されたすべてのフィールドタイプがデータディスカバリーに使用できるわけではありません。

次の表に示すように、一部の高リスクフィールドタイプはデフォルトでオフになっています。フィールドの詳細については、「フィールドタイプ」を参照してください。

データディスカバリーでサポートされているフィールドタイプ

フィールドタイプ	デフォルトで利用可能
audio	いいえ
状態	いいえ
condition_string	いいえ
currency	はい
decimal	はい
due_date	はい
浮動小数点数	はい
glide_date	はい
glide_date_time	はい
glide_duration	いいえ
glide_time	はい
HTML	いいえ

フィールドタイプ	デフォルトで利用可能
アイコン	いいえ
integer	はい
ip_addr	いいえ
ip_address	いいえ
journal	はい
journal_input	はい
journal_list	はい
longint	はい
name_values	いいえ
percent_complete	いいえ
phone_number_e164	はい
price	はい
string	はい
string_full_utf8	はい
Translation_html	いいえ
translated_text	いいえ
url	いいえ
user_image	いいえ
video	いいえ
wiki_text	いいえ

粒度の構成によるスキャン

粒度スキャンを使用して、検出のために特定のテーブル列をスキャンできます。従来のデータディスカバリージョブはテーブル全体をスキャンしてデータを検出しますが、粒度スキャンはテーブルの特定の列を対象とするため、検出プロセスをより詳細に制御できます。

始める前に

必要なロール：admin、sn_data_discovery_admin、data_privacy_admin

手順

1. 移動先 すべて > システムセキュリティ > データディスカバリー (クラシック) > 粒度の構成.
2. [粒度の構成 **(Granular Configurations)]** リストで、[新規] を選択します。
3. [粒度の構成 **(Granular Configuration)]** フィールドフォームで、フィールドに入力します。

[粒度の構成 **(Granular Configuration)]** フィールド

フィールド	説明
辞書エントリー	スキャンする列
テーブル	スキャンするテーブル

フィールド	説明
列ラベル	スキャンする列のラベル
スキャン開始点 (Scan start point)	機密データは、スキャン開始点の当日以降についてのみ検出されます。 **注:** スキャン開始点を空のままにすると、列のすべてのエントリがスキャンされます。スキャンの開始点が変更されると、スキャンは新しく構成されたタイムスタンプから開始されるようにリセットされます。
アクティブ	オンにすると、粒度の構成が有効になります。

4. [送信] を選択します。
   拡張可能なテーブルの子を選択して構成できます。
5. オプション: 親テーブルの [粒度の構成 (Granular Configuration)] を選択し、[子テーブルを選択] を選択して、スキャンする子テーブルを選択します。

結果
ターゲットテーブルの特定の列が検出のためにスキャンされます。粒度スキャンでは、[アクティブなデータパターン] のデータパターンを使用します。次で検出結果を確認し、アクションを実行できます。 すべて > システムセキュリティ > データディスカバリー (クラシック) > 粒度の検出結果. 詳細については、「粒度の検出結果」を参照してください。

粒度の検出結果

粒度の検出結果は、[粒度の検出結果 (Granular Findings)] ツールを使用してレビューできます。

粒度の検出結果のレビュー

[粒度の検出結果 **(Granular Findings)]**は、[粒度の構成 (Granular Configurations)] から最大 500 件の検出結果をレビューするために使用されます。検出結果が 500 件に達すると、アクションが実行されるまで、[粒度の構成 (Granular Configurations)] のスキャンは一時停止されます。

列ラベル	説明
レコード	検出されたレコード。
テーブル	レコードの親テーブル。
列	機密情報が検出されたレコードの列。
データパターン	レコードの検出に使用されるパターン。
アクション	検出結果に対して実行するアクション。選択して変更できます。 **重要:** レコードに対して匿名化アクションを実行するには、data_privacy_admin ロールが必要です。 レビュー

列ラベル	説明
	レコードはレビュー待ちです。これは、新しい粒度の検出にアサインされます。 無視 レコードに対してアクションは実行されません。 匿名化 レコードは匿名化されます。
ステータス	レコードのステータス。 新規 検出結果が最初に報告されたときにアサインされるステータス 処理済み ユーザーが選択したアクションが検出結果に正常に適用されたとき 注: 処理済みの検出結果は、[粒度の検出結果 (Granular Findings)] テーブルに 3 日間保存されてから削除されます。 手動レビュー ユーザーが選択したアクションの適用に失敗したとき 警告: 手動レビューの検出結果は、適切なアクションを実行した後にユーザーが削除する必要があります。

警告: 粒度の検出結果からトリガーされた匿名化では、ロールバックはサポートされていません。

コンテキストベースのディスカバリー

コンテキストベースのディスカバリーを使用すると、固定パターンに従わない機密データを検出できます。

データディスカバリーは、名前付きエンティティ認識 (NER) モデルの使用 名前、組織、国籍、所属政党などのデータの検出をサポートしています。モデルタイプのデータパターンでは、この機能を使用します。詳細については、データディスカバリー パターンの構成のパターンタイプを参照してください。

警告: この機能を有効にするには、顧客によるライセンスチェックが必要です。

AI モデルベースのディスカバリーは、デフォルトで 5 つのパターンをサポートしています。

コンテキストディスカバリーパターン

パターン	説明	テキスト例	一致するテキスト
ユーザー	個人の名前	ジョン・ドウがコーヒーを飲みにやって来た。	John Doe
国籍、宗教または政治団体(NRP)	国家的、宗教的、政治的所属。	彼はアメリカ人でした。	アメリカ人
場所	住所または場所情報	彼はニューヨークにいました	ニューヨーク
Date_Time	日付と時刻の情報	彼は今日の9時30分に来ました。	本日 9:30
組織	組織名	彼は ServiceNow に勤務しています。	ServiceNow

注: AI モデルベースのディスカバリーは、テーブルとデータキットチャネルの リアルタイム匿名化 (RTA) で使用する英語のみをサポートしています。

データディスカバリー ポリシー

データディスカバリーポリシーを使用して特定のテーブルをスキャンし、列ベースのジョブを有効にします。

データディスカバリー ポリシーを使用すると、ジョブに含める必要がある特定のデータパターン、テーブル、および列の定義をきめ細かく制御できます。さらに、ポリシーを使用すると、1 つのグローバル構成ではなく、複数のディスカバリージョブ設定を同時に作成できます。データディスカバリーポリシーを確認するには、次の場所に移動します。 すべて > システムセキュリティ > データディスカバリー (クラシック) > データディスカバリーポリシー.

注: ポリシーに対して列が選択されていない場合、ターゲットテーブルのすべての列がスキャンされます

データディスカバリーポリシーの作成方法については、データディスカバリー ポリシーの作成を参照してください。

データディスカバリー ポリシーの作成

データディスカバリージョブをきめ細かく制御するためのデータディスカバリーポリシーを作成します。

始める前に

必要なロール：admin、sn_data_discovery_admin、data_privacy_admin

手順

1. 移動先 すべて > システムセキュリティ > データディスカバリー (クラシック) > データディスカバリーポリシー.
2. **[New** (新規)] ボタンをクリックします。
3. フォームに入力します。

データディスカバリーポリシー

ラベル	説明
名前	ポリシーの名前
ターゲット列	ポリシーで使用するターゲット列を選択します 注: 列が選択されていない場合、ターゲットテーブルのすべての列がスキャンされます
ターゲットテーブル	ポリシーで使用するターゲットテーブルを選択します
アクティブ	ポリシーがアクティブか非アクティブかを決定します 注: デフォルトはアクティブです
データパターン	ポリシーが使用するデータパターン。

4. **[Submit** (送信)] を選択します。

結果
データディスカバリー ジョブの作成時にポリシーを選択します。

データディスカバリー 店

ServiceNow インスタンス内の機密データの検出と追跡

データディスカバリーストア

データディスカバリーを使用して、インスタンス内の機密データを検出します。クレジットカード番号、電話番号、社会保障番号 (SSN) などの構造化された機密データをスキャンします。すぐに利用可能な一般的なデータパターンを使用してすぐに開始することも、独自のパターンを作成してより詳細なエクスペリエンスを作成することもできます。名前や住所などの非構造化でコンテキストベースの機密データの場合は、AI と機械学習を使用して識別に役立てることができます

データディスカバリー ストアの概要
: データディスカバリーメトリクスの概要と、ツールを使用するための開始点を取得します。

データディスカバリー 政策
: 現在のデータディスカバリーポリシーを確認して構成します。

データディスカバリー ソース
: ディスカバリージョブで現在使用されているデータパターンとターゲットテーブルを表示します

データディスカバリー スケジュール済みディスカバリー
: 詳細な列ディスカバリージョブを含むディスカバリージョブをスケジュールしてレビューします。

データディスカバリー ストアの概要

インスタンスの機密データを可視化し、機密データの損失や公開を防ぐためのセキュリティ対策の実装に向けて取り組みます。

[概要] タブは、重要なディスカバリーメトリクスと検出結果を表示するとともに、データディスカバリーを使用するための簡単な開始点を提供します。移動先 すべて > データディスカバリー > 概要 をクリックしてタブの使用を開始します。

ディスカバリーポリシーを作成し、ディスカバリージョブをスケジュールします

推奨アクションアイテムで データディスカバリー の使用を開始できます。[ポリシーを作成] を選択して、識別する必要がある機密データパターンとその処理方法を定義します。新しいポリシーの作成の詳細については、 データディスカバリー 政策 を参照してください。[ジョブをスケジュール] を選択して、インスタンスの機密情報をスキャンするデータディスカバリージョブをスケジュールします。新しいディスカバリージョブのスケジュールの詳細については、「 データディスカバリー スケジュール済みディスカバリー 」を参照してください。

データディスカバリーのサマリー

重要な データディスカバリー メトリクスを一目で確認できます。さらに、チャートを選択してさらにドリルダウンすることもできます。ディスカバリー結果のレビューの詳細については、「 検出結果のレビュー」を参照してください。

検出されたデータ

検出されたデータの分類と量を重ね合わせた棒グラフ。インスタンス内のさまざまなテーブルで検出された PII のインスタンスが検出された場所と数が、データパターンタイプ別に分類されてまとめられます。

ディスカバリーステータス

検出されたすべてのパターンの現在のステータスの割合 (レビュー待ちの新しい検出結果、分類済み、無視としてマークされたかどうか) を示すドーナツグラフ

検出されたデータと最近のジョブ

このセクションでは、新しく検出されたデータと最近のディスカバリージョブを確認できます。これらのテーブルとその内容の詳細については、 データディスカバリー スケジュール済みディスカバリー

.

データディスカバリー 政策

データディスカバリーポリシーを使用して、識別する必要がある機密データパターンとその処理方法を定義します。

データディスカバリー の [ポリシー] タブは、現在のディスカバリーポリシーをレビューします。[詳細を表示] ボタンを選択して、ポリシーの追加の詳細を表示します。3 つのドットを選択して、現在のポリシーのステータスを編集、削除、および切り替えます ⋮。新しいポリシーを作成するには、「 新しいポリシーを作成」を参照してください。

新しいポリシーを作成

テーブルのデータパターンのスキャンを開始する データディスカバリー ポリシーを作成します。

始める前に
必要なロール:discovery.admin

手順

1. 移動先 すべて > データディスカバリー > ポリシー.
2. [新しいポリシーを作成] ボタンを選択します。
   既存のポリシーを編集または削除するには、3 つの ⋮ アイコンを選択します。
3. ポップアップウィンドウのフィールドに入力します。

新しいディスカバリーポリシーフォーム

フィールド	説明
名前	ポリシーの一意の名前を入力します
データパターン	スキャンする機密データパターンを選択します。複数のデータパターンを選択できます 注: AI/ML モデルベースのデータパターンには追加のライセンスが必要であり、追加の契約条件が適用されます。
ユーザーテーブルと列を選択	スキャンするテーブルを確認 注: 選択したすべてのテーブル列がデフォルトでスキャンされます。選択を特定の列に絞り込むには、以下の手順を参照してください
選択されたテーブルと列	ポリシーに対して現在選択されているテーブルと列のリストを表示します

4. オプション: 矢印を選択します テーブルの特定の列を表示して選択するための > アイコン。
5. **[Save** (保存)] を選択します。

結果

デフォルトでは、新しいポリシーは [アクティブ] に設定されます。3 ⋮ アイコンを選択し、[ポリシーの非アクティブ化]/[ポリシーのアクティブ化] を選択してステータスを切り替えます。

データディスカバリー ソース

データディスカバリーで使用するデータパターンとスキャンするテーブルを作成して選択します。

データディスカバリーの [ソース] タブでは、すべてのデータパターンを確認し、スキャンに使用するパターンを設定して、スキャンするテーブルを選択できます。

すべてのパターン

[すべてのパターン] テーブルには、現在のすべてのパターンが一覧表示されます。非アクティブなパターンがこのリストに表示されることに注意してください。新しいデータパターンを作成するには、「 新しいデータパターンを作成 」を参照してください。

注: ディスカバリージョブで使用するには、パターンをアクティブにする必要があります。

[すべてのパターン] テーブル

ラベル	説明
名前	データパターンの名前
式	データパターンを検出するために使用される正規表現
キーワード	式について検索する特定の単語 (またはカンマ区切りの単語)。
キーワードの近接性	キーワード検索時の式との近接性。
プライバシー手法構成	パターンに使用されるプライバシー手法

アクティブなパターン

[アクティブなパターン] テーブルには、現在アクティブなすべてのデータパターンが一覧表示されます。パターンをアクティブに設定し、ディスカバリージョブで使用する方法については、「 アクティブなデータパターンを選択 」を参照してください。

アクティブなパターンテーブル

ラベル	説明
名前	データパターンの名前
式	データパターンを検出するために使用される正規表現
キーワード	式について検索する特定の単語 (またはカンマ区切りの単語)。
キーワードの近接性	キーワード検索時の式との近接性。
プライバシー手法構成	パターンに使用されるプライバシー手法

ターゲットテーブル

[ターゲットテーブル] テーブルには、ディスカバリージョブに対して現在選択されているすべてのテーブルが一覧表示されます。ディスカバリージョブで使用するテーブルを選択する ターゲットテーブルを選択 を参照してください。

ターゲットテーブルテーブル

ラベル	説明
テーブル名	テーブルの名前
表示名	テーブルの表示名 (レポートで使用)
アプリケーション	テーブルのアプリケーションスコープ。

新しいデータパターンを作成

新しい データディスカバリー ストアパターンを作成します。

始める前に
必要なロール:discovery.admin

手順

1. 移動先 すべて > データディスカバリー > ソース.
2. ナビゲーションペインで [すべてのパターン] を選択します。
3. [新規作成] ボタンを選択します。
4. フォームに入力します。

[新しいデータパターン **(New data pattern)]** フォーム

フィールド	説明
説明	パターンの説明。
名前	データパターンの名前
式	データパターンを検出するために使用される正規表現 注: 表現の長さは 1000 文字未満にする必要があります。
キーワード	式について検索する特定の単語 (またはカンマ区切りの単語)。キーワードの近接性に基づいて使用する必要があります。 注: キーワードを使用して、パターンの追加コンテキストを検索できます。たとえば、キーワードを使用して、生年月日と雇用日を区別するのに役立ちます (MM/DD/YY 形式が同じ場合)。
キーワードの近接性	キーワード検索時の式との近接性。キーワードとともに使用する必要があります 注: デフォルトは 30、上限値は 64 です
プライバシー手法構成	パターンに使用されるプライバシー手法
合成値	パターンに代入された値のリスト
タイプ	パターンのタイプ
アプリケーション	パターンのアプリケーションスコープ
スコープ	パターンのスコープ

5. オプション: 必要に応じて、[テスト] ボタンを選択して正規表現をテストします。
6. **[Submit** (送信)] を選択します。

結果
ディスカバリージョブで使用するには、新しいデータパターンをアクティブに設定する必要があります。詳細については、「アクティブなデータパターンを選択」を参照してください。

アクティブなデータパターンを選択

データディスカバリージョブに使用するアクティブなデータパターンを選択します。

始める前に
必要なロール:discovery.admin

手順

1. 移動先 すべて > データディスカバリー > ソース.
2. ナビゲーションペインで [アクティブなパターン] を選択します。
3. [編集] ボタンを選択します。
4. 有効にするパターンにチェックを入れると、ポップアップの右側に表示されます。
5. オプション: パターンを選択してドラッグし、並べ替えます。

   注:

   データパターンの順序によって、データパターンの匿名化に適用される匿名化手法の順序も決定されます。

6. [**Save**] ボタンをクリックします。

ターゲットテーブルを選択

ディスカバリージョブで使用するターゲットテーブルを選択します。

始める前に
必要なロール:discovery.admin

手順

1. 移動先 すべて > データディスカバリー > ソース.
2. ナビゲーションペインで [ターゲットテーブル] を選択します。
3. [編集] ボタンを選択します。
4. ターゲットとするテーブルにチェックを入れると、ポップアップの右側に表示されます。

   注: ターゲットテーブルは、 ポリシーで特に指定されていない限り、すべての列をスキャンします。

5. [**Save**] ボタンをクリックします。

結果
選択したテーブルは、 スケジュール済みディスカバリージョブのターゲットになります。

データディスカバリー スケジュール済みディスカバリー

インスタンスで機密情報をスキャンするデータディスカバリージョブを設定してスケジュールします。

ジョブのディスカバリー

ディスカバリージョブテーブルには、すべてのディスカバリージョブが一覧表示されます。新しいディスカバリージョブを作成する ディスカバリージョブを作成 を参照してください。

ディスカバリージョブテーブル

名前	ジョブの名前
説明	ジョブの説明
完了率 (%)	ジョブの完了率
スキャンタイプ	ジョブのスキャンタイプ。可能なステータスは次のとおりです。 • サンプル：10,000 エントリをスキャンします。 • 完全：すべてのエントリをスキャンします。
開始日	ジョブの開始日
更新日時	ジョブが最後に更新された日時。
状況	データディスカバリー ジョブのステータス。可能なステータスは次のとおりです。 • スケジュールの準備ができました：新しいジョブのデフォルトステータス。 • スケジュール済み：ジョブの実行がスケジュールされています。 • 対応中：ジョブはアクティブに実行されています。 • 完了：ジョブの実行が正常に終了しました。 • エラー：エラーのため、ジョブの実行が停止しました。 • キャンセル済み：ジョブがキャンセルされました。 • 一時停止：ジョブは一時停止されています。
実行	ジョブの実行がスケジュールされている頻度。

ディスカバリー結果

[ディスカバリー結果] テーブルには、ディスカバリージョブから検出されたすべての検出結果が一覧表示されます。ディスカバリージョブの結果を確認および分類する 検出結果のレビュー を参照してください。

ディスカバリー結果テーブル

ラベル	説明
辞書エントリー	データがあったターゲットテーブルの列
テーブル	データが見つかったターゲットテーブル
データパターン (Data Pattern)	データの検索に使用されるデータパターン

ディスカバリー結果テーブル (続く)

ラベル	説明
データパターン一致数 (Data Pattern Match Count)	データパターンに一致するデータエントリの数
合計行スキャン数 (Total Row Scan Count)	ジョブ中にスキャンされた行数
一致する行の割合 (Percentage of Matching Rows)	データパターンに一致するターゲットテーブルの行の割合
データディスカバリージョブ	ターゲットテーブルで使用されるジョブ
ステータス	エントリのステータス

粒度の構成

[粒度の構成] テーブルには、すべての粒度ディスカバリージョブがリストされます。詳細なディスカバリージョブの作成方法については、「 詳細なジョブを作成 」を参照してください。

粒度の構成テーブル

ラベル	説明
テーブル	粒度ディスカバリージョブのテーブル
列ラベル	粒度の検出ジョブの列ラベル
スキャン開始点	機密データは、スキャン開始点の当日以降についてのみ検出されます。 注: スキャン開始点を空のままにすると、列のすべてのエントリーがスキャンされます。
アクティブ	詳細なディスカバリージョブのステータス。

粒度の検出結果

[粒度の検出結果 **(Granular Findings)**] テーブルには、粒度の検出ジョブから粒度の検出結果がすべてリストされます。詳細なディスカバリージョブの結果を確認するには、「 粒度の検出結果のレビュー 」を参照してください。

[粒度の検出結果] テーブル

ラベル	説明
レコード	検出されたレコード
テーブル	レコードの親テーブル
データパターン	レコードの検出に使用するパターン
アクション	レコードに対して実行するアクション レビュー レコードはレビュー待ちです。これは、新しい粒度の検出にアサインされます

[粒度の検出結果] テーブル (続く)

ラベル	説明
	無視 レコードに対してアクションは実行されません。 匿名化 レコードは匿名化されます
ステータス	レコードのステータス 新規 検出結果が最初に報告されたときにアサインされるステータス 処理済み ユーザーが選択したアクションが検出結果に正常に適用されたとき 注: 処理済みの検出結果は、[粒度の検出結果 (Granular Findings)] テーブルに 3 日間保存されてから削除されます。 手動レビュー ユーザーが選択したアクションの適用に失敗した場合 警告: 手動レビューの検出結果は、適切なアクションを実行した後にユーザーが削除する必要があります。

ディスカバリージョブを作成

新しいデータディスカバリーストアジョブを作成してスケジュールします。

始める前に

必要なロール:discovery.admin

手順

1. 移動先 すべて > データディスカバリー > スケジュール済みディスカバリー.
2. 右側のナビゲーションペインで [ディスカバリージョブ] を選択します。
3. フォームに入力します。

[新規ディスカバリーをスケジュール] フォーム

フィールド	説明
名前	ジョブの名前。
説明	ジョブの説明。

フィールド	説明
スキャンタイプ	スキャンするエントリの数。可能なステータスは次のとおりです。 ○ サンプル：10,000 エントリをスキャンします。 ○ 完全：すべてのエントリをスキャンします。
ポリシーを選択	スケジュール済みジョブに使用するポリシー。
開始日	ジョブの開始日を設定します。
期間開始	このジョブを実行する期間の開始時間。ジョブは、このフィールドに入力された時間の後に実行されます。[期間開始] フィールドに入力された時間は、[期間終了] フィールドに入力された時間より前である必要があります。 注: 有効な時間値は、24 時間表記に基づく世界標準時です。
期間終了	このジョブを実行する期間の終了時間。ジョブは、このフィールドに入力された時間まで実行されます。ジョブがまだ完了していない場合、ジョブは一時停止し、次の期間の開始時に再開されます。[期間終了] フィールドに入力された時間は、[期間開始] フィールドに入力された時間より後である必要があります。 注: 有効な時間値は、24 時間表記に基づく世界標準時です。

4. [スケジュール] ボタンを選択します。

検出結果のレビュー

データディスカバリーストアでのデータの分類

始める前に
必要なロール:discovery.admin

手順

1. 検索項目 すべて > データディスカバリー > スケジュール済みディスカバリー.
2. 右側のナビゲーションペインで [ディスカバリー結果 **(Discovery Findings**)] を選択します。
3. 分類するエントリをチェックします。
   [編集] ボタンを選択して、選択したエントリの分類を編集します。
4. [分類] ボタンを選択します。

   注: [利用可能な保護] ボタンを選択して、分類されたデータの保護オプションを確認します。

5. テーブルエントリに関連付けるデータ分類を選択します。

6. [分類] を選択します。

結果
これで、選択したデータが分類されました。

詳細なジョブを作成

データディスカバリーストアの特定のテーブル列をスキャンします。

始める前に
必要なロール:discovery.admin

手順

1. 移動先 すべて > データディスカバリー > スケジュール済みディスカバリー.

2. 右側のナビゲーションペインで [粒度の構成 **(Granular Configuration**)] を選択します。

3. **[Create new** (新規作成)] を選択します。

4. フォームに入力します。

[新しい粒度の構成を作成 **(Create new granular configuration)]** フォーム

フィールド	説明
テーブル	スキャンするテーブル
列ラベル	スキャンする列
スキャン開始点	機密データは、スキャン開始点の当日以降についてのみ検出されます。 **注:** スキャン開始点を空のままにすると、列のすべてのエントリがスキャンされます。スキャンの開始点が変更されると、スキャンは新しく構成されたタイムスタンプから開始されるようにリセットされます。

5. アクティブスライダーを設定します。

6. **[Save** (保存)] を選択します。

結果
粒度スキャンがターゲットテーブルと列で実行されるようにスケジュールされています。実行後にスキャン結果を確認できます。

粒度の検出結果のレビュー

データディスカバリーストアで粒度の検出結果を確認する

始める前に
必要なロール:discovery.admin

手順

1. 移動先 すべて > データディスカバリー > スケジュール済みディスカバリー.
2. 右側のナビゲーションペインで [粒度の検出結果 **(Granular Findings**)] を選択します。
3. テーブルのエントリを確認します。

列ラベル	説明
レコード	検出されたレコード。
テーブル	レコードの親テーブル。
データパターン	レコードの検出に使用されるパターン。
アクション	検出結果に対して実行するアクション。選択して変更できます。 **重要:** レコードに対して匿名化アクションを実行するには、data_privacy_admin ロールが必要です。 レビュー レコードはレビュー待ちです。これは、新しい粒度の検出にアサインされます。 無視 レコードに対してアクションは実行されません。 匿名化 レコードは匿名化されます。
ステータス	レコードのステータス。 新規 検出結果が最初に報告されたときにアサインされるステータス 処理済み ユーザーが選択したアクションが検出結果に正常に適用されたとき **注:** 処理済みの検出結果は、[粒度の検出結果 (Granular Findings)] テーブルに 3 日間保存されてから削除されます。 手動レビュー ユーザーが選択したアクションの適用に失敗したとき

列ラベル	説明
	⚠ 警告: 手動レビューの検出結果は、適切なアクションを実行した後にユーザーが削除する必要があります。

⚠ 警告:

粒度の検出結果からトリガーされた匿名化では、ロールバックはサポートされていません。

データ分類

事前定義またはユーザー定義のデータ分類を使用して、データをタイプ別にグループ化します。ユーザーがデータ分類アドミニストレーターまたは監査人ロールを持っている場合は、さまざまなデータクラスを管理したり、インスタンス内のさまざまなタイプのデータの現在のステータスを視覚的に分析したりすることができます。

データ分類の詳細

データ分類について学習します。

データ分類の構成

独自のデータクラスを作成して構成します。

自動翻訳

データ分類の参照

デモデータを使用してデータ分類がどのように機能するかを説明します。

データ分類の分析

データ分類の分析方法について説明します。

データ分類の概要

データ分類について説明します。

データ分類を使用して次のサポートを有効にできます。

- ServiceNow AI Platform インスタンスでホストされているデータのタイプの可視化。
- プライバシー法を遵守し、金融サービスや医療機器の製造など、業界の規制要件を満たします。

データ分類

データ分類は、任意のテーブルの既存のディクショナリーエントリにデータ分類を手動で適用するスタンドアロンプロセスです。詳細については、「データディクショナリテーブル」を参照してください。

- ビジネスに適したデータ分類を行い、必要に応じて利用可能なデータクラスを変更できます。
- データを分類するときは、事前定義されたデータ分類を使用することも、独自に作成することもできます。事前定義されたデータ分類の使用はオプションですが、開始点として使用することをお勧めします。これらの事前定義されたデータ分類は、インスタンスにインストールできるデモデータに含まれています。詳細については、「データ分類 プラグインのデモデータをインストールする」と「データ分類デモデータと一緒にインストールされるコンポーネント」を参照してください。
- 独自のデータ分類を作成する場合は、親と子のデータ分類を使用して階層型の階層システムを設計することもできます。

概要ダッシュボード

概要ダッシュボードを使用して、現在のデータテーブルがさまざまなデータ分類にどのようにマッピングされるかを把握することができます。また、グローバル、地域、国際的なユーザーが、データの使用またはアクセスに関して、データ分類にどのように異なるアプローチを必要とするかを分析することもできます。概要ダッシュボードのコンテンツとレイアウトをニーズに合わせてカスタマイズすることもできます。

利用可能なスクリプト化された REST API を使用して、既存のプロセス、ワークフロー、およびアプリケーション内に分類メタデータを適用する方法については、以下を参照してください。

- データ分類 - REST API
- DCManager - グローバル
- ScopedDCManager - スコープ指定

注: データ分類 はドメインセパレーションをサポートしており、data_classification テーブル自体がプロセス分離されています。

ユースケース

一般データ保護規則 (GDPR) は、個人が自分の個人データを制御できるようにすることを目的とした欧州連合の規制です。個人を特定できる情報などのデータ分類を使用して、個人データがインスタンス内のどこに保存されているかを特定できます。適切なセキュリティメカニズムを適用して個人データの漏洩を防ぐことで、組織は GDPR 要件を満たすことができます。

顧客情報を ServiceNow AI Platform に保存する場合は、現地のプライバシー法の規制の対象となるデータを追跡するために必要な個人情報 (PII) 分類コードを使用します。デモデータをインストールすると、この分類コードがユーザー [sys_user] テーブルの特定のセキュリティ機密フィールドに自動的に適用されます。詳細については、以下を参照してください。

- データ分類デモデータと一緒にインストールされるコンポーネント
- 辞書エントリへのデータ分類の割り当て

制限付きデータ分類は、社会保障番号 (SSN) などの従業員の機密情報を格納する従業員テーブル列に適用できます。アドミニストレーターと監査人は、概要ダッシュボードを使用して、データ分類が正しい列に割り当てられていることを確認できます。また、制限されたタイプの情報の分類の詳細を表示することもできます。

データ分類 プラグインのデモデータをインストールする

Zurich (以降) にアップグレードまたはインストールすると、データ分類(com.glide.data_classification) プラグインが自動的に有効になります。ただし、プラグインに付属するデモデータは手動でインストールする必要があります。事前定義された重要なデータ分類がいくつか含まれており、そのうちの 1 つをインスタンスの特定のユーザー [sys_user] テーブル列に割り当てます。

始める前に
必要なロール：admin

このタスクについて
デモデータをインストールするかどうかに関係なく、アクティブ化された データ分類 プラグインは次のユーザーロールをインスタンスに追加します。

- data_classification_admin：データ分類のセットアップとアサインを含むデータ分類アプリケーションのすべての側面を管理できます。
- data_classification_auditor：ユーザーのテーブルと列に対するデータ分類コード割り当てを監査できます。

重要: お客様による使用に関する通知：

このアプリケーションの実装に関連するすべての意思決定は、お客様独自の判断によるものとします。お客様は、アプリケーションの使用は法律または規制の遵守に関する ServiceNow による表明ではなく、アプリケーションですぐに利用できる推奨文言、分野または分類が ServiceNow による法的な助言によるものでないことを承認および同意するものとします。

お客様は、データの保護、セキュリティ要件および個人データの収集、使用、開示、およびプライバシーに関する法律を含む (がそれに限定されない) 該当の法律に基づく法的義務を遵守する責任を単独で負い、お客様の要件を満たすために、テンプレートを含む (がそれに限定されない) このアプリケーションに対する必要な変更を構成および実施する責任を負うものとします。

手順

1. 移動先 システムアプリケーション > 利用可能なすべてのアプリケーション > すべて.
2. フィルター条件と検索バーを使用して、データ分類プラグインを検索します。
   プラグインを見つけると、「インストール済み」というメッセージが表示されます。
3. 縦に並んだ 3 つのドットアイコンをクリックし、[修理] を選択して [プラグインをアクティブ化] ダイアログを選択します。
4. [デモデータのロード] を選択し、[修復] をクリックします。

データ分類デモデータと一緒にインストールされるコンポーネント

データ分類 (com.glide.data_classification) プラグインに含まれるデモデータをインストールすると、いくつかのタイプのコンポーネントがインスタンスにインストールされます。これらのコンポーネントには、特定のユーザー [sys_user] 列の事前定義されたデータ分類とコード割り当てが含まれます。

インストールされるデータ分類

データ分類	説明
機密	侵害された場合に操作に悪影響を与える可能性がある機密データ。
内部	一般公開用ではない内部データ。
個人識別可能情報	PII とも呼ばれます。特定の個人を識別するために使用される可能性があるデータ。
公開	自由に公開される可能性のあるデータ。
制限付き	機密性の高い企業データで、侵害された場合に組織が財務的または法務的なリスクに晒される可能性があります。

データ分類 割り当て

テーブル	割り当てられた列	割り当てられたデータ分類
sys_user	zip	個人識別可能情報
sys_user	first_name	個人識別可能情報
sys_user	メール	個人識別可能情報
sys_user	city	個人識別可能情報
sys_user	middle_name	個人識別可能情報
sys_user	street	個人識別可能情報
sys_user	mobile_phone	個人識別可能情報
sys_user	last_name	個人識別可能情報
sys_user	country	個人識別可能情報
sys_user	gender	個人識別可能情報
sys_user	name	個人識別可能情報
sys_user	photo	個人識別可能情報
sys_user	state	個人識別可能情報
sys_user	home_phone	個人識別可能情報

データ分類の作成

データ分類 [data_classification] テーブルに独自のユーザー定義のデータ分類を作成し、特定のテーブルの特定の列に割り当てることができます。

始める前に
必要なロール：data_classification_admin、admin

手順

1. 移動先 すべて > システムセキュリティ > データ分類 > データクラス.
2. **[New** (新規)] を選択します。
3. フォームのフィールドに入力します。

フィールド	説明
分類名	データ分類名。
説明	データ分類の説明。
親	このデータ分類が従属している親データ分類の名前。このデータ分類が子データ分類の親ではない場合はフィールドを空のままにします。
アプリケーション	このデータ分類のアプリケーションスコープ。

4. このデータ分類を子データ分類の親にする必要がある場合は、[新規] をクリックします。
子データ分類を作成しない場合は、この手順をスキップします。

5. フォームのフィールドに入力します。

タイトル

フィールド	説明
分類名	子データ分類名。
説明	子データ分類の説明。
親	このデータ分類が従属している親データ分類の名前。このデータ分類が子データ分類の親ではない場合はフィールドを空のままにします。
アプリケーション	この子データ分類のアプリケーションスコープ。

6. [送信] をクリックします。

辞書エントリへのデータ分類の割り当て

辞書 [sys_dictionary] テーブルの特定のテーブル列にデータ分類を割り当てます。データ分類を割り当てると、辞書データクラス [m2m_dictionary_dataclass] テーブルにエントリが作成されます。作成されたエントリは概要ダッシュボードで確認できます。

始める前に
必要なロール：data_classification_admin、admin

手順

1. [ナビゲーション] ペインで、「sys_dictionary.list」と入力します。
2. 辞書エントリで、特定のデータ分類を割り当てる各要素を選択します。
3. 要素を選択したら、[選択した行のアクション] をクリックし、[分類] を選択します。

   注: 選択した辞書要素に以前に割り当てたデータ分類をクリアするには、[分類をクリア] を選択します。

4. [データクラスにアサイン] ダイアログが表示されたら、選択した辞書要素に割り当てるデータ分類を選択し、[分類] をクリックします。

   警告: これにより、選択した辞書アイテムの既存の分類が上書きされます。

   必要に応じて、複数のデータ分類を選択できます。

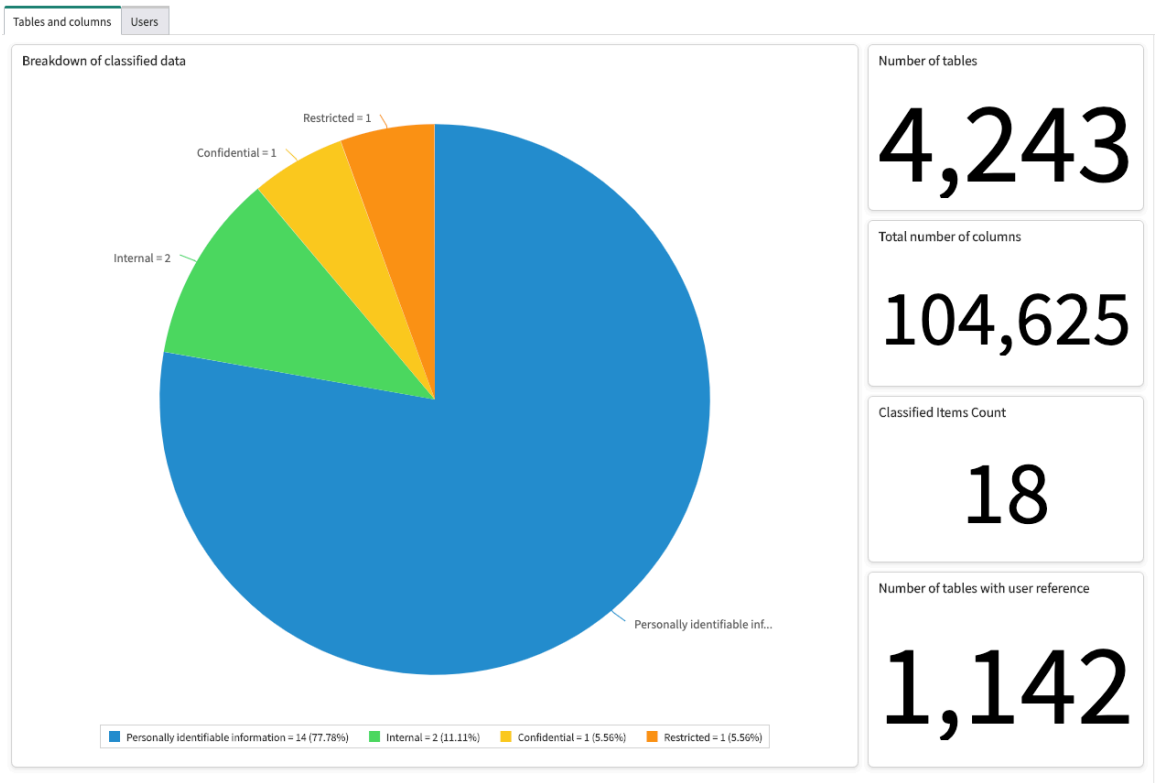
   詳細については、「データディクショナリテーブル」を参照してください。

概要ダッシュボードを使用したデータ分類の分析

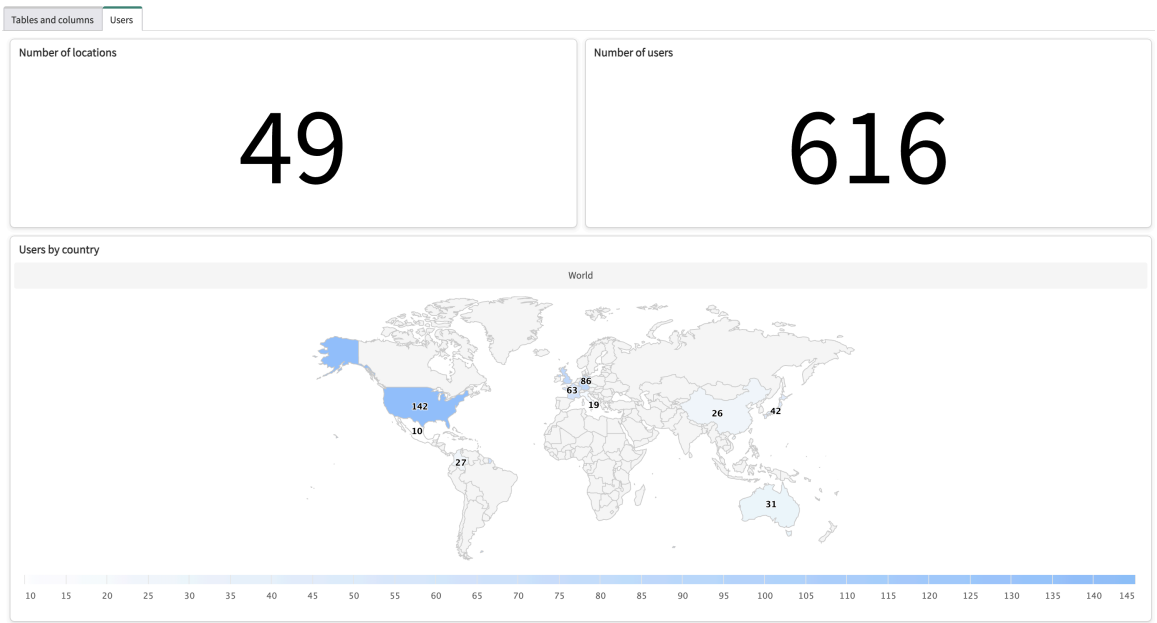
概要ダッシュボードは、インスタンス内のデータ分類の現在のステータスと、ユーザーが場所ごとにどのように分布されているかをレポートします。

ユーザーがデータ分類管アドミニストレーターであるか監査人ロールを持っている場合、インスタンスデータの現在の機密性を視覚化できるため、セキュリティが高まり、プライバシー法への遵守が向上します。アドミニストレーターは、ユーザーレコードの [場所] フィールドを使用して、プライバシー規制が異なるさまざまな地域にユーザーをマッピングできます。

[テーブルと列] タブ



[ユーザー] タブ



必要な **ServiceNow AI Platform** ロール

- data_classification_admin：データ分類のセットアップとアサインを含む データ分類 アプリケーションのすべての側面を管理できます。
- data_classification_auditor：データ分類 コードのアサインを監査します。

ユースケース

組織内のさまざまなユーザーがこのダッシュボードをどのように使用するかのその他の例については、データ分類のユースケースを参照してください。

ユーザー	ダッシュボードの使用法
data_classification_admin	インスタンスの正しいフィールドにデータ分類が割り当てられていることを確認します。
data_classification_auditor	インスタンス内のデータのシニリティとセキュリティを監査します。

レポート

役職	タイプ	ソーステーブル	説明
分類済みデータのブレークダウン	ドーナツグラフ	m2m_dictionary_dataclass	データクラス別に、インスタンスデータ分類のブレークダウンを提供します。中心に合計が表示されます。
合計テーブル数	単一スコア	sys_dictionary	ディクショナリ全体のデータテーブルの合計数。
合計列	単一スコア	sys_dictionary	ディクショナリ全体の分類済みデータ列の合計数。
ユーザー参照列 (User reference columns)	単一スコア	sys_dictionary	ディクショナリ内のユーザー参照列の合計数。
ユーザーの場所の数 (User location count)	単一スコア	sys_user	ユーザーに対して見つかった個別の場所の合計数
ユーザー数	単一スコア	sys_user	インスタンス内のユーザーレコードの合計数。
国別のユーザー	マップ	sys_user	国別のユーザーレコードのブレークダウン。

ドメインセパレーションとデータ分類

データ分類 ではドメインセパレーションがサポートされています。 ドメインセパレーションでは、データ、プロセス、および管理タスクをドメインと呼ばれる論理的なグループに分けることができます。どのユーザーがデータを表示できるか、データにアクセスできるかなど、このアプリケーションのいくつかの側面を制御できます。

サポートレベル：拡張

- ベーシックおよびスタンダードレベルのサポートのすべての側面が含まれます。
- データドリブンプロセスにより、サービスプロバイダーの顧客は定義されたユースケースに基づくビジネスロジックを変更できます。これらの構成は UI ベースでフェイルセーフであるため、1 人の顧客による構成が別のユーザーに影響を与えることはありません。
- インスタンスのテナントは、それ自体、MVP (minimum viable product) ビジネスロジックとデータパラメーターを設定できる必要があります。アプリケーションの通常の関数では、このロジックとパラメーターが想定されます。

サンプルユースケース：共有環境のテナント顧客は、影響度、緊急性、または優先度のマトリクスに変更を加えて、ドメイン内で優先順位を設定できる必要があります。

サポートレベルの詳細については、「アプリケーションでのドメインセパレーションのサポート)」を参照してください。

データ分類 におけるドメインセパレーションの仕組み

ドメインセパレーションの場合、アプリケーションは データ分類 [sys_data_classification] テーブルにプロセス分離を使用します。辞書データクラス [m2m_dictionary_dataclass] テーブルには、データ分離を使用します。データ分離とプロセス分離の詳細については、「ドメインセパレーションの説明」を参照してください。

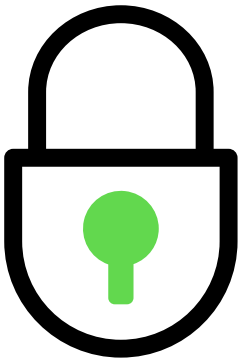
関連トピック

サービスプロバイダーのドメインセパレーション

暗号化

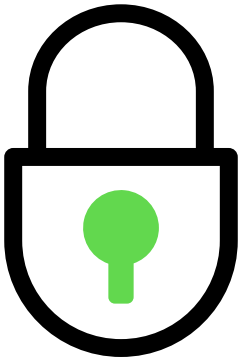
機密データを保護し、規制要件や標準への準拠を維持します。

キー管理フレームワーク



キー管理フレームワーク (KMF) を使用して、ServiceNow インスタンスでの暗号化操作の実行方法を完全にカスタマイズして管理します。

フィールド暗号化



使用方法

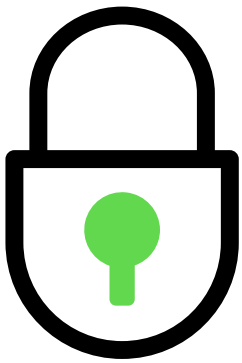
フィールド暗号化

で、ユーザーロールに基づいて暗号化データへのアクセスを許可または拒否します。

フィールド暗号化

には、暗号化モジュールを使用した基本的なキー管理が含まれています。

フィールド暗号化 エンタープライズ



使用方法

フィールド暗号化

で、ユーザーロールに基づいて暗号化データへのアクセスを許可または拒否します。

フィールド暗号化

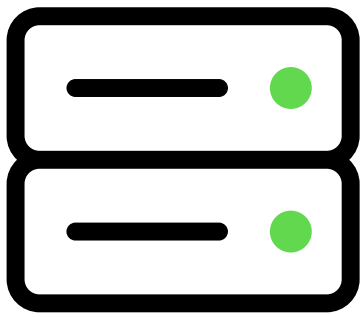
には、暗号化モジュールを使用した基本的なキー管理が含まれています。

クラウド暗号化

ブロック暗号化と拡張キー管理を使用してインスタンスデータベースを暗号化します。

Platform Encryption エンタイトルメントバンドル

Full Disk Encryption (フルディスク暗号化) **(FDE)**



自動翻訳

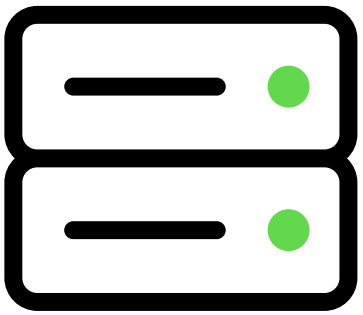
無制限利用へのアップグレード

フィールド暗号化
エンタープライズ

、クラウド暗号化、およびデータベース暗号化。

フルディスク暗号化はデータベースサーバー内のストレージシステム全体にのみ暗号化を適用します。これは、顧客データを格納する唯一のコンポーネントであるためです。

エッジ暗号化

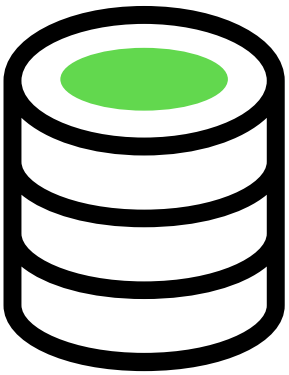


社内の機密データを暗号化してから、データをインターネット経由で ServiceNow インスタンスに送信します。データはインスタンス上で暗号化されたままになります。

証明書

証明書を使用してセキュア接続を確立し、署名を検証します。

データベース暗号化



すべての保存データをリアルタイムで暗号化し、機能を損なうことなくオンラインとオフラインのデータを保護します。

自動翻訳

キー管理フレームワーク

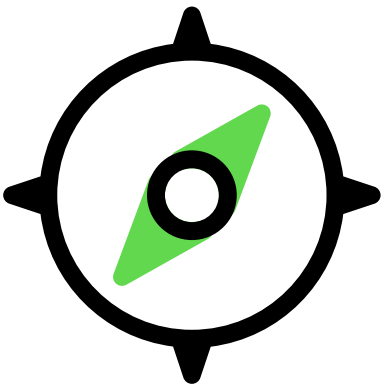
キー管理フレームワーク (KMF) を使用して、ServiceNowインスタンスの機密データの暗号化と復号化に使用される暗号化キーを生成、交換、保存、使用、および置換します。

キー管理とは、キーのライフサイクル中に暗号化キーと関連するセキュリティパラメーターの処理に関連するアクティビティを指します。 キー管理フレームワーク は、 米国国立標準技術研究所 (NIST) 800-57 ガイドラインに基づいています。これらのガイドラインに従って、 KMF を使用して次のことができます。

- 暗号化の管理と操作、監査、および統合に専用のロールを割り当てます。
- 一意の暗号化目的とキータイプに合わせて暗号化仕様を構成する暗号化モジュールを作成します。
  - 対称キー：暗号化と復号化、キーのラッピングとラップ解除、および認証
  - 非対称キー：デジタル署名の生成と検証、暗号化と復号化、キーのラッピングとラップ解除
- キーのライフサイクルを管理して、キーを生成、ローテーション、取り消し、一時停止します。これには、いくつかのキーライフサイクル状況のサポートも含まれます
- モジュールアクセスポリシー (MAP) を作成してアクセス制御を適用し、選択したユーザーとスクリプトにのみアクセス権を付与します。
- 連邦情報処理標準 (FIPS) 140-2-L3 ハードウェアの Root of Trust (RoT)、公開鍵インフラストラクチャ (PKI)、キー階層、およびエンベロープ暗号化を使用して暗号化キーを保護します。
- 監査ロールをユーザーにアサインすると、主要な使用状況統計情報などの監査情報を表示できます。

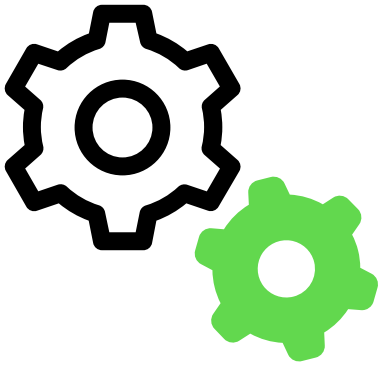
開始

キー管理フレームワークの詳細



キー管理フレームワークのコンポーネントと、それらを使用してインスタンスでの暗号化操作の実行方法を管理する方法について説明します。

キー管理フレームワークの構成



キー管理コンポーネントを作成および管理して、暗号化操作の実行方法をカスタマイズおよび管理します。

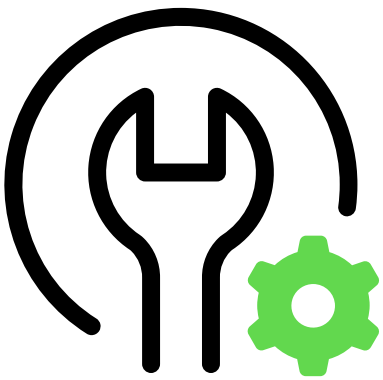
ServiceNow インスタンス

キー管理フレームワークリファレンス

キー管理コンポーネントを作成および管理して、暗号化操作の実行方法をカスタマイズおよび管理します。

ServiceNow インスタンス

キー管理フレームワークのアクション



KMF 機能を使用してキーを管理、取り消し、またはローテーションし、最新の暗号化素材とライフサイクル操作で機密データを保護します。

アクティベーション情報

ServiceNow Platform Encryption サブスクリプションバンドルは、キー管理フレームワーク、フィールド暗号化エンタープライズ、クラウド暗号化、およびデータベース暗号化を含む商用グループエンタイトルメントです。

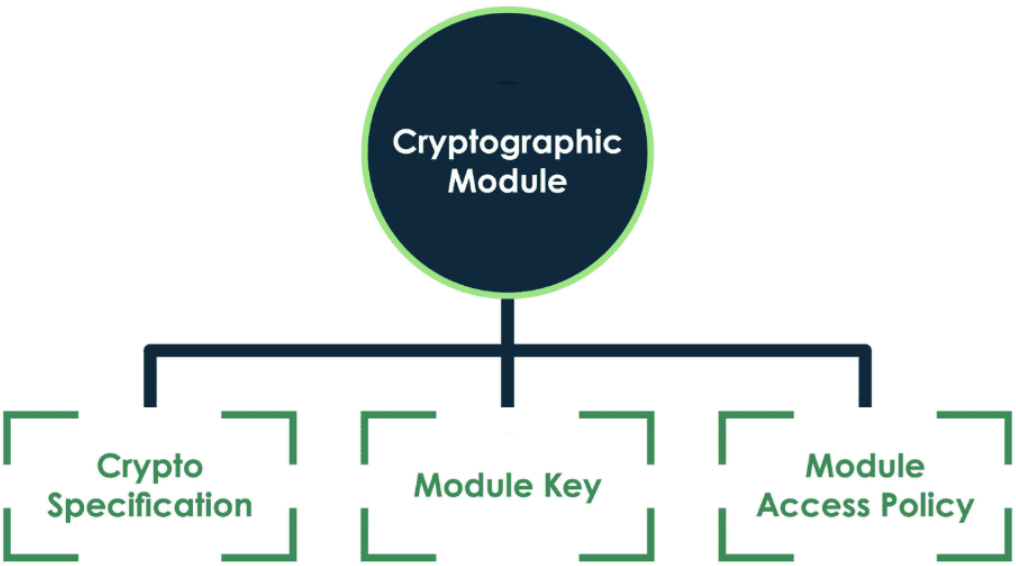
フィールド暗号化エンタープライズ は フィールド暗号化 の無制限ライセンスです。フィールド暗号化 Enterprise プラグインは、 com.glide.now.platform.encryption プラグインをアクティブ化すると利用できます。詳細については、「暗号化とキー管理のサブスクリプションバンドル」を参照してください。

注: KMF はドメインセパレーションをサポートしていませんが、オンプレミスのインスタンスで使用できます。

キー管理フレームワークの詳細

キー管理フレームワーク (KMF) のコンポーネントと、それらを使用してインスタンスでの暗号化操作の実行方法を管理する方法について説明します。

キー管理フレームワークのコンポーネント



キー管理フレームワーク は次のコンポーネントで構成されています。

暗号化モジュール

KMF は暗号化モジュールの管理を中心としています。これらのモジュールは、他のコンポーネントの親レコードとして機能します。これらは、インスタンス上のどのデータを暗号化するか、および使用する暗号化方法を定義します。複数のモジュールを使用して、インスタンスのさまざまな領域を異なる仕様で暗号化できます。

たとえば、特定のロールを持つユーザーに対してヒューマンリソース (HR) アプリケーションのデータを保護するモジュールを作成できます。その後、作成したスクリプトに基づいて特定のユーザーに表示されるインシデントの説明を暗号化する別のモジュールを作成できます。

モジュールアクセスポリシーを見つけるには、次に移動します: すべて > キー管理 > 暗号化モジュール > すべて. これらのモジュールの詳細については、「 暗号化モジュールの概要」を参照してください。

モジュールキー

暗号化キーは、暗号化で使用される文字列です。暗号化アルゴリズムと一緒に使用すると、データをエンコードまたはデコードできます。これらのキーは、モジュールに割り当てられた暗号化仕様で使用されます。ServiceNow で生成されたキーを使用するか、独自のキーをアップロードするかを選択できます。

暗号化モジュールレコードの モジュールキー 関連リストから、暗号化モジュールのモジュールキーにアクセスできます。モジュールキーの詳細については、「 キー管理フレームワークのインスタンスレベルのキー」を参照してください。

暗号化仕様

暗号化仕様では、データの暗号化に使用されるアルゴリズムを定義します。これらのアルゴリズムでは、暗号化キーを使用してデータをエンコードまたはデコードします。モジュールに暗号化仕様を割り当てると、そのモジュールに割り当てられたデータの暗号化方法が決まります。

暗号化モジュールレコードの [暗号化仕様] 関連リストから、暗号化モジュールのモジュールキーにアクセスできます。モジュールキーの詳細については、「 暗号化仕様の概要」を参照してください。

モジュールアクセスポリシー

モジュールアクセスポリシー (MAP) は、暗号化モジュールに適用するアクセス制御です。これらのポリシーを使用して、暗号化モジュールで暗号化されたデータにアクセスできるユーザーとスクリプトを決定します。

モジュールアクセスポリシーを検索するには、暗号化モジュールレコードの [アクセスポリシーを表示] リンクを選択します。詳細については、「モジュールアクセスポリシーの概要」を参照してください。

キー管理フレームワークワークフロー

1. KMF ロールのアサイン

   アドミニストレーターは、まず自分自身に sn_kmf.admin ロールを割り当てることから始める必要があります。このロールでは、KMF 機能を使用し、KMF ロールを他のユーザーに割り当てることができます。

2. KMF 設定を構成する

   フィールド暗号化の設定を構成して、指定したキーまたは独自の顧客指定のキー (CSK) のいずれかを選択して暗号化します。

3.暗号化モジュールを作成する

   暗号化モジュールを使用して、暗号化するインスタンス上のデータのセットを選択します。後のステップで、暗号化仕様を割り当ててこのデータの暗号化方法を決定し、モジュールアクセスポリシーを割り当ててデータを復号化できるユーザーを決定します。

4. 暗号化仕様を作成する

   暗号化仕様は、暗号化の方法を定義します。モジュールに割り当てられると、そのモジュールに割り当てられたデータの暗号化方法が定義されます。

5. モジュールアクセスポリシーを作成する

   データを保護するモジュールを作成したら、モジュールアクセスポリシーを作成して、暗号化されたデータにアクセスできるユーザーとスクリプトを制御します。

6. 暗号化モジュールライフサイクルポリシーを作成する

   これらのポリシーは、暗号化キーの有効期間など、暗号化モジュールに制限を課します。これらのポリシーは、公開を制限することで暗号化モジュールを保護できます。

キー管理フレームワークのメリット

メリット	機能	ユーザー
機密データや専有データを保護します。	暗号化とキー管理	すべて

メリット	機能	ユーザー
NIST 800-57 ガイドラインの遵守を維持します。これらのガイドラインは、ネットワークとデータに対するサイバーセキュリティリスクを軽減するために、米国国立標準技術研究所によって提供されています。	暗号化とキー管理	セキュリティアドミニストレーター
キー管理フレームワークを使用して、暗号化キーを生成、アップロード、表示、管理します。手動またはスケジュールされたキーローテーションを使用して、セキュリティを強化します。	キー管理フレームワーク	セキュリティアドミニストレーター

暗号化モジュールの概要

キー管理フレームワーク (KMF) は、暗号化モジュールの管理を中心としています。これらのモジュールを使用して暗号化メカニズムを選択し、インスタンスに適用する場所を定義します。

暗号化モジュールは KMFの中心です。特定のユースケースの暗号化操作に使用される特定の暗号化メカニズムを定義します。

たとえば、256 ビットの対称キーを持つ AES-CBC を使用して、ヒューマンリソース (HR) アプリケーションのデータを保護します。そのためのモジュールを作成できます。

暗号化モジュールは、キーのライフサイクル管理もサポートしています。暗号化キーを作成してローテーションし、暗号化方法を定義できます。暗号化モジュールは次のコンポーネントで構成されています。

暗号化仕様

暗号化に使用するアルゴリズムと、キーの取得元を定義します。すべてのキーで Advanced Encryption Standard with Cipher Block Chaining (AES CBC) が使用されますが、128 ビットまたは 256 ビットを選択できます。この仕様は、非対称キーベースと対称キーベースの暗号化操作の両方を対象としています。

注: 対称暗号化では、暗号化と復号化の両方に 1 つのキーを使用します。非対称暗号化では、暗号化用の公開鍵と復号化用の秘密鍵のペアを使用します。

暗号化キー

モジュールが暗号化データのエンコードまたはデコードに使用するキー。このキーは、インスタンスで生成することも、作成してアップロードした顧客指定のキーで生成することもできます。

モジュールアクセスポリシー

モジュールアクセスポリシーは、データの暗号化または復号化を制限するアクセス制御メカニズムです。

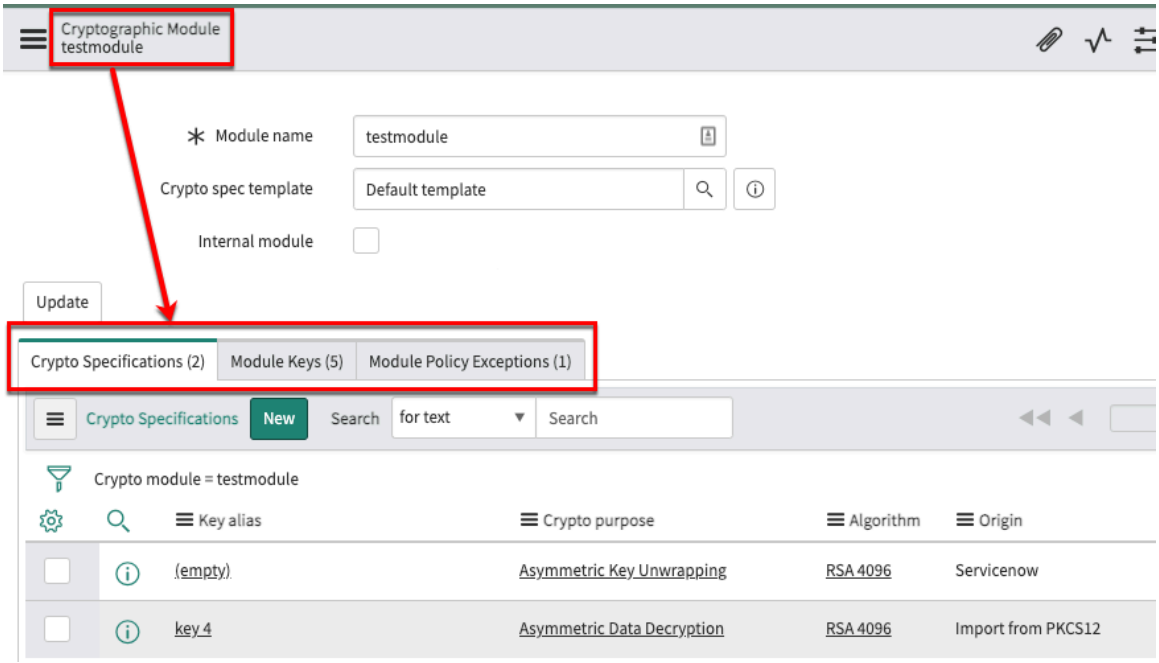
モジュールポリシーの例外

モジュールアクセスポリシーの例外を定義する制御メカニズム。

次の画面は、暗号化モジュール内のこれらのハイレベルのコンポーネントを示しています。

自動翻訳

暗号化モジュールのコンポーネント



暗号化モジュールの作成の詳細については、「暗号化モジュールを作成する」を参照してください 。

暗号化仕様の概要

暗号化仕様は、暗号化の目的や使用する暗号化アルゴリズムなど、暗号化モジュールの側面を定義するコンポーネントです。

暗号化仕様は、非対称キーと対称キーベースの暗号化の両操作を対象に、指定された暗号化の目的に合わせてカスタマイズすることができます。暗号化の目的を選択すると、サポートされているアルゴリズムとキーの長さの設定をさらに選択できます。

暗号化の目的、アルゴリズム、および重要な情報

暗号化の目的	アルゴリズム	重要な情報
非対称データ復号化	RSA	非対称 – 2048 ビット、3072 ビット、4096 ビットのキー
非対称データ暗号化	RSA	非対称 – 2048 ビット、3072 ビット、4096 ビットのキー
非対称鍵のアンラッピング	RSA	非対称 – 2048 ビット、3072 ビット、4096 ビットのキー
非対称鍵のラッピング	RSA	非対称 – 2048 ビット、3072 ビット、4096 ビットのキー
署名生成	RSA	非対称 – 2048 ビット、3072 ビット、4096 ビットのキー
署名検証	RSA	非対称 – 2048 ビット、3072 ビット、4096 ビットのキー
対称真正性	HMAC	対称 – 256 ビット、384 ビット、512 ビットのキー

暗号化の目的、アルゴリズム、および重要な情報 (続く)

暗号化の目的	アルゴリズム	重要な情報
対称データの暗号化/復号化*	AES-CBC*	対称 – 128 ビット、192 ビット、256 ビットのキー
	AES-CFB	
	AES-OFB	
	AES-CTR	
	AES-GCM**	
対称キーのラッピング/アンラッピング*	AES-CBC*	対称 – 128 ビット、192 ビット、256 ビットのキー
	AES-CFB	
	AES-OFB	
	AES-CTR	
	AES-GCM**	

* AES-CBC は等価性保存オプションをサポートしています。 フィールド暗号化エンタープライズは AES-CBC を利用します。

** AES-GCM にはデータ整合性が組み込まれています。

これらのパラメーターの設定については、「暗号化モジュールを作成する」で説明しています。

モジュールアクセスポリシーの概要

モジュールアクセスポリシー (MAP) は、暗号化モジュールに適用するアクセス制御です。これらのアクセスポリシーを使用して、暗号化モジュールで暗号化されたデータにアクセスできるユーザーとスクリプトを決定します。

モジュールアクセスポリシー

注: フィールド暗号化エンタープライズ 機能を使用するには、サブスクリプションが必要です。フィールド暗号化エンタープライズの詳細については、「フィールド暗号化 のアクティブ化」を参照してください。

モジュールアクセスポリシーは、ベースシステムの キー管理フレームワーク (KMF) で導入されます。

モジュールアクセスポリシーは、暗号化モジュールで提供されたロールベースの指定に対して展開されます。モジュールアクセスポリシーは、以下に基づきます。

- 基本 (スコープ)
- ロール
- システムユーザー
- スクリプト

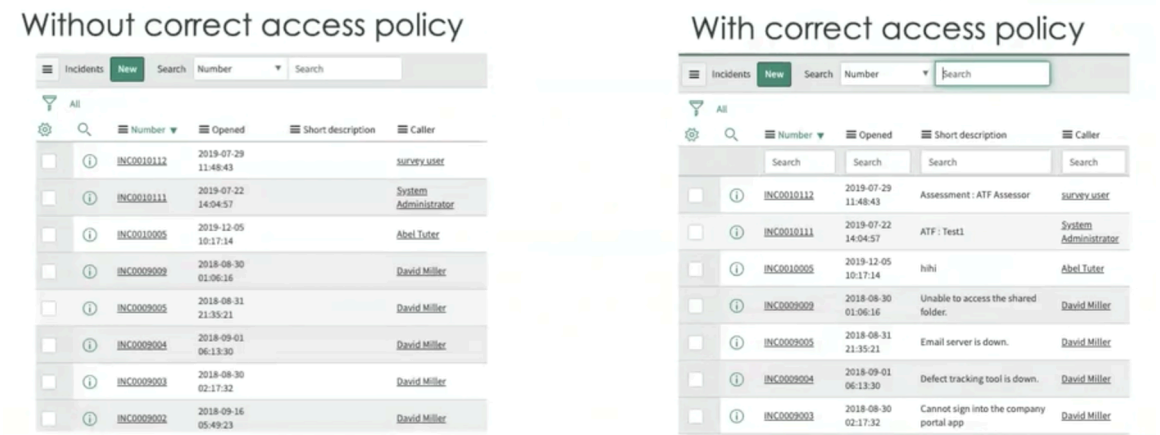
- リソース交換

  注: 詳細については、「キー管理フレームワークリソース交換」を参照してください。

暗号化モジュールでは、暗号化データへのアクセスを許可する正しいモジュールアクセスポリシーを構成する必要があります。暗号化モジュールに関連付けられたモジュールアクセスポリシーがないと、暗号化されたデータはユーザーに表示されず、リスト内の関連するフィールドと列は空として表示されます。

この例では、暗号化された [簡単な説明] フィールドにモジュールアクセスポリシーがないため、インシデントテーブルにアクセスするすべてのユーザーからコンテンツが非表示になります。モジュールアクセスポリシーを設定すると、特定のロールを持つユーザーのみが暗号化されたデータを表示できます。

暗号化された簡単な説明 (モジュールアクセスポリシーあり/なし)



注: 列のデータは、モジュールアクセスポリシーで指定された正しいロールを持たないユーザーにも空白で表示されます。

セットアップについては、「モジュールアクセスポリシーを作成する」を参照してください。

自動生成ポリシー

自動生成ポリシーは、指定された暗号化モジュールに定義されたデフォルトのモジュールアクセスポリシーに基づいて自動的にシステムで生成されます。システムまたはスクリプトが特定の暗号化モジュールにアクセスしようとしたときに詳細なレベルのポリシーが定義されていない場合は、これらのグローバルポリシーが生成されて適用されます。

重要:

自動生成ポリシールールは、スケジュール済みジョブタイプまたはフィールド暗号化モジュール (親モジュールが フィールド暗号化モジュールであるモジュール) には適用されません。

キー管理フレームワークのインスタンスレベルのキー

エンベロープ暗号化を使用して、KMF管理下にあるすべてのプラットフォームキーがキーのチェーンによって保護されるようにする、キー管理フレームワーク (KMF) キー構造について説明します。KMF によって作成された顧客データ暗号化キー (CDEK) もこの構造に含まれます

.

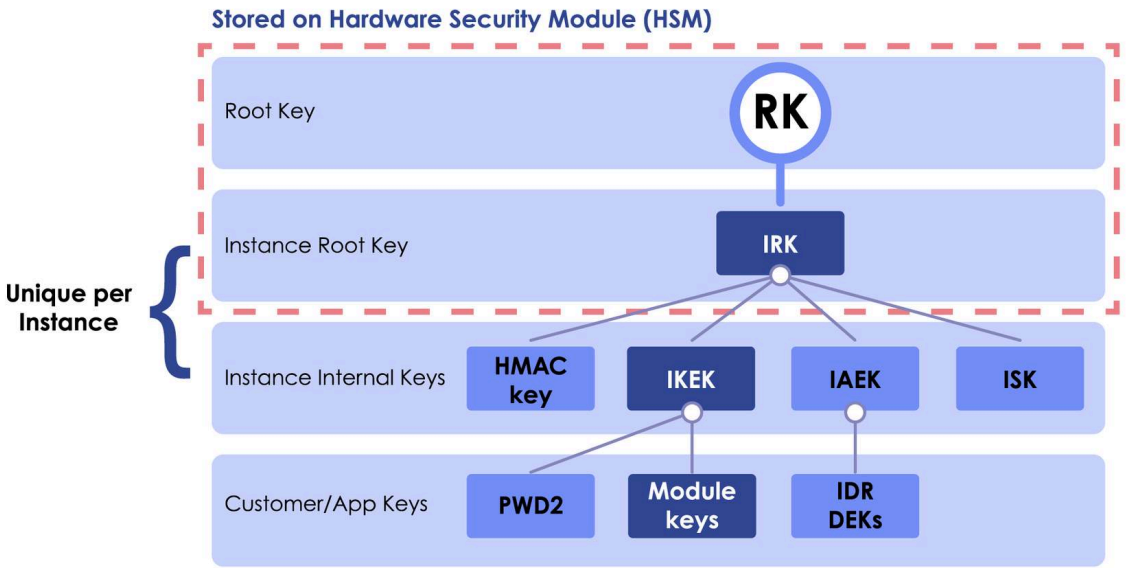
自動翻訳

KMF キーストレージアーキテクチャ

KMFキー構造では、SafeNet KeySecure ハードウェアセキュリティモジュール (HSM) を使用します。HSM は、 FIPS 140-2-L3 セキュリティ標準 を満たすために、物理的および電子的に改ざん防止されるように設計されています。 KMF はエンベロープ暗号化を使用して、 KMF 管理下にあるすべてのプラットフォームキーが、 KMFによって生成できるモジュールキーを含むキーのチェーンによって保護されるようにします。

エンベロープ暗号化

エンベロープ暗号化は、キーを別のキーで暗号化する手法です。これは別名を「ラッピング」といいます。モジュールキーはインスタンスキー暗号化キー (IKEK) によってエンベロープ暗号化され、IKEK はインスタンスルートキー (IRK) によってエンベロープ暗号化され、最終的にルートキー (RK) によってエンベロープ暗号化されます。HSM のみが IRK にアクセスできるため、復号化のために IKEK をアップロードする必要があります。



インスタンスレベルでは、KMF は ServiceNow AI Platform 全体でさまざまな暗号化の目的で内部的に使用される複数のキーを定義します。

次の表は、 KMFによって管理および保護される使用可能なキーのサブセットの例を示しています。

キー	ロケーション	説明
ルートキー (RK)	ハードウェアセキュリティモデル (HSM)	IRK の復号化に使用されるルートキー
インスタンスルートキー (IRK)	HSM	複数のインスタンス内部キーをエンベロープ暗号化するために使用される、インスタンスごとに固有のキー。
インスタンス HMAC キー (IHK)	インスタンス	インスタンスごとに固有の IHK が、ハッシュベースのメッセージ認証コード (HMAC) のために内部的に使用されます。

キー	ロケーション	説明
		IHK は、モジュールキーの信頼性と整合性を検証するのに役立ち、KeySecure またはファイルキーストアのいずれかにラップされます。
インスタンスキー暗号化キー (IKEK)	インスタンス	IKEK はモジュールキーをラップし、KeySecure またはファイルキーストアのいずれかでラップされます。
インスタンス非対称暗号化キー (IAEK)	インスタンス	非対称暗号化のために内部的に使用される、インスタンスごとに固有のキー。 IAEK は、鍵交換 中またはインスタンスデータレプリケーション (IDR)のコンシューマー承認中に、インスタンス間で機密メッセージを送信するために使用されます。
インスタンス署名キー (ISK)	インスタンス	署名目的で内部的に使用される、インスタンスごとに固有のキー。
Password2 (PW2)	インスタンス	KMF では、PW2 フィールドのキーは KMF によって完全に管理されます。
顧客データ暗号化キー (CDEK)	インスタンス	KMF を介して作成された暗号化キーは、IKEK によってエンベロープ暗号化されます。
インスタンスデータレプリケーション (IDR) データ暗号化キー (DEK)	インスタンス	IDR プロセスに使用される特定の暗号化キー

キー管理フレームワークの構成

キー管理コンポーネントを作成および管理して、 ServiceNow インスタンスでの暗号化操作の実行方法をカスタマイズおよび管理します。

キー管理フレームワーク ロールの割り当て

security_admin ロールを持つアドミニストレーターは、 キー管理フレームワーク (KMF) アドミンをアサインでき、そのアドミンは他の キー管理フレームワーク ロールを割り当てることができます。

始める前に

必要なロール：admin および security_admin

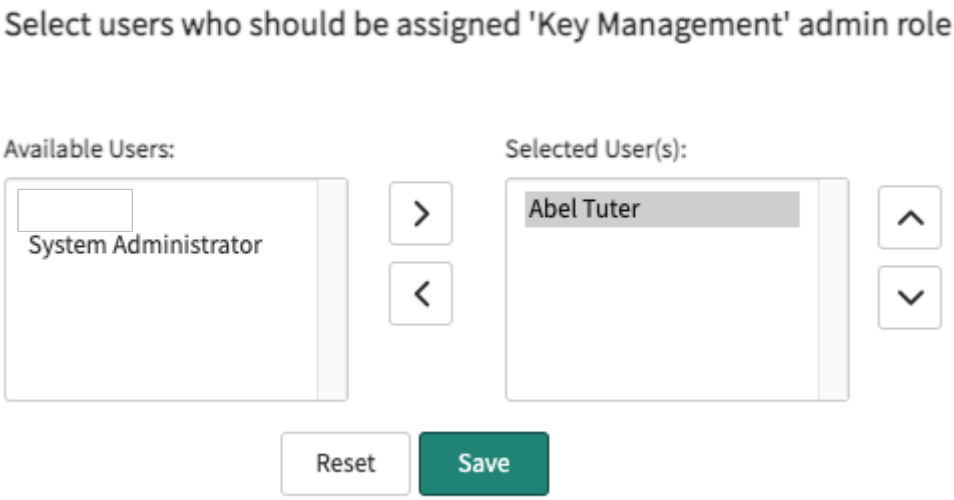
KMF admin ロールを割り当てる前に、security_admin ロールに昇格させる必要があります。手順については、「特権ロールへの昇格」を参照してください。

重要: KMFキー管理フレームワークを使用するにはロールが必要です。KMFロールを持たないユーザーは、キー管理の構成に使用されるリスト、テーブル、およびモジュールにアクセスできません。

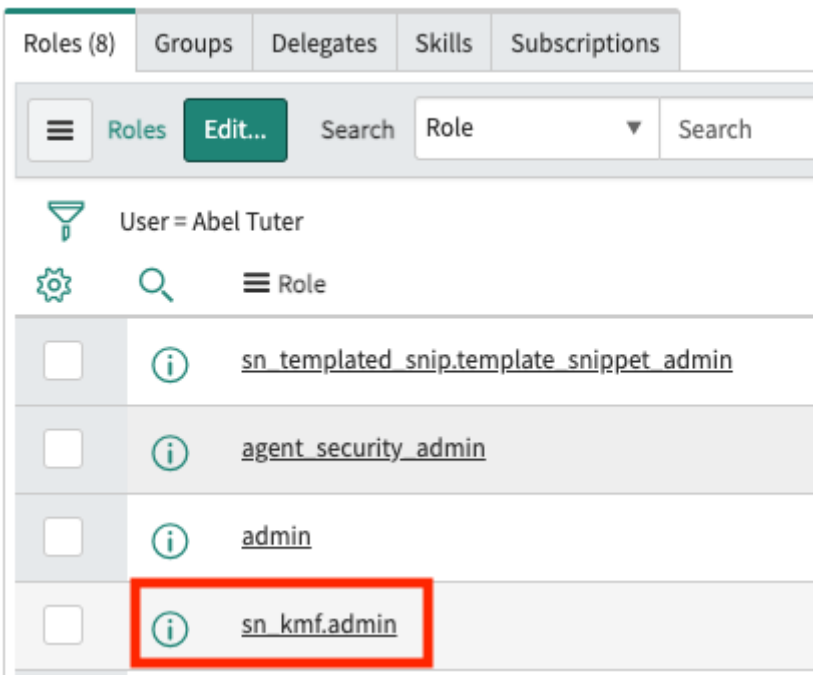
手順

1. セキュリティ管理者ロールに昇格します。
2. 移動先 ユーザー管理 > ユーザー をクリックし、 KMF アドミンにするユーザーを選択します。

3. ユーザーがまだアドミンロールとsecurity_adminロールを持っていない場合は、[ロール] 関連リストの [編集] を選択し、 アドミン とセキュリティ の**_admin**を追加します。
4. 移動先 システムセキュリティ > キー管理の管理.
5. [利用可能なユーザー] 列で KMF admin にするユーザーを選択し、[選択したユーザー] 列に移動します。

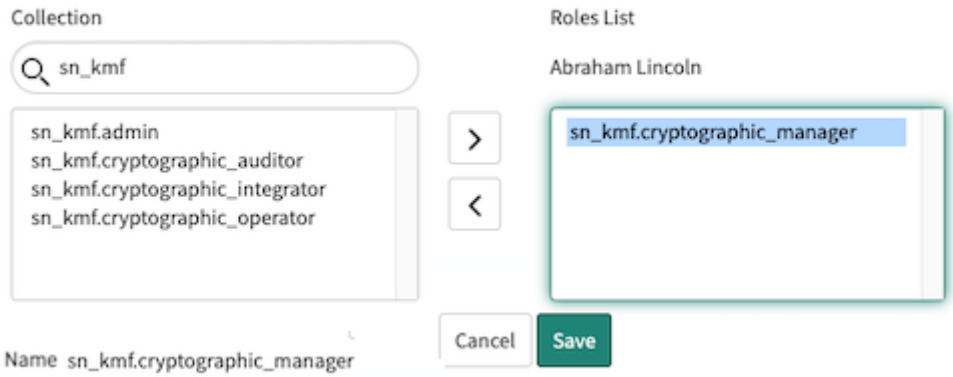


6. **[Save** (保存)] を選択します。
7. 移動先 ユーザー管理 > ユーザー をクリックし、KMF admin ロールを付与したユーザーを選択します。
これで、ロール関連リストに sn_kmf.admin ロールが付与されます。そのユーザーは他の KMF ロールを割り当てることができます。



次のタスク
KMF admin ロールを持っている場合は、次の手順に従って他の KMF ロールを割り当てます。

1. 移動先 ユーザー管理 > ユーザー をクリックし、KMF Cryptographic Manager などの別の KMF ロールを持つユーザーを選択します。
2. ロール関連リストで、[編集] をクリックし、ユーザーに割り当てる KMF ロールを選択します。すべての KMF ロールは sn_kmfで始まります。



次のタスク
利用可能な KMF ロールの詳細については、「 キー管理フレームワークとともにインストールされるロール」を参照してください。

キータイプを選択するためにフィールド暗号化を設定する

ServiceNow AI Platformの暗号化にServiceNow指定のキーまたは独自の顧客指定のキー (CSK) を使用するようにフィールド暗号化を設定します。

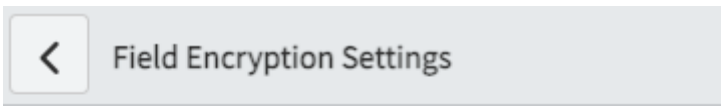
始める前に
顧客指定のキーは フィールド暗号化エンタープライズ でのみサポートされています。

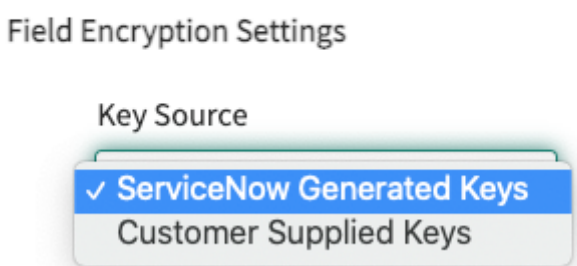
必要なロール：sn_kmf.cryptographic_manager および security_admin

手順

1. 移動先 すべて > システムセキュリティ > フィールド暗号化設定.
2. [フィールド暗号化設定] で、**[ServiceNow** によって生成されたキー] または [顧客指定のキー] をキーソースリストから選択します。

キーソースの選択





自動翻訳

このオプションで、com.glide.encryption.cle_kmf.key_source プロパティを **[ServiceNow** によって生成されたキー]または[顧客指定のキー]に変更します。

3. [保存] を選択します。

次のタスク

- 独自の顧客指定のキーを使用する場合は、「での顧客指定のキーの使用 フィールド暗号化エンタープライズ」を参照してください。
- ServiceNow 指定のキーを使用している場合は、暗号化モジュールの作成を開始します。「暗号化モジュールを作成する」を参照してください。

暗号化モジュールを作成する

暗号化操作に使用するメカニズムを定義する暗号化モジュールを作成します。モジュールを作成した後、暗号化のアルゴリズムを定義し、キーを生成する暗号化仕様を作成します。

始める前に

独自のキーを指定する場合は、顧客指定のキーの構成とアップロード に移動します。

必要なロール：sn_kmf.cryptographic_manager

このタスクについて

この手順では、ServiceNow プラットフォームベースシステムにおいて、KMF で使用できるオプションについて説明します。フィールド暗号化エンタープライズ 機能は *com.glide.now.platform.encryption* プラグインがアクティブな場合にのみ使用できます。フィールド暗号化エンタープライズ の取得についての詳細は「フィールド暗号化 のアクティブ化」を参照してください。「の暗号化モジュールを作成 フィールド暗号化」を参照してください。

注: 暗号化モジュール [sys_kmf_crypto_module] レコードは削除できません。

手順

1. 移動先 すべて > キー管理 > 暗号化モジュール > 新規作成.
2. フォームのフィールドに入力します。

暗号化モジュールのフィールド

フィールド	説明
モジュール名	スクリプトの実行時に参照される英数字の文字列。
暗号化仕様テンプレート	暗号化仕様でサポートされているアルゴリズムのマッピングが含まれているため、暗号化モジュールの作成に使用するデフォルトのテンプレートを選択します。
デフォルトのモジュールアクセスポリシー値	○ システムのデフォルトに依存： ○ 却下 ○ 追跡
実際のモジュールアクセスポリシーの結果	デフォルトのポリシー値またはモジュールアクセスポリシーの作成時に選択された値に基づいて、却下または追跡します。
名前	アプリケーションスコープ名が先頭に付加される暗号化モジュール名。

自動翻訳

フィールド	説明
暗号化モジュールライフサイクル状況	ライフサイクルとは、暗号化モジュールの作成、使用、非アクティブ化を指します。設定時に最初は [ドラフト] に設定されます。モジュールを使用する場合は、このフィールドを [公開] に設定します。 [デフォルト] テンプレートは自動的に [公開] に設定されます。

3. [送信] を選択します。

> ⚠ 警告:
>
> 従来の暗号化サポートユーザーの場合：
>
> エンタープライズ以外のバージョンの フィールド暗号化 を使用している場合は、5 つのモジュールに制限されます。この上限を超えると、次の警告が表示されます。
>
> この挿入は、サブスクリプション製品でエンタイトルメントされた フィールド暗号化 の公開モジュール数の制限を超えています。追加のモジュールには、フィールド暗号化 の Enterprise サブスクリプションが必要です。アカウントチームにお問い合わせください。

正常に送信すると、暗号化モジュールが暗号化モジュールテーブルにリストされます。他のスコープ対象アプリケーションとの競合を避けるために、名前の先頭にスコープが付加されます。たとえば、グローバルアプリケーションスコープで my_crypto_module という名前のモジュールを作成した場合、その名前は global.my_crypto_module として保存されます。

次のタスク
暗号化仕様を作成する

。

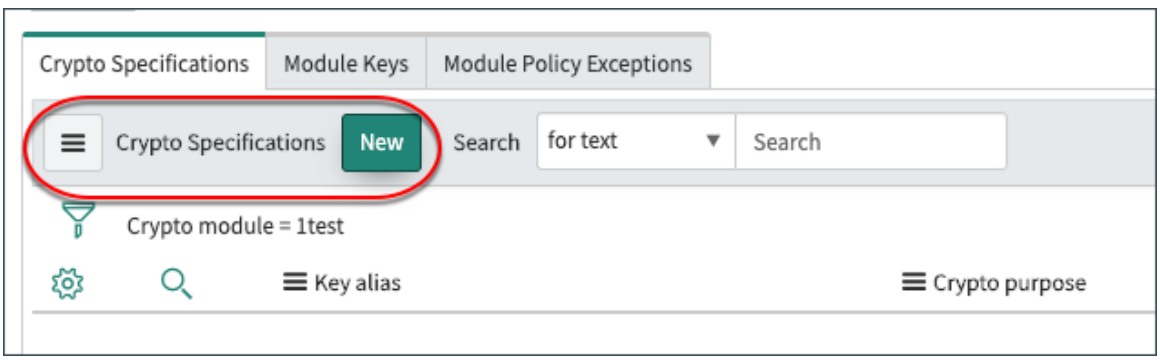
暗号化仕様を作成する

暗号化モジュールを作成してから、モジュールアルゴリズムを定義する暗号化仕様を作成します。

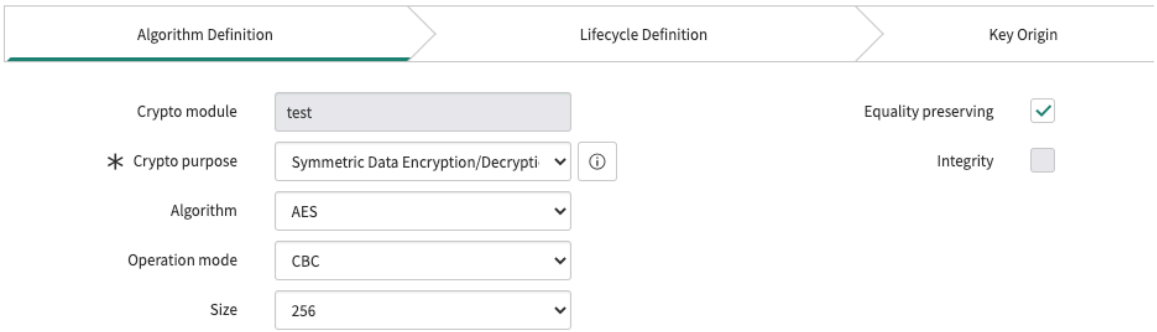
始める前に
必要なロール：sn_kmf.cryptographic_manager

手順

1. 移動先 キー管理 > 暗号化モジュール > すべて.
2. 定義用の暗号化モジュールを選択して、設定オプションを開きます。
3. [暗号化仕様] タブで、 [新規] を選択します。



4. [アルゴリズム定義] フォームに入力します。
詳細については、「暗号化仕様の概要」を参照してください。



[アルゴリズム定義] 画面が開きます。キー生成のオプションを選択します。選択した暗号化モジュールの複数のキーを生成するには、この手順を繰り返します。

アルゴリズム定義フィールド

フィールド	
暗号化モジュール	読み取り専用。選択した暗号化モジュールの名前が表示されます。
暗号化の目的	このモジュールの目的を選択します。たとえば、データの暗号化、署名の生成、またはキーのラッピングに使用できます。利用可能なアルゴリズムは、選択した暗号化の目的に基づいて調整されます。詳細については、「暗号化仕様の概要」を参照してください。
アルゴリズム	暗号化の目的を達成するために使用されるアルゴリズムのタイプ。アルゴリズムはキーの作成元も制御します。選択した暗号化の目的に基づいて自動的に調整されます。詳細については、「暗号化仕様の概要」を参照してください。
操作モード	このフィールドは、選択した暗号化の目的に基づいて表示されます。
サイズ	ビットサイズを選択します。
ハッシュ	このフィールドは、選択したアルゴリズムに基づいて使用可能になります。
等価性保存	非決定的暗号化を有効にします。 このオプションは、**AES** を使用した対称データの暗号化/復号化を暗号ブロックチェーン **(CBC)** モードで選択した場合に表示されます。 このオプションを選択すると、同じデータを再度暗号化する場合、エンコードされるデータは毎回同じになります。非決定的暗号化は、等価比較演算子を使用した暗号化データのリストを

フィールド	
	フィルタリングすることをサポートしていません。
完全性	GCM 操作モードは完全性を提供します。

5. **[Next** (次へ)] をクリックします。
暗号化仕様は、選択したアルゴリズムに基づいてキーライフサイクルテーブルにリストされます。

次のタスク
次のいずれかの操作を実行します。

- キーライフサイクルテーブルのエントリを選択して、キーライフサイクルの動作を定義します。キーライフサイクル定義の完了に関する詳細については、「キーライフサイクル状況の設定」を参照してください。
- [次へ] を選択して暗号化キーを作成します。キーの生成については、次のいずれかのタスクを参照してください。
  - ServiceNow 暗号化キーを生成する。
  - 顧客指定のキーのプロパティを設定する。
  - ラッピング/ラップ解除キーペアのインポート。

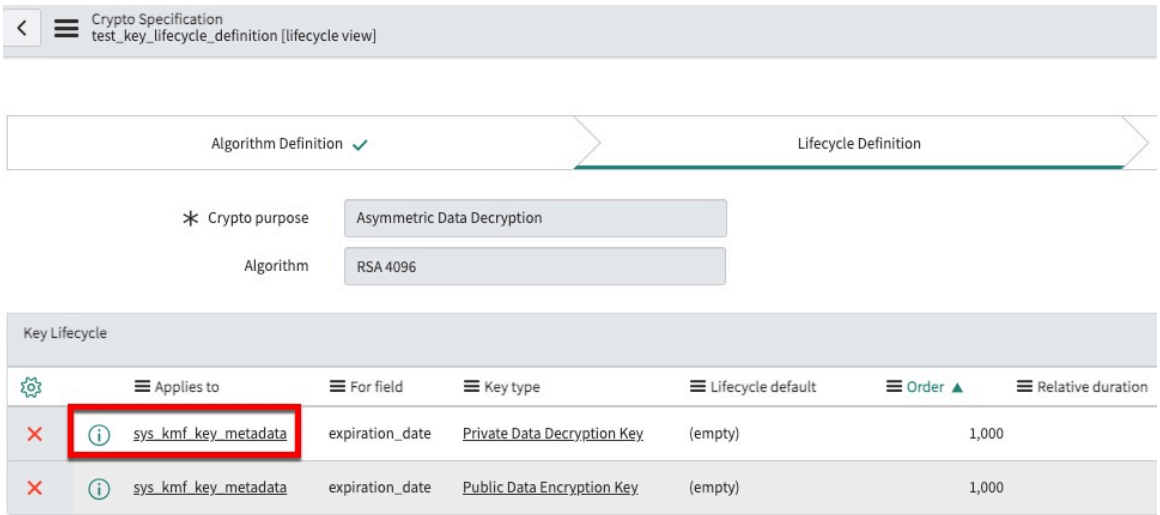
キーライフサイクル状況の設定

暗号化仕様を作成すると、インスタンス内のキーのライフサイクルアクションを設定できます。

始める前に
必要なロール：sn_kmf.admin

手順

1. 移動先 キー管理 > 暗号化モジュール > すべて.
2. キーのライフサイクルを設定する暗号化モジュールを選択します。
3. [暗号化仕様] タブでキーエイリアスを選択します。

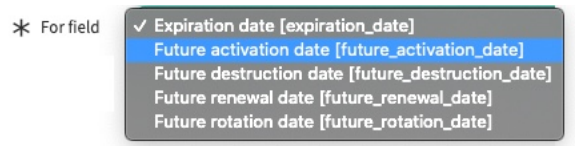
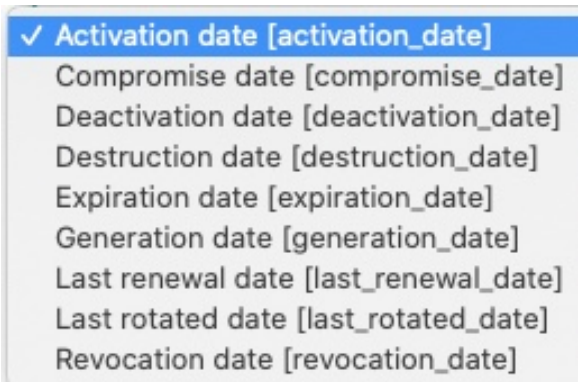


4. **[Next** (次へ)] をクリックします。

フィールドライフサイクルテンプレートがロードされます。デフォルトのキーライフサイクル値は、定義された暗号化仕様に対して選択されたアルゴリズムに基づいて作成されます。

5. 暗号化仕様のライフサイクル定義ステップの [適用先] 列から、キーライフサイクルを選択します。

キーライフサイクルフィールド

フィールド	説明
適用先	ライフサイクルが適用される選択されたキー。
フィールド用	ライフサイクルが適用されるキーの制御タイプを選択します。 キーライフサイクル管理の「フィールド用」の値 
タイプ	キーライフサイクルの評価が相対値か絶対値かを選択します。 ○ 相対：キーの生成、アクティブ化、非アクティブ化など、システム内の他のデータエントリに依存する値を入力します。 ○ 絶対：日付などの正確な値を入力します。
ライフサイクルのデフォルト	読み取り専用。設定されている場合は値を表示します。
順番	暗号化仕様のキーライフサイクル状況を処理する順序を入力します。
相対期間タイプ	ライフサイクルの期間：年、月、または日。
相対期間	キーが有効である年数、月数、または日数。
相対操作	前または 後。
相対的	期間が関連するフィールド。相対期間または操作が選択されているかどうかを表示します。 

ServiceNow 暗号化キーを生成する

機密データを暗号化するための ServiceNow 暗号化キーをアップロードして設定するには、次の手順に従います。

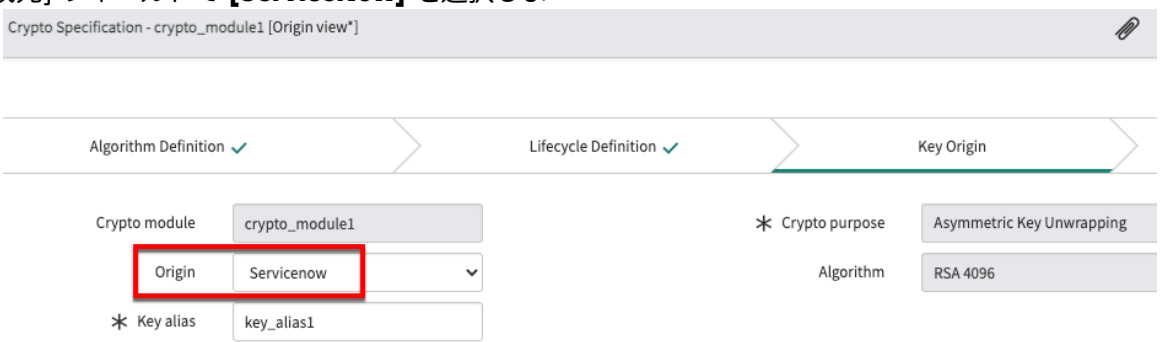
始める前に

必要なロール：sn_kmf.cryptographic_manager

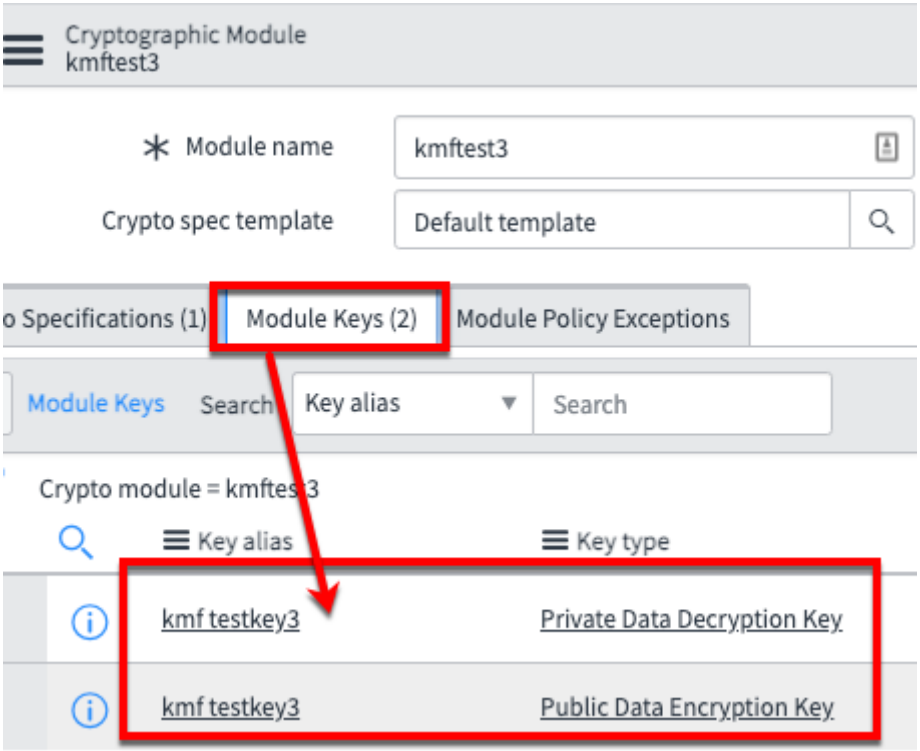
このタスクについて
暗号化マネージャーは、フィールド暗号化エンタープライズ による ServiceNow AI Platform での暗号化に ServiceNow 提供のキーを使用するか、独自の顧客指定のキー (CSK) を使用するかを選択できます。CSK の詳細については、「顧客指定のキーのプロパティを設定する」を参照してください。

手順

1. ServiceNow で生成されたキーを使用するようにフィールド暗号化を設定します。
   詳細については、「キータイプを選択するためにフィールド暗号化を設定する」を参照してください。
2. 移動先 キー管理 > 暗号化モジュール > すべて.
3. 対応する暗号化モジュールを選択して、[暗号化モジュールの詳細] ページを開きます。
4. [暗号化仕様] タブでキーエイリアスエントリの行を選択します。
   キーがまだ生成されていない場合、キーエイリアスフィールドは空です。
5. [次へ] を選択して、暗号化仕様コンポーネントの [キーの作成元] タブに移動します。
   [ライフサイクル定義] タブがキーライフサイクルテーブルとともに表示され、確認または編集できます。詳細は、「キーライフサイクル状況の設定」を参照してください。
6. [作成元] フィールドで **[ServiceNow]** を選択します。



   このフィールドは、ステップ 1 のフィールド暗号化と選択したアルゴリズムによって異なります。インポートされたキーを使用するには、「ラッピング/ラップ解除キーペアのインポート」を参照してください。独自のキーを使用する場合は、「顧客指定のキーのプロパティを設定する」を参照してください。
7. キーエイリアスのわかりやすい名前を入力します。
8. [次へ] を選択して [キーの作成] タブに移動します。
9. [キーの生成] を選択します。
   キーを生成すると、暗号化モジュールフォームが再ロードされ、暗号化仕様が表示されます。
10. [モジュールキー] タブを選択してキーを表示します。
    キーの保護情報は、暗号化仕様に存在するキーの数とともに [モジュールキー] タブに保存されます。



11. キー管理アクションを実行するキーを選択します。
詳細は、「キー管理アクション」を参照してください。

モジュールアクセスポリシーを作成する

モジュールアクセスポリシーを作成して、暗号化モジュールで暗号化されたデータにアクセスできるユーザーとスクリプトを決定します。

始める前に

必要なロール：sn_kmf.cryptographic_manager または sn_kmf.admin

このタスクについて

フィールド暗号化 はロールベースのモジュールアクセスポリシーをサポートしており、追加の構成オプションが (CLE_Ent) 機能で利用可能になります。

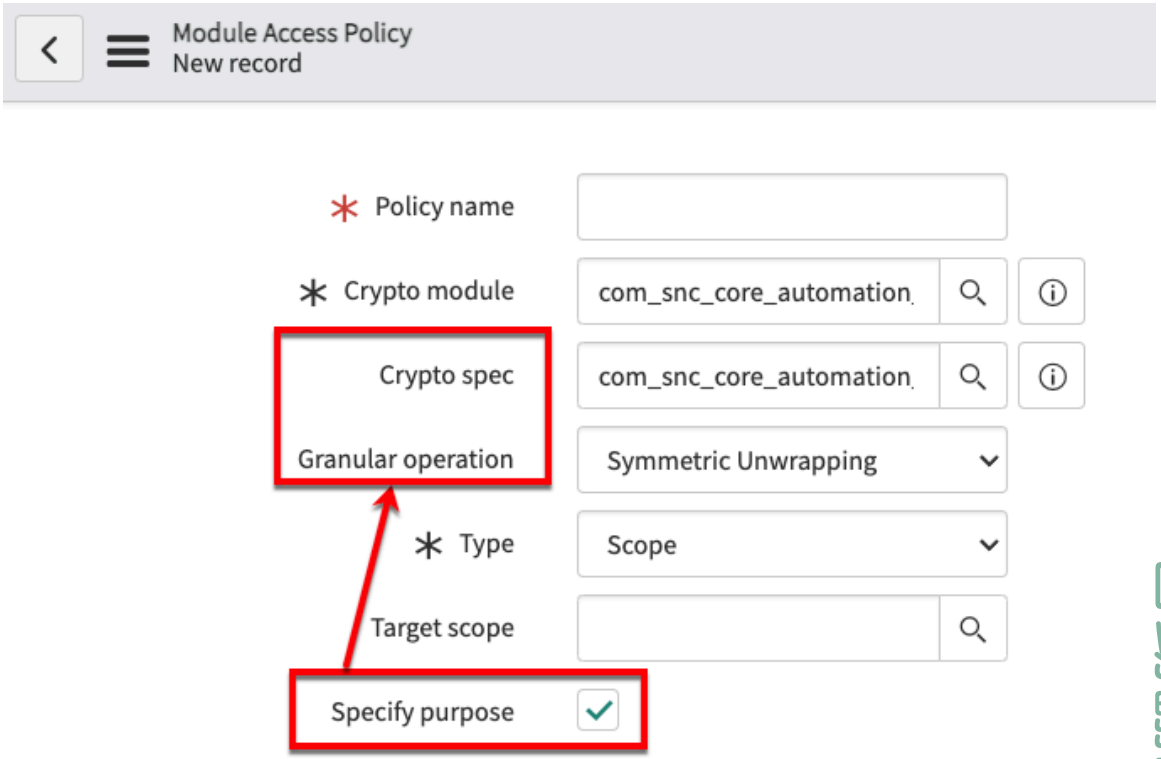
- 対称操作をサポートする暗号化モジュール向けに、モジュールアクセスポリシーで特定の暗号化操作を設定します。たとえば、ユーザーはデータの暗号化を有効にし、復号化は行わないことができます。
- デフォルトのモジュールアクセスポリシー値を設定するか、暗号化モジュールに従います。
- スクリプトの変更を追跡するスクリプトバージョンを関連付けてスクリプトポリシーを無効にし、スクリプトタイプのモジュールアクセスポリシーのセキュリティを強化します。

CLE_Ent 機能は有料サブスクリプションで利用できます。サポート対象機能と各製品で使用できるオプションについては、「暗号化とキー管理のサブスクリプションバンドル」を参照してください。詳細については、「フィールド暗号化エンタープライズ」を参照してください。

注: MAP レコードで明示的に宣言されていない限り、モジュールアクセスポリシー (MAP) のデフォルトの動作は、不正アクセスを防ぐために [拒否] です。

手順

1. 移動先 すべて > キー管理 > モジュールアクセスポリシー > すべて。
   対称データの暗号化/復号化用に構成された暗号化モジュールを作成しない場合は、自動生成されたモジュールアクセスポリシーが作成され、テーブルにリストされます。
2. **[New** (新規)] を選択します。
   - [目的を指定] を選択して [暗号化仕様] を選択し、[詳細な操作] を設定しま



す。

- 対称データの暗号化/復号化および対称ラッピング/ラップ解除の暗号化仕様では、[目的を指定] チェックボックスをオンにすると [詳細な操作] フィールドを使用できます。

3. フォームに入力します。

[モジュールアクセスポリシー] フィールド

フィールド	説明
ポリシー名	ポリシーの名前を入力します。
暗号化されたモジュール	検索アイコン (を選択してモジュールを選択します。
暗号化仕様	モジュールアクセスポリシーの生成時に暗号化仕様を選択または作成します。このフィールドは、[目的を指定] チェック ボックスがオンの場合に利用可能です。
詳細な操作	暗号化仕様の暗号化の目的を選択します。使用可能な値は、選択した暗号化仕様のタイプによって異なります。 暗号化の目的の詳細については、「暗号化の目的、アルゴリズム、およびキーの情報」を参照してください。
タイプ	○ スコープ：アプリケーションスコープ別にアクセスを制御します。 ○ システムユーザー：システムユーザーが暗号化モジュールにアクセスできるようにします。

フィールド	説明
	○ スクリプト：スクリプトでアクセスを制御します。詳細については、「暗号化されたデータへのスクリプトアクセスを設定する」を参照してください。 ○ ロール：ユーザーロール別にアクセスを制御します。 ○ リソース交換：リソース交換 を使用してアクセスを制御します。詳細については、を参照してください。 **注:** フィールド暗号化ではロールタイプのみがサポートされています。他のすべてのタイプは フィールド暗号化エンタープライズ で使用できます。
ターゲットスコープ	フィールドはスコープタイプの識別子として表示されます。ポリシーの機能を参照します。検索メニューでアプリケーションを選択します。 **注:** ターゲットスコープはサポートされておらず、フィールド暗号化エンタープライズ でのみ設定できます。
ターゲットロール	フィールドはロールタイプの識別子として表示されます。このポリシーが適用されるロール。
スクリプトテーブル ターゲットスクリプト	このフィールドは、タイプとして [スクリプト] を選択した場合に表示されます。 フィールドはスクリプトタイプの識別子として表示されます。このポリシーが適用されるテーブルを選択します。このポリシーが適用されるドキュメント。[テーブル名] を選択してから、ポリシーの関連ドキュメントを選択します。 スクリプトが暗号化モジュールを初めて呼び出すと、モジュールへのアクセスが拒否され、開発者はエラーを受け取ります。このエラーにより、モジュールオーナーはモジュールへのアクセスを許可または拒否できます。
リソース交換： ○ 暗号化仕様 ○ 承認タイプ ○ ターゲットインスタンスホスト	これらのオプションは、タイプに [リソース交換] を選択すると表示されます。 親モジュールが **column_level_encryption** の場合、リソース交換 は KMF と両方でサポートされます。 暗号化仕様、1 回または繰り返し、およびターゲットインスタンスの URL を選択します。詳細を参照してください。
代理操作	ロールベースのモジュールアクセスポリシーでは、ユーザーは代理操作セッションを使用して暗号化されたデータにアクセスできます。アドミンなどのユーザーが他のユーザーの代理操作を行うと、そのような代理操作が有効なモジュールアクセスポリシーが適用されます。
目的を指定	[暗号化仕様] フィールドをポリシーで使用可能なフィールドとして切り替える場合に選択します。
アクティブ	ポリシーを有効にする場合に選択します。

フィールド	説明
結果	次のいずれかを選択します。 ○ **StrictReject** は、すべての状況下でアクセスを拒否します。 ○ 却下：ターゲットロールまたはターゲットスコープを持つユーザーは、別のポリシーで許可されない限り、この暗号化モジュールにアクセスできません。 ○ 追跡してモジュールのアクセスを許可し、モジュールの使用を監視します。

4. [送信] を選択します。

⚠ 警告:

従来の暗号化サポートユーザーの場合：

エンタープライズ以外のバージョンの フィールド暗号化 を使用している場合は、5 つのモジュールに制限されます。この制限を超えると、次の警告が表示されます。

この挿入は、サブスクリプション製品のエンタイトルメントが付与されているフィールド暗号化 に許可されている公開モジュールの数を超えています。追加のモジュールには、 フィールド暗号化 の Enterprise サブスクリプションが必要です。アカウントチームにお問い合わせください。

5. 調べる暗号化モジュールに関連付けられているポリシー名を選択します。
スクリプトタイプモジュールアクセスポリシーの使用：

スクリプトが実行されると、デフォルトのアクセス設定に基づいてモジュールアクセスポリシーが自動生成されます。モジュール名の先頭には「AutoGen-」が付きます。たとえば、「Module-TestPolicy」モジュールは、[ポリシー名] 列に「AutoGen-Module-TestPolicy」と表示されます。

暗号化発信者ポリシーフォームに、選択した発信者ポリシーがリストされます。[ターゲットスコープ] フィールドで、モジュールを使用しようとするスクリプトのスコープを指定します。詳細については、「暗号化されたデータへのスクリプトアクセスを設定する」を参照してください。

ℹ 注: フィールド暗号化 では最大 5 つのモジュールアクセスポリシーが許可されます。設定オプションについては、「暗号化とキー管理のサブスクリプションバンドル」を参照してください。

対称暗号化の詳細なロールモジュールアクセスポリシーの作成

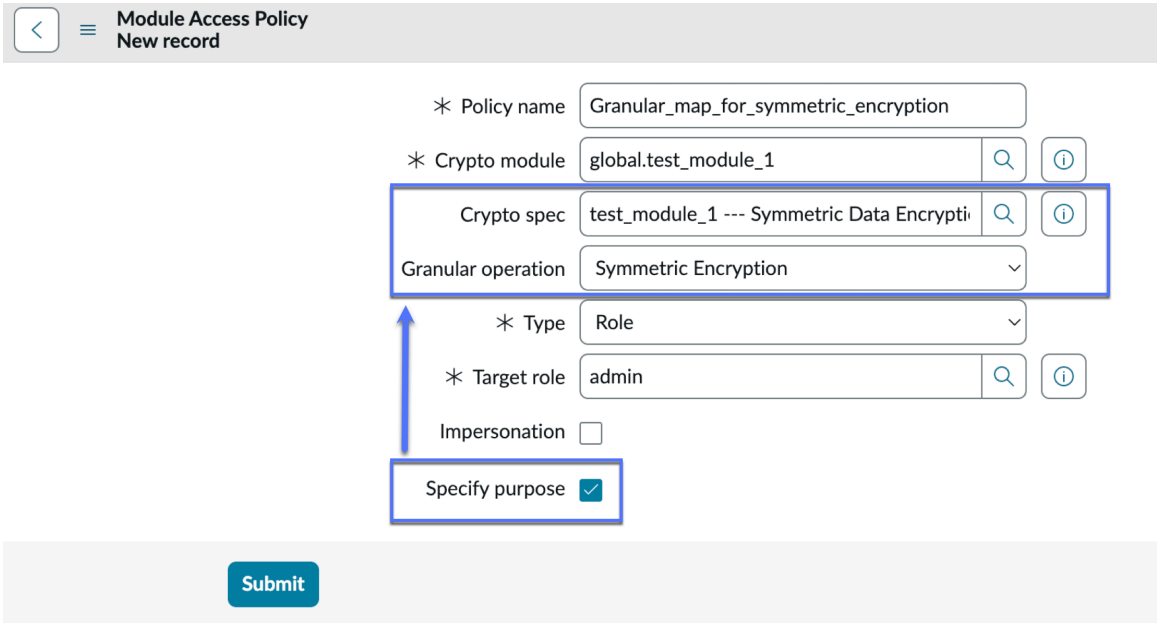
詳細なロールモジュールアクセスポリシー (MAP) を作成してデータを保護すると同時に、特定の MAP にアサインされていないユーザーが公開レコードでフォームを送信できるようにします。

始める前に
必要なロール：security_admin

手順

1. 移動先 すべて > キー管理 > モジュールアクセスポリシー > すべて.
2. **[New** (新規)] を選択します。
3. ポリシーの名前を入力します。
4. アクセス権を付与する 暗号化モジュール を選択します。

5. 適切な ターゲットロールを選択します。
6. [目的を指定] ボックスを選択し、新しく表示されたフィールドに目的の詳細を次のように入力します



- 暗号化仕様:使用する暗号化仕様を選択します。
- 詳細な操作:ドロップダウンリストから [対称暗号化] を選択します。

7. [アクティブ] ボックスを選択します。
8. **[Result]** で **[Track]** を選択します。

結果
詳細なロール MAP を使用すると、MAP にアサインされていないユーザーが、データを保護しながら公開レコードフォームを送信できます。

暗号化モジュールライフサイクルポリシーを作成する

暗号化モジュールのライフサイクルポリシーを作成し、キーの有効期間など、暗号化モジュールに制限を設けます。公開を制限して暗号化モジュールを保護するポリシーを作成します。

始める前に
必要なロール：sn_kmf.cryptographic_manager

このタスクについて
暗号化モジュールライフサイクルポリシーは、インスタンスレベルのポリシーです。暗号化キーが公開される機会が多いほど、侵害される可能性が高くなります。キーの使用期間と使用者を制限して、キーを保護します。

暗号化モジュールは次の機能で制御されます。

- インスタンスポリシーは、インスタンスの境界を設定します。たとえば、有効期限がアクティブ化日から 2 年を超えないようにインスタンスポリシーで指定した場合、ライフサイクルルールを使用してアクティブ化日から 5 年後に有効期限を設定することはできません。

- インスタンスライフサイクルテンプレートを使用すると、異なるキーに異なるポリシーを設定できます。テンプレートには暗号化モジュールのデフォルトのライフサイクルルールが用意されているため、モジュールごとに作成し直す必要はありません。たとえば、対称データ暗号化キーには、公開鍵ラッピングキーとは異なる有効期限を設定できます。

- ライフサイクルルールはキーに直接影響します。たとえば、ライフサイクルルールで有効期限がアクティブ化日から 2 年後に設定されている場合、キーはアクティブ化日から 2 年後に期限切れになります。

手順

1. 移動先 すべて > キー管理 > ライフサイクルポリシー > インスタンスポリシー.
2. 選択 新規.
3. フォームを完了します。

暗号化ライフサイクルポリシーフィールド

フィールド	説明
適用先	読み取り専用。ライフサイクルが適用されるキー。
アクティブ	ポリシーを有効にする場合に選択します。
ポリシー条件	暗号化モジュールをアクティブ化、更新、非アクティブ化、および破棄するタイミングを指定する条件ステートメント。
結果	[却下] で暗号化モジュールへのアクセスを取り消すか、[追跡] で暗号化モジュールの使用を許可して監視します。

次のタスク

このライフサイクルポリシーにモジュールレベルで例外を追加する場合は、「モジュールライフサイクルポリシー例外の作成」を参照してください。

モジュールライフサイクルポリシー例外の作成

モジュールポリシー例外を作成して、1 つのインスタンス上の特定のキーについてのみキーのライフサイクルポリシーを変更します。

始める前に

必要なロール：sn_kmf.cryptographic_manager および sn_kmf.admin

例外はそのモジュールにのみ適用され、インスタンス全体には適用されません。たとえば、アドミニストレーターがインスタンスレベルで対称キーを 1 年間に制限するように設定したとします。モジュールレベルでは、2年間の例外を設けることができます。

手順

1. 移動先 すべて > キー管理 > 暗号化モジュール すべて.
2. ポリシー例外を使用する暗号化モジュールを選択します。
3. [暗号化モジュール] テーブルで、 モジュールポリシー例外 確認してください。
4. 選択 新規.
5. フォームを完了します。

モジュールポリシー例外フィールド

フィールド	説明
暗号化されたモジュール	選択したモジュールの名前。このフィールドは読み取り専用です。
適用先	指定されたキーが自動入力されます。
キータイプ	例外ポリシーが関連するキータイプ。 注: 選択できるキータイプは 1 つだけですが、暗号化モジュールごとに複数の例外ポリシーを作成できます。
ポリシー条件	ポリシー例外が適用されるタイミングを決定するカスタマイズ可能な条件。
結果	[ポリシー条件] フィールドの条件が満たされたときに発生する結果。 ○ 却下 :キーの使用を却下します。 ○ [トラック] では、キーを使用できます。

6. 選択 送信 暗号化モジュールテーブルに戻されます。

キー管理フレームワークリファレンス

キー管理フレームワーク (KMF) API/UX を使用すると、ServiceNow インスタンスでの暗号化操作の実行方法を完全にカスタマイズして管理できます。ServiceNow キー管理フレームワーク は、インスタンス側の暗号化キー管理サービスのための安全で包括的なインターフェイスを提供します。

キー管理フレームワークの主要なライフサイクル状況

KMF は、特定の許容可能なアクションを適用することで、いくつかの暗号化キーのライフサイクル状況をサポートします。たとえば、アクティブ状況のキーのみが、意図した暗号化目的のために十分使用することができます。次の表に、さまざまなキーのライフサイクル状況の詳細を示します。

キー管理フレームワークとともにインストールされるロール

キー管理フレームワーク (KMF) では、暗号化モジュールの特定のロールとキー管理関連の設定が導入されます。

モジュールアクセスポリシーのビジュアル化

モジュールアクセスポリシーのビジュアル化を使用して、関連するすべての暗号化モジュール情報を単一の UI ページに表示します。

モジュールアクセスポリシーデバッガ

モジュールアクセスポリシーデバッガを使用してログ記録情報を確認し、ユーザーに暗号化コンテキストへのアクセスが許可される理由と許可されない理由を把握します。

暗号化とキー管理のサブスクリプションバンドル

キー管理により、 フィールド暗号化 は追加料金なしで高度に構成可能な暗号化モジュールにアップグレードされます。無制限に使用できるライセンスにアップグレードすることもできます。フィールド暗号化エンタープライズおよびクラウド暗号化を含む、新しい暗号化エンタイトルメントバンドルである Platform Encryption に登録することができます。

キー管理フレームワークの主要なライフサイクル状況

KMF は、特定の許容可能なアクションを適用することで、いくつかの暗号化キーのライフサイクル状況をサポートします。たとえば、アクティブ状況のキーのみが、意図した暗号化目的のために十分使用することができます。次の表に、さまざまなキーのライフサイクル状況の詳細を示します。

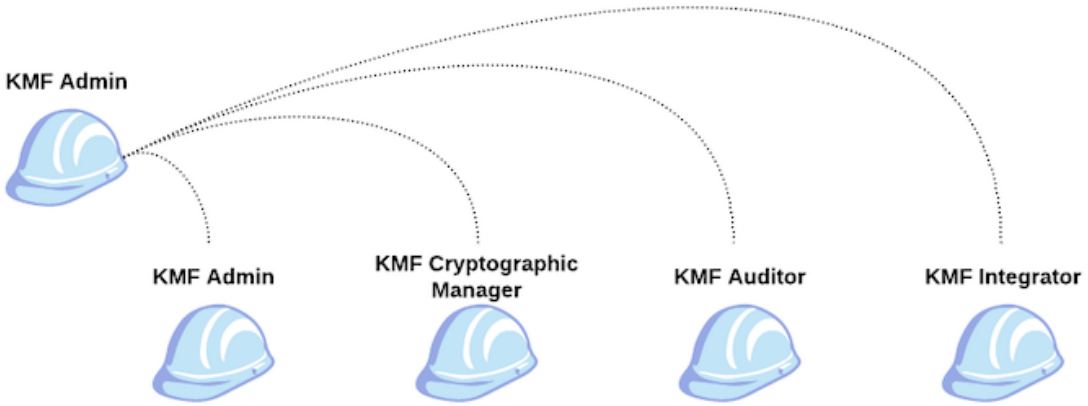
キーのライフサイクル状況またはアクション	説明
アクティブ	アクティブなキーは、暗号化や署名などの新しいコンテンツを生成するために使用されます。暗号化モジュール内の特定の暗号化仕様に対して有効なキーは 1 つだけです。
侵害されています	侵害されたキーは、暗号化や署名などの新しいコンテンツを生成するために使用することはできませんが、復号化や検証などの既存のコンテンツの目的を特定するために使用することはできます。 暗号モジュールの特定の暗号化仕様では、複数のキーが侵害された状態で存在する可能性があります。アクティブなキーまたは一時停止されたキーは、侵害された状態に移行する可能性があります。
非アクティブ化済み	任意のアクティブなキーを非アクティブ化できます。暗号モジュールの特定の暗号化仕様では、複数のキーが非アクティブ化された状態で存在する可能性があります。 たとえば、キーがローテーションされると、現在アクティブなキーが非アクティブになります。非アクティブ化されたキーは、暗号化や署名などの新しいコンテンツを生成するために使用することはできませんが、復号化や検証などの既存のコンテンツの目的を特定するために使用することはできます。 **注:** 侵害されたキーや失効したキーは、非アクティブ化されたキーとして扱われます。
破壊	キーが破棄されると、キーマテリアルは完全に削除され、暗号化の目的のために使用できなくなります。非アクティブ化されたキーは、設定された指定期間内に使用されなかった場合、ライフサイクルの自動化を使用して破棄できます。暗号モジュールの特定の暗号化仕様では、複数のキーが破壊された状態で存在する可能性があります。 **警告:** 破棄されたキーに関連付けられたデータにはアクセスできなくなるため、キーを破棄するアクションを実行するときは注意が必要です。
生成済み	暗号モジュールの特定の暗号化仕様では、複数のキーが生成された状態で存在する可能性があります。 指定された暗号化仕様に対応するアクティブなキーが存在しない場合、生成されたキーをアクティブステータスに移行できます。生成された最初のキーは自動的にアクティブに設定されます。

キーのライフサイクル状況またはアクション	説明
	注: 新しいキーの生成を選択した場合は、指定された暗号化仕様に生成済みステータスのキーがある場合でも、新しいキーが生成されてアクティブになります。
更改済み	有効期限のあるアクティブなキーは、キーのライフサイクル期間を延長するために何回でも更新できます。 **注:** アクティブ化日と有効期限の差が計算され、有効期限が当日から延期されます。
再開	UI アクションは、指定された暗号化仕様に他のアクティブなキーが存在しない場合に、一時停止されたキーで使用して、アクティブ状態に戻すことができます。
取り消し済み	アクティブなキーまたは一時停止されたキーは、取り消された状態に移行する可能性があります。 取り消されたキーは、暗号化や署名などの新しいコンテンツを生成するために使用することはできませんが、復号化や検証など、既存のコンテンツの目的を特定するために使用することはできます。 暗号モジュールの特定の暗号化仕様では、複数のキーが取り消された状態で存在する可能性があります。
ローテーション済み	キーのローテーションにより、現在のアクティブなキーが非アクティブ化され、別のキーがアクティブになります。新しいアクティブなキーを次から選択します。 • 新しいキーの生成。 • 既存のインポートされたキーをポイントします。任意のアクティブなキーをローテーションできます。
一時停止	暗号モジュールの特定の暗号化仕様では、複数のキーが一時停止された状態にある可能性があります。キーが一時停止され、その暗号化仕様に他のアクティブなキーが存在しない場合は、キーを再開してアクティブ状況に再アサインできます。

キー管理フレームワークとともにインストールされるロール

キー管理フレームワーク (KMF) では、暗号化モジュールの特定のロールとキー管理関連の設定が導入されます。

重要: KMFキー管理フレームワークを使用するにはロールが必要です。KMFロールを持たないユーザーは、キー管理の構成に使用されるリスト、テーブル、およびモジュールにアクセスできません。



KMF アドミン **[sn_kmf.admin]**

ServiceNow キー管理フレームワークに関する操作を実行するためのロールを他のユーザーにアサインします。

ロールを含む

ロール内に含まれるロールのリスト。

なし。

グループ

このロールがデフォルトでアサインされているグループのリスト。

なし。

特別な考慮事項

重要: より分化したロールが利用可能な場合は、アドミンロールを付与しないでください。

- このロールは、キー管理フレームワーク ロールの割り当てに示すプロセスを介してアサインされます。
- このロールを持つユーザーには、admin と security_admin も必要です
- KMF ロールをアサインし、さらに KMF 暗号化マネージャーのすべての機能を実行するには、このロールが必要です。

KMF 暗号化マネージャー **[sn_kmf.cryptographic_manager]**

暗号化モジュールとモジュールアクセスポリシーに対する作成、読み取り、更新操作 (暗号化使用およびアルゴリズム構成へのキーの関連付け) を実行します。また、KMF 暗号化マネージャーは、キー管理 (生成、ローテーション、取り消し) およびライフサイクル操作を実行できます。

ロールを含む

ロール内に含まれるロールのリスト。

なし。

グループ

このロールがデフォルトでアサインされているグループのリスト。

なし。

特別な考慮事項

このロールは、KMF アドミンのみがユーザーにアサインできます。

KMF 暗号化監査人 **[sn_kmf.cryptographic_auditor]**

暗号化モジュール情報、キーメタデータ、ライフサイクル関連の詳細、およびモジュールアクセスポリシー (MAP) 情報を表示します。

ロールを含む

ロール内に含まれるロールのリスト。

なし。

グループ

このロールがデフォルトでアサインされているグループのリスト。

なし。

特別な考慮事項

このロールは、KMF アドミンのみがユーザーにアサインできます。

KMF 暗号化インテグレーター **[sn_kmf.cryptographic_integrator]**

キー管理フレームワーク を外部のキーストアやシステムと統合します。

ロールを含む

ロール内に含まれるロールのリスト。

なし。

グループ

このロールがデフォルトでアサインされているグループのリスト。

なし。

特別な考慮事項

このロールは、KMF アドミンのみがユーザーにアサインできます。

KMF 暗号化演算子 **[sn_kmf.cryptographic_operator]**

ServiceNow キー管理フレームワーク キーライフサイクルの一部 (更新、ローテーション、取り消し) にアクセスします。

ロールを含む

ロール内に含まれるロールのリスト。

なし。

グループ

このロールがデフォルトでアサインされているグループのリスト。

なし。

特別な考慮事項

なし。

KMF ロールの割り当て

アドミニストレーターに KMF ロールを割り当て、そのアドミニストレーターは他の KMF ロールを割り当てることができます。

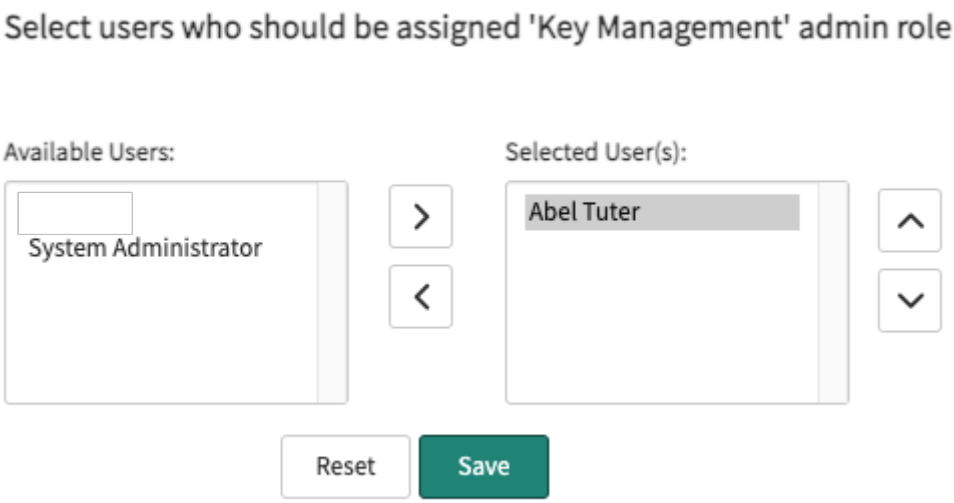
始める前に

必要なロール：admin および security_admin

KMF admin ロールを割り当てる前に、security_admin ロールに昇格させる必要があります。詳細については、「特権ロールへの昇格」を参照してください

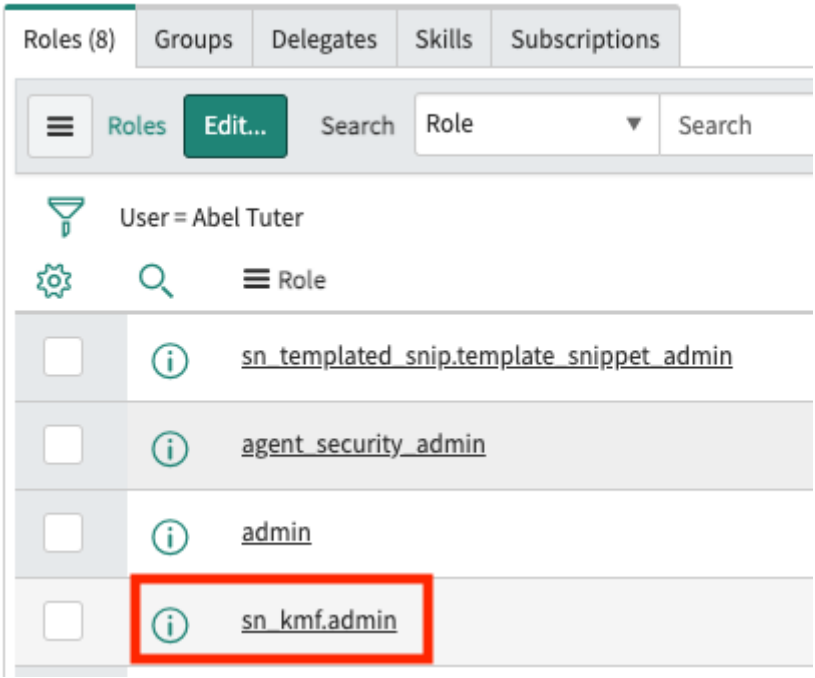
手順

1. セキュリティ管理者ロールに昇格します。
2. 移動先 ユーザー管理 > ユーザー をクリックし、 KMF アドミンにするユーザーを選択します。
3. ユーザーが既に admin ロールと security_admin ロールを持っていることを確認します。
   そうでない場合は、ロール関連リストで [編集] を選択し、**admin** と **security _admin** を追加します。
4. 移動先 システムセキュリティ > キー管理の管理.
5. [利用可能なユーザー] 列で KMF admin にするユーザーを選択し、[選択したユーザー] 列に移動します。



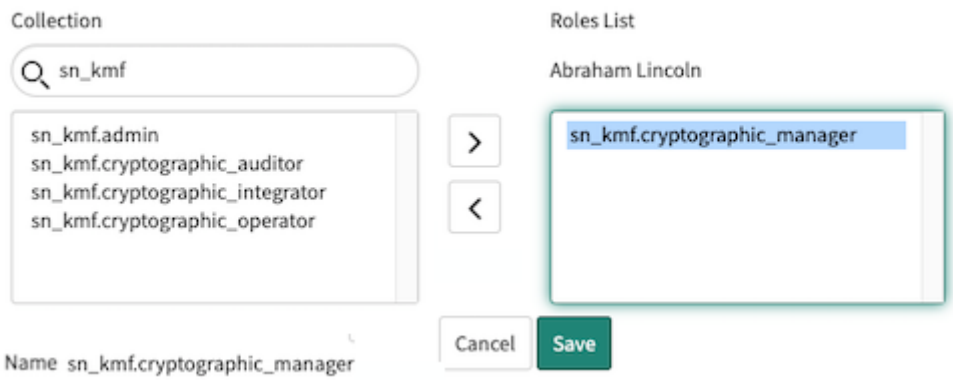
6. **[Save** (保存)] を選択します。
7. 移動先 ユーザー管理 > ユーザー をクリックし、sn_kmf.admin ロールを付与したユーザーを選択します。

ユーザーには [ロール] 関連リストに sn_kmf.admin ロールがあり、他の KMF ロールをアサインできます。



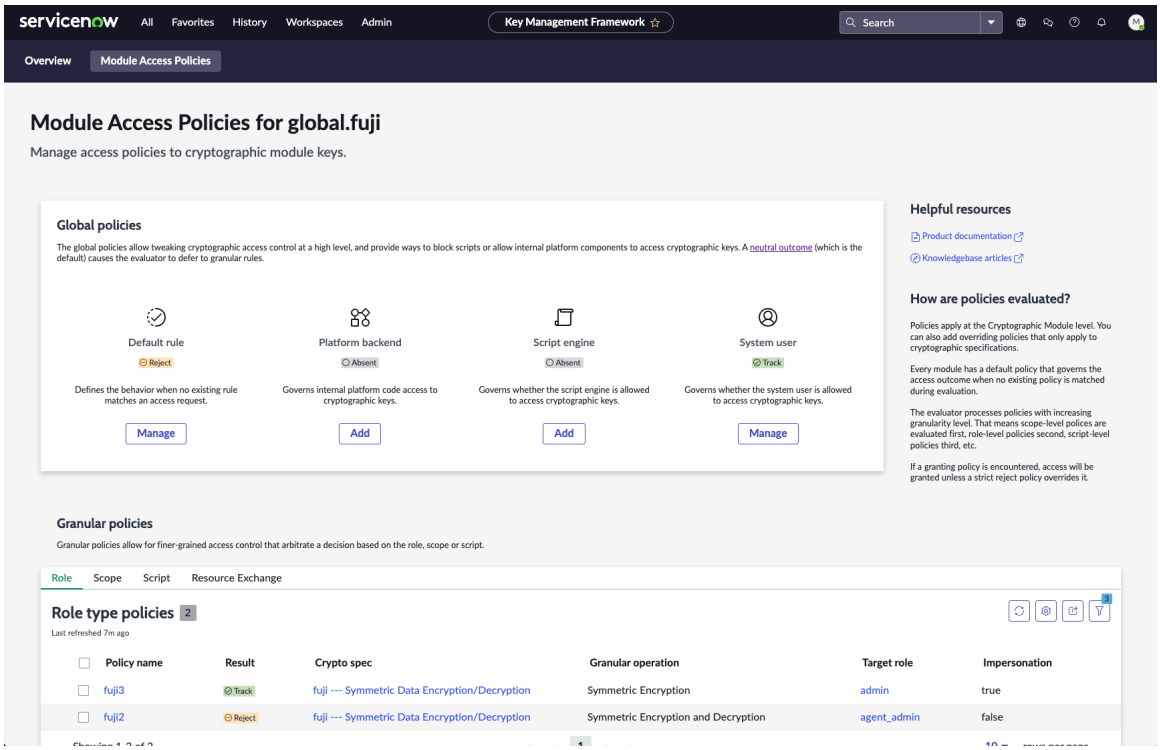
次のタスク
KMF admin ロールを持っている場合は、次の手順に従って他の KMF ロールを割り当てます。

1. 移動先 ユーザー管理 > ユーザー をクリックし、KMF Cryptographic Manager などの別の KMF ロールを持つユーザーを選択します。
2. ロール関連リストで、[編集] を選択し、ユーザーに割り当てる KMF ロールを選択します。すべての KMF ロールは sn_kmfで始まります。



モジュールアクセスポリシーのビジュアル化

モジュールアクセスポリシーのビジュアル化を使用して、関連するすべての暗号化モジュール情報を単一の UI ページに表示します。



キー管理フレームワークのアドミニストレーターと暗号化マネージャーは、モジュールアクセスポリシー UI ページを使用して、単一の暗号化モジュールに関連するすべてのアクセス制御メカニズムを表示できます。この UI ページで収集された情報を使用して、インスタンスの暗号化された情報にアクセスできるユーザーを決定します。

sn_kmf.admin または *sn_kmf.cryptographic_manager* ロールを持つユーザーは、次の場所に移動してモジュールアクセスポリシーの可視化 UI ページにアクセスできます すべて > キー管理 > 暗号化モジュール > すべて.

結果ラベル

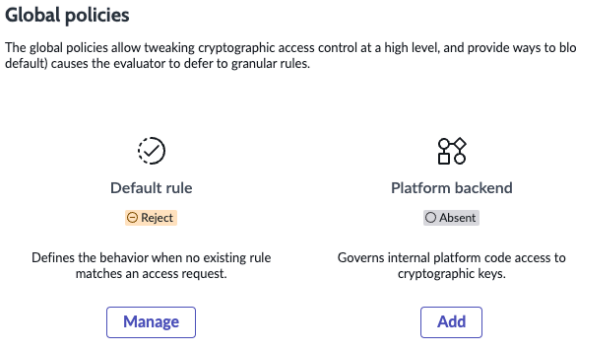
モジュールアクセスポリシーには、選択した暗号化モジュールへのアクセスを許可するかどうかを決定する [結果] フィールドが含まれています。UI ページには、そのフィールドの値に基づいて UI ページの要素にラベルが表示されます。

UI ラベル	結果フィールド値	定義
Track	*Track* または *Allow*	すべてのユーザーにスクリプトを含むアクセスが許可されます。
Reject	*Reject*	追跡モジュールアクセスポリシーが見つからない場合、アクセスは拒否されます。
StrictReject	*StrictReject*	アクセスは拒否されます。
Absent	*N/A*	モジュールアクセスポリシーがインスタンスに存在しません。すべてのアクセスは拒否されます。

グローバルポリシー

[グローバルポリシー **(Global policies)**] セクションを使用して、プラットフォームレベルのアクセスを制御するモジュールアクセスポリシーを確認します。

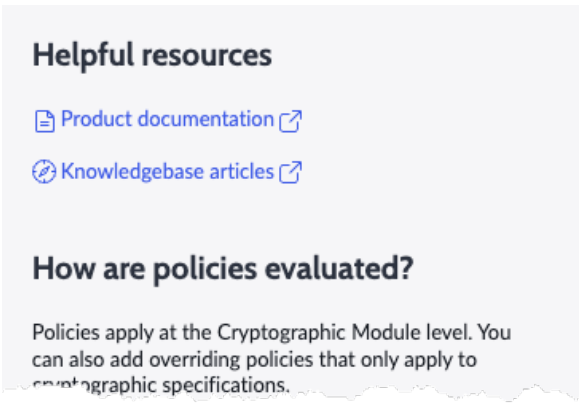
いずれかのポリシーの下にある [管理] ボタンを選択して、そのポリシーレコードに移動します。ポリシーが存在しない場合は、そのエントリの下に [追加] ボタンが表示されます。[追加] ボタンを選択して、ポリシーを定義できる新しいポリシーレコードに移動します。



ポリシー	定義
デフォルトルール	デフォルトのルールポリシーは、アクセス要求に一致する既存のルールがない場合の動作を定義します。
プラットフォームバックエンド	プラットフォームバックエンドポリシーは、暗号化キーへの内部プラットフォームコードアクセスを管理します。
スクリプトエンジン	スクリプトエンジンポリシーは、スクリプトエンジンが暗号化キーへのアクセスを許可されるかどうかを管理します。
システムユーザー	システムユーザーポリシーは、システムユーザーに暗号化キーへのアクセスを許可するかどうかを管理します。

役に立つリソース

[役に立つリソース] セクションを使用して、製品ドキュメント、関連ナレッジ記事、プラットフォームでモジュールアクセスポリシーを評価する方法に関する簡単な説明へのリンクを見つけます。モジュールアクセスポリシーを評価する方法の詳細については、「モジュールアクセスポリシーデバッガ」を参照してください。

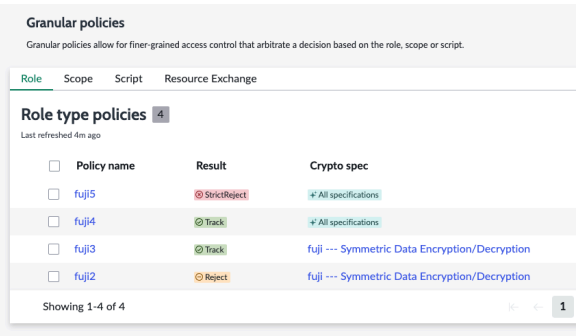


詳細なポリシー

[詳細なポリシー **(Granular policies)**] セクションを使用して、モジュールアクセスポリシーのリストをポリシータイプ別に表示します。リストの上にあるタブを使用して、表示するポリシーカテゴリを選択します。

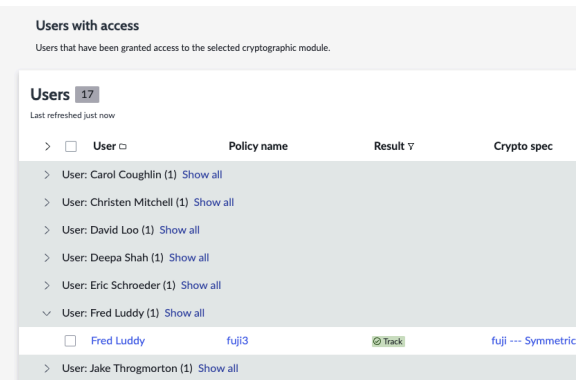
- ロール
- スコープ
- スコープとドメイン (ドメインセパレーションがアクティブな場合)
- スクリプト
- リソース交換 (暗号化モジュールが Password2 または フィールド暗号化 サブモジュールの場合)
- ID (シークレット管理 Enterprise がアクティブな場合)

デフォルトでは、各リストにはアクティブなポリシーのみが表示されます。[フィルター] アイコンを使用して、リストのデフォルトのフィルターを変更します。



アクセス権を持つユーザー

[アクセス権を持つユーザー **(Users with access)**] セクションを使用して、選択した暗号化モジュールにアクセスできるすべてのユーザーのリストを表示します。1 人のユーザーが暗号化モジュールへのアクセスを許可する複数のロールを持てるため、リストはユーザー別にグループ化されています。

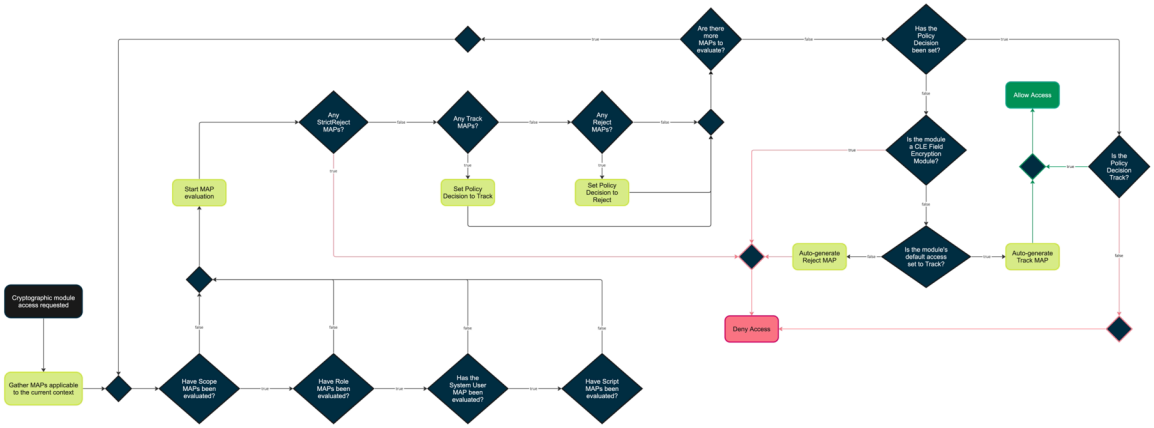


モジュールアクセスポリシーデバッガ

モジュールアクセスポリシーデバッガを使用してログ記録情報を確認し、ユーザーに暗号化コンテキストへのアクセスが許可される理由と許可されない理由を把握します。

モジュールアクセスポリシー (MAP) は、暗号化モジュールへのアクセスに対するインスタンスレベルの制御を定義します。問い合わせユーザー (ユーザーやスクリプトなど) は、暗号化と復号化に暗号化モジュールを使用するために明示的なアクセス権を必要とします。デバッガを使用して、発信者が暗号化モジュールにアクセスを試みたときに評価されるポリシーを確認します。また、デバッガーを使用して、アクセスが許可される理由と付与されない理由を確認することもできます。

このフローチャートは、インスタンスが暗号化モジュールへのアクセス要求を評価する方法を示しています。



デバッグログへのアクセスの制御

モジュールアクセスのデバッグログへのアクセスは、ロールによって決まります。*sn_kmf.admin* ロールと *sn_kmf.cryptographic_manager* ロールを持つユーザーは、デバッガーにアクセスできます。*glide.kmf.module_access_policies.debugger.authorized.roles* システムプロパティを使用して、他のロールにアクセス権を付与します。このプロパティの値は、デバッグログにアクセスできるロールのカンマ区切りリストです。

デバッガの有効化または無効化

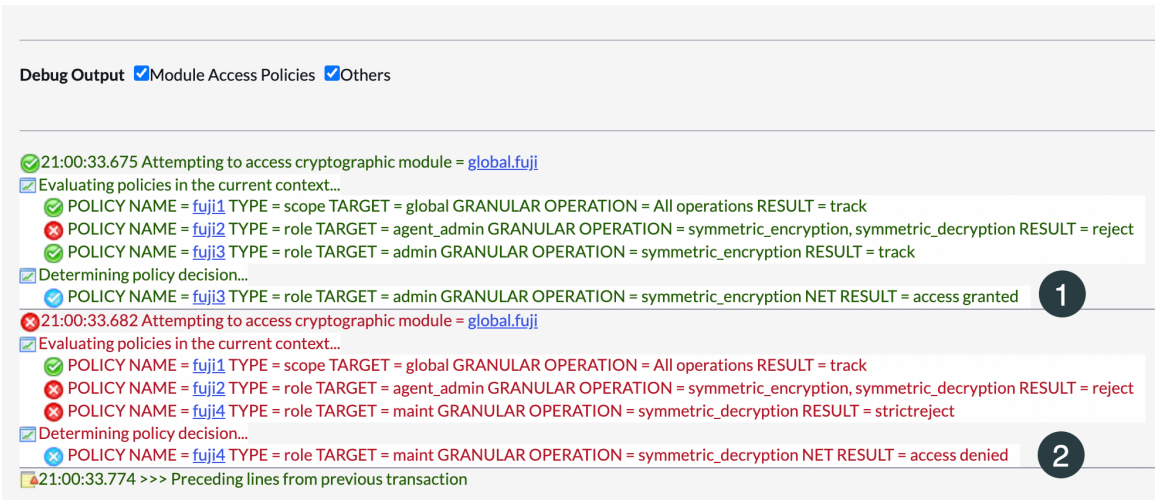
モジュールアクセスポリシーのデバッグログメッセージを有効にするには、 すべて > 診断 > セッションのデバッグ > モジュールアクセスポリシーのデバッグ > .

デバッグが完了したら、 すべて > 診断 > セッションのデバッグ > すべて無効化 > .

ログへのアクセス

デバッグを有効にした後、MAP 評価をトリガーするページに移動して、MAP デバッグログを表示します。デバッグメッセージがページの下部に表示されます。

ヒント: 代理操作を使用して、他のユーザーのアクセスのトラブルシューティングを行うことができます。代理操作の詳細については、「ユーザーの代理操作」を参照してください。別のユーザーの観点からデバッグログを表示するには、*role* タイプのモジュールアクセスポリシーで [代理操作] フィールドが **true** に設定されていることを確認してください。



この例では、問い合わせユーザーが global.fuji 暗号化モジュールへの 2 つのアクセス要求を呼び出します。対称暗号化は許可され、対称復号化は拒否されています。

ログエントリについて

デバッグ情報は、この形式を使用して構造化されます。

1. この最初の行にはアクセス要求を受信する暗号化モジュールが表示されます。
2. 最初の行と最後の行の間にある行には、評価済みの MAP が評価された順に表示され、名前、タイプ、ターゲット、詳細な操作、および結果が含まれています。
3. 最後の行には、ポリシー決定 (該当する場合) と問い合わせユーザーの正味アクセス結果 (問い合わせユーザーにアクセス権が付与されているかどうか) が表示されます。

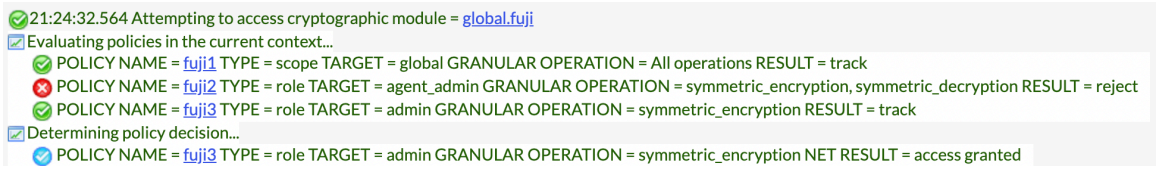
各行はメッセージタイプを示すアイコンで始まります。

メッセージアイコン

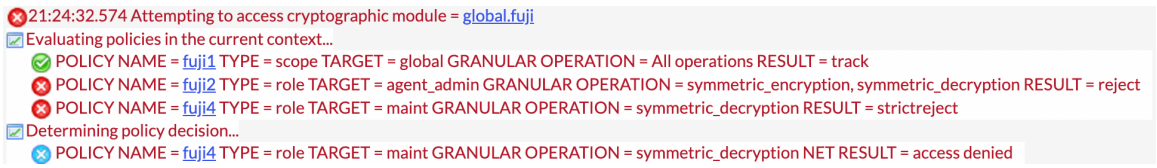
アイコン	メッセージタイプ
	情報メッセージ
	モジュールアクセスポリシーによりアクセス権が付与される
	モジュールアクセスポリシーによりアクセスが拒否される
	問い合わせユーザーのアクセスが許可される
	問い合わせユーザーへのアクセスが拒否される
	評価するモジュールアクセスポリシーなし

デバッグログの例

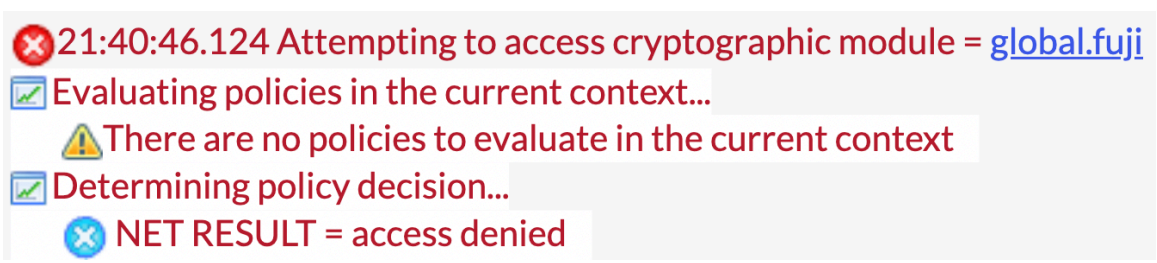
アクセスが許可された場合のメッセージ



アクセスが拒否された場合のメッセージ



アクセスが拒否されました (評価するモジュールアクセスポリシーなし)



アクセスが拒否されました (十分な権限がない)

暗号化とキー管理のサブスクリプションバンドル

キー管理により、 フィールド暗号化 は追加料金なしでアップグレードされ、高度に構成可能な暗号化モジュールが含まれます。必要に応じて、無制限に使用できるライセンスにアップグレードすることもできます。フィールド暗号化エンタープライズおよびクラウド暗号化を含む、新しい暗号化エンタイトルメントバンドルである Platform Encryption に登録することができます。

フィールド暗号化の機能

フィールド暗号化 暗号化モジュール付きはインスタンスに無料で含まれており、 NIST 800-57 キー管理が含まれています。

ServiceNow Platform Encryption グループ機能

Platform Encryption グループは、次の機能と製品を追加します。

- 暗号化。
- キー管理を使用したクラウド暗号化。

追加情報

キー管理の詳細については、「キー管理フレームワークの詳細キー管理フレームワークの詳細」を参照してください。

キー管理アクション

KMF のコア機能の 1 つは、キーの取り消しやローテーションなどのキー管理機能を提供することです。KMF は、最新の暗号化素材とライフサイクル操作を使用して、機密データを適切に保護します。

次の表に、主要なライフサイクル操作と管理アクションの概要を示します。暗号化モジュールの目的は暗号化モジュール設定のデータに適用され、データには影響しません。

キー管理アクション	説明
キーの生成	指定された暗号化モジュールの新しいキーを生成します。最初に生成されたキーはアクティブに設定されます。
キーのローテーション	現在のキーを非アクティブにし、新しいキーを生成します。新しいモジュールキーが current (アクティブ) に設定されます。
キーの取り消し	現在のキーとライフサイクル状況を取り消し済みとしてマークします。暗号化モジュールは、新しいデータに新しいキーを自動生成し、キーのステータスをアクティブに設定します。取り消しは、キーが暗号化に使用されなくなったことを意味します。ただし、復号化には引き続き使用できます。キーを破棄することはできません。

キー管理アクション	説明
キーの一時停止	現在のキーを一時停止としてマークします。暗号化モジュールを再度使用する前に、一時停止したキーを手動で再開するか、一時停止したキーを取り消して新しいモジュールキーを生成します。
キーの再開	一時停止されたキーをアクティブなキーとしてマークします。
キーの更新	現在のキーの有効期間を延長します。[更新] ボタンは、次の状況で使用できるようになります。 • 暗号化マネージャーロールが割り当てられている。 • ライフサイクル状況がアクティブまたは更新にマークされている。 • 有効期限がモジュールライフサイクル定義に設定されている。

キーの表示と管理

キーのステータスを確認して、現在のキーを更新、ローテーション、一時停止、非アクティブ化、破棄するタイミングなどのキーアクションを決定します。

始める前に

必要なロール：sn_kmf.cryptographic_manager

手順

1. 移動先 すべて > キー管理 > 暗号化モジュール > すべて.
2. 暗号化モジュールを選択します。

   暗号化モジュール <モジュール名> フォームが表示されます。
3. [モジュールキー] タブでキーエイリアスを選択して、ライフサイクル <キー名> フォームでキーのステータスを確認します。
4. フィールドはすべて読み取り専用であるため、フォームを確認します。

暗号化ライフサイクルキーフィールド

フィールド	説明
生成日	キーが生成された日付が表示されます。
アクティブ化日	キーがアクティブ化された日付が表示されます。
前回の更改日	キーが最後に更新された日付が表示されます。
前回のローテーション日	キーが最後にローテーションされた日付が表示されます。
非アクティブ化日	キーが非アクティブ化された日付が表示されます。
破棄日	キーが破棄された日付が表示されます。
キーのライフサイクル状況	キーのライフサイクル状況が表示されます。

フィールド	説明
今後のアクティブ化日	将来のキーのアクティブ化日が表示されます。
今後の更改日	今後のキーの更改日が表示されます。
今後のローテーション日	キーローテーションの将来の日付が表示されます。
今後の破棄日	キーが破棄される将来の日付が表示されます。
有効期限	キーの有効期限が切れる日付が表示されます。

5. キーに対してアクションを実行するには、次のいずれかを選択してすぐに有効にします。

- キーの取り消し:キーを非アクティブ化して新しいキーを生成する場合に選択します。キーを取り消す理由を入力します。
- キーのローテーション:現在のキーを非アクティブ化し、代わりに新しいキーを生成する場合に選択します。新しいキーが [モジュールキー] テーブルに表示され、キーのバージョン番号が 1 ずつインクリメントします。詳細をご確認ください。
- キーの一時停止:現在のキーを無効にする場合に選択します。
- キーを再開：一時停止したキーをアクティブなキーとしてマークする場合に選択します。このオプションは、アクティブなキーが一時停止された後にのみ使用できます。

キーのローテーション

セキュリティを強化するために、事前に決定されたスケジュールで暗号化キーをローテーションすることができます。暗号化キーが廃止され、新しい暗号化キーを生成して古いキーと置き換えたときにキーロテーションが行われます。

始める前に

必要なロール：sn_kmf.cryptographic_manager

このタスクについて

暗号化モジュールは、暗号化コンテキストとは異なり、新しいキーを使用して再暗号化するためにレコードのリキーをサポートしています。暗号化モジュールでキーローテーション操作を手動で実行する方法を以下に示します。

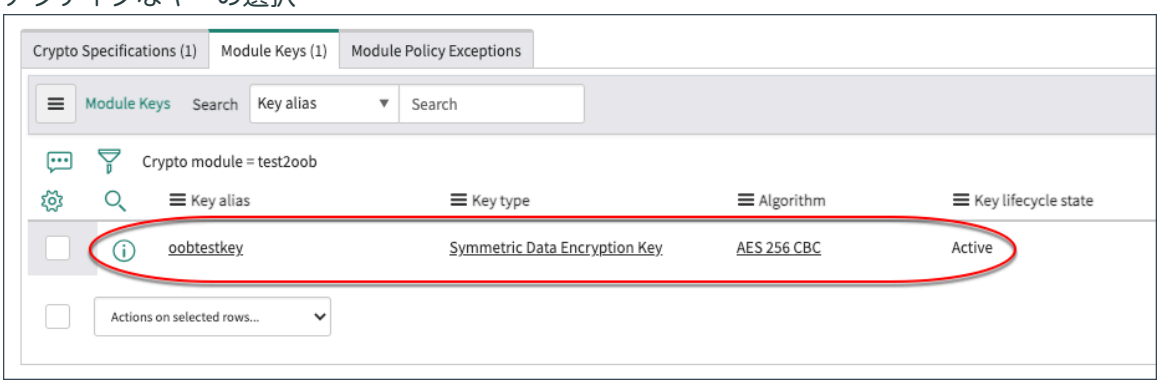
手順

1. 移動先 キー管理 > 暗号化モジュール > すべて.

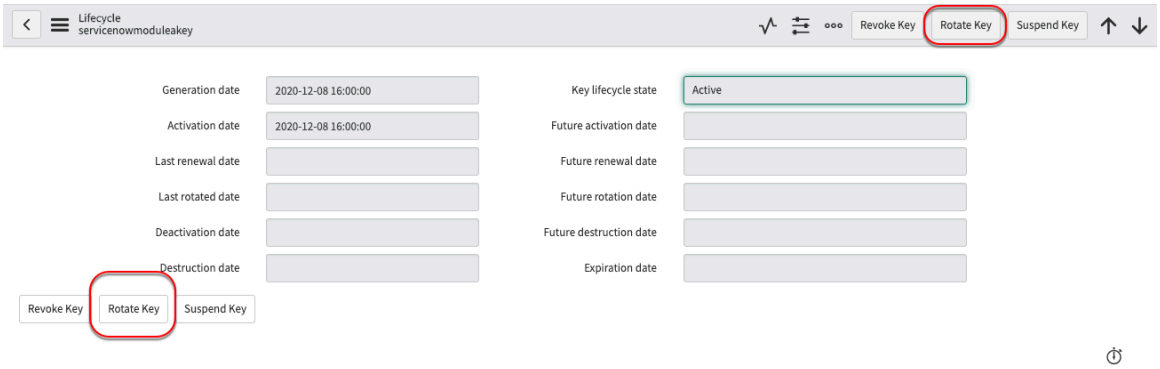
2. キーローテーションの暗号化モジュールを選択します。

3. [モジュールキー] タブで、アクティブなキーを選択します。

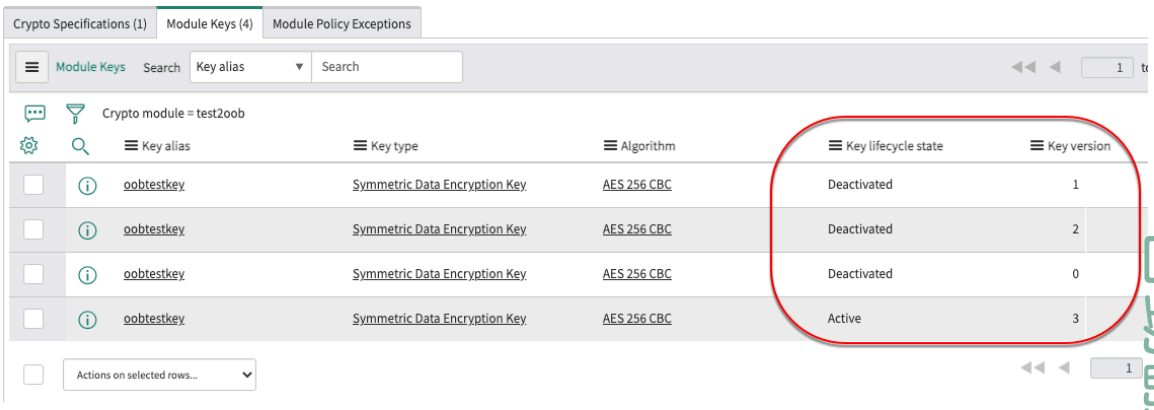
アクティブなキーの選択



自動翻訳



4. [キーのローテーション] を選択します。
   キーのライフサイクル状況が「非アクティブ化済み」に変更されます。[前回のローテーション日]、[非アクティブ化日]、および [キーバージョン] フィールドが更新されます。
5. 戻る 暗号化モジュール > モジュールキー。



テーブルに追加のモジュールキーがリストされています。新しくローテーションされたキーは「アクティブ」になり、最後のキーは「非アクティブ化済み」になります。

Web サービスからのキーのインポート

Web サービス (キー REST API など) からのキーのインポートを使用して、外部顧客キーをインスタンスに安全にアップロードします。対称キーと非対称公開キーの両方をターゲットの KMF 暗号化モジュールにアップロードできます。

インポートするキー (ターゲットキー) は、インスタンスのターゲットの暗号化モジュールにアップロードする前に、ラッピングキーで暗号化する必要があります。このラッピングキーは、公開鍵/秘密鍵のペアの公開コンポーネントであり、インスタンスに存在する必要があります。このキーは、ラップされたターゲットキーが [Web サービスからインポート] を介してアップロードされる前の前提条件です。

2 つの別々の手順 (ラッピングキーペアのインポートと、Web サービスからのラップされたターゲットキーのインポート) について以下のドキュメントで説明します。このキーペアをインスタンスの内部キーインポート暗号化モジュールで使用できるようにするには、キーペアを生成してアップロードする必要があります。

> **注:** この例では、キーと証明書の生成に OpenSSL を使用し、Postman API テストツールを使用して REST API の使用方法が示されています。会社の要件に基づいて、他の同等のツールで置き換えてください。

ラッピング/ラップ解除キーペアのインポート

キーをインポートする前に キー管理フレームワーク インポート設定を構成します。

始める前に

必要なロール：sn_kmf.cryptographic_manager

このタスクについて

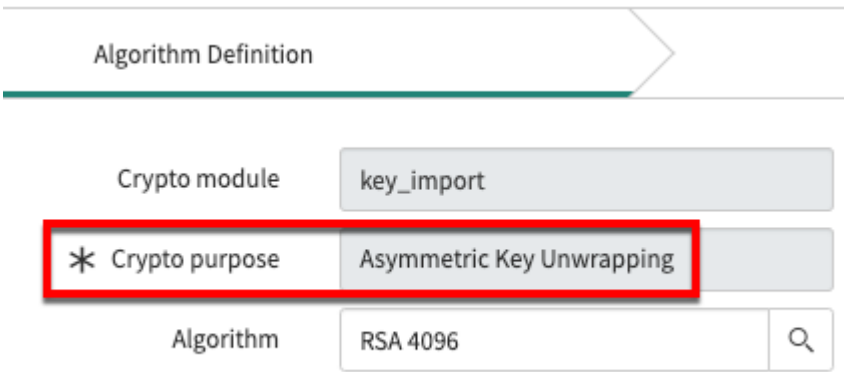
この例では、OpenSSL を使用してキーと証明書を生成します。会社の要件に基づいて、他の同等のツールで置き換えてください。

手順

1. ローカル環境で、ターミナルを使用して証明書を作成します。
   例：`openssl req -x509 -sha256 -nodes -days 365 -newkey rsa:4096 -keyout wrapping_private.key -out wrapping_public.crt`

   この証明書は、キーを含む公開コンポーネントです。証明書は、AES 対称キーをラップするために使用されます。

2. ローカル環境で、ターミナルを使用して公開証明書 (ラッピングキー付き) と秘密ラップ解除キーを含むキーストアを作成します。
   例：`openssl pkcs12 -export -in wrapping_public.crt -inkey wrapping_private.key -name "wrapping_key_alias" -out wrapping_keystore.p12`
3. インスタンスで、 すべて > キー管理 > 設定をインポート > キーインポート設定.
4. [アルゴリズム定義] セクションで、[暗号化の目的] が [非対称キーのラップ解除] に設定されていることを確認します。



5. インポートされたキーストアの非対称キーマテリアルに適したアルゴリズムを選択します。
   詳細については、「暗号化仕様の概要」を参照してください。
6. [次へ] を選択します。
7. [ライフサイクル定義] セクションで、[次へ] を選択して続行します。
8. [キーの作成元] セクションの、[作成元] フィールドで **[PKCS12 からインポート]** または **[BCFKS からインポート]** を選択します。

   注: ステップ 1 のキーストア例を使用する場合は、**[PKCS12 からインポート]** を選択します。

9. キーを識別できるようにキーエイリアスを入力します。
   このエイリアスは、アップロードする証明書またはキーストアを生成するときに指定したキーエイリアス (または「分かりやすい名前」) と一致する必要があります。上記の例を続けた場合、これは `wrapper_key_alias` になります。

10. [次へ] を選択します。
    [キーの作成] セクションには **[** キーのインポート **]** リンクが含まれ、キーストアをアップロードするためのダイアログが表示されます。この例を続けた場合、これは wrapping_keystore.p12 になります。

Web サービスからラップされたキーをインポートする

Web サービスからのキーのインポート機能を使用して、ラップされたキーを暗号化モジュールにアップロードします。この例では、対称キーを使用しています。同様の手順を使用して非対称キーをインポートできます。

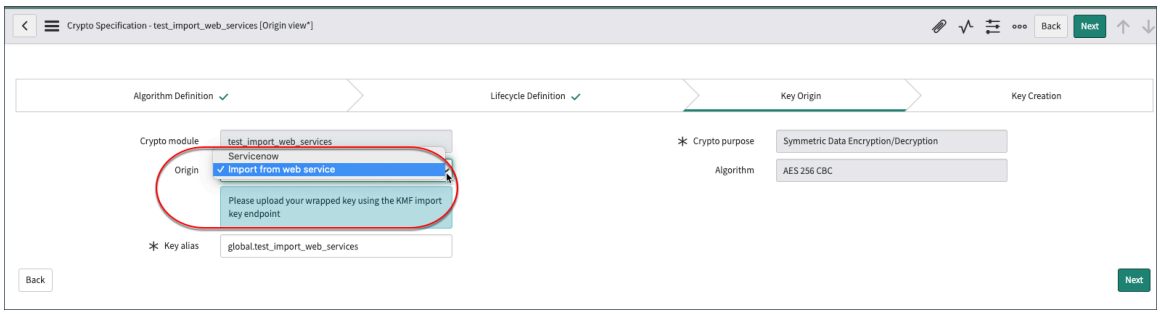
始める前に
必要なロール：sn_kmf.cryptographic_manager (モジュール構成)、sn_kmf.cryptographic_operator (REST 操作のベーシック認証)

このタスクについて
キーのインポートプロセスを完了するには、KMF インポートキーエンドポイントにアクセスする必要があります。

この例では、OpenSSL を使用してキーと証明書を生成します。要件に基づいて、他の同等のツールに置き換えることもできます。

手順

1. ローカルデバイスのターミナルで、キーインポートモジュールの公開鍵ラッピングキーを使用して対称キーをラップします。
   例：openssl pkeyutl -encrypt -pubin -inkey public_wrapping_key.pem -in symmetric_key.bin -pkeyopt rsa_padding_mode:oaep -pkeyopt rsa_oaep_md:sha256 -out wrapped_symmetric_key.txt
   この例では、wrapped_symmetric_key.txt という名前のラップされたキーファイルが作成されます。
2. API に関連付ける暗号化モジュールを作成します。
   詳細をご確認ください。
3. 次の選択肢を使用して暗号化仕様を追加します。
   - 暗号化の目的：対称データの暗号化/復号化
   - キーの作成元：*Web* サービスからインポート



   詳細を参照してください。

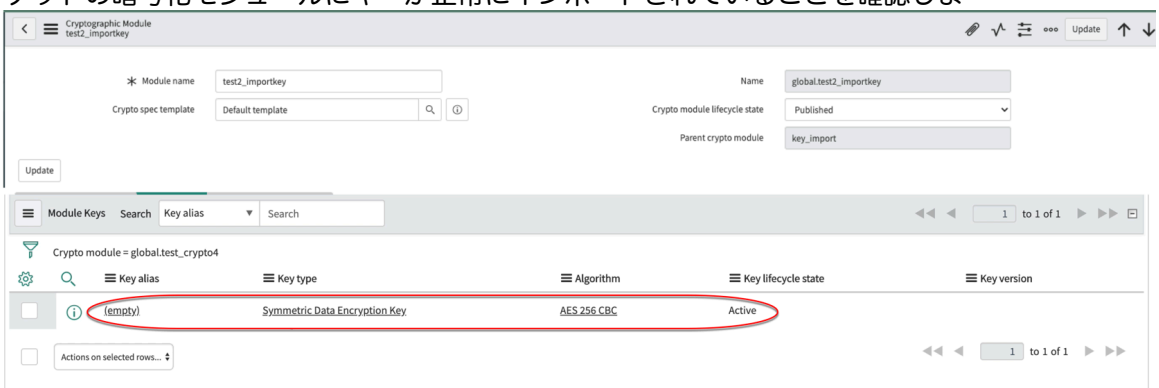
4. *HTTP POST* 要求を実行して、Web サービス REST エンドポイントからインポートします。

オプション	値/形式
エンドポイントの URL	*https://<instance>/api/sn_kmf/key/import?cryptoSpecSysID=<sys_id_of_crypto_spec>*

オプション	値/形式
CryptoSpecSysID パラメーター	新しく作成された暗号化仕様の *sys_id* 。 ヒント: 暗号化仕様のヘッダーを右クリックして、*sys_id* をコピーします。
ヘッダーのコンテンツタイプ	アプリケーション/オクテットストリーム
本文	インポートするファイル添付バイナリと公開鍵 (wrapped_symmetric_key.txt) を含める必要があります。
Web サービス REST エンドポイントからのインポート	*<username/password>* のベーシック認証を使用します。 注: 指定されたユーザーが sn_kmf_cryptographic_operator ロールを持っていることを確認します。

公開鍵のインポートに成功すると、ステータス *200* と書かれた HTTP 応答メッセージが返されます。

5. ターゲットの暗号化モジュールにキーが正常にインポートされていることを確認します。



キー管理フレームワークの健全性

キー管理フレームワーク のオンデマンド健全性ステータス情報にアクセスします。警告と誤動作エラーには、詳細なメッセージが含まれています。

始める前に

必要なロール：sn_kmf_cryptographic auditor または sn_kmf_admin または sn_kmf_cryptographic_manager

このタスクについて

キー管理フレームワーク の各コンポーネントの概要が示され、次のステータスと色が示されます。

- 緑/操作可能：コンポーネントは操作可能で、報告するエラーはありません。
- グレー/無効：コンポーネントが非アクティブであるため、健全性チェックは実行されません。
- 黄/デグレード：警告。コンポーネントは動作していますが、遅延/一時的な問題が発生する可能性があります。
- 赤/故障：致命的なエラーによりコンポーネントが動作できず、部分的な機能停止が発生する可能性があります。

コンポーネントには、個々のレポートが含まれるサブコンポーネントを含めることができ、その健全性ステータスは次のように親に影響を与えます。

- すべてのサブコンポーネントが非アクティブである場合、親は非アクティブとして表示されます。非アクティブなサブコンポーネントは、親の健全性に影響を与えません。
- 1 つまたは複数のサブコンポーネントがデグレードまたは誤動作している場合、親の健全性はデグレードとして表示されます。
- すべてのサブコンポーネントで誤動作が報告される場合、親でも誤動作が報告されます。

サブコンポーネントの追加情報については、「キー管理フレームワークのインスタンスレベルのキー」を参照してください。

注: 健全性チェックは 15 秒ごとに実行されます。[健全性] ページを更新してレポートを再実行します。

手順

1. 移動先 すべて > キー管理 > > 診断.
2. 次の健全性ステータス情報を確認します。

診断情報

カテゴリ	詳細
キーセキュア	暗号化が試行されているかどうかを確認します。
ファイルキーストア	インスタンスルートキー (IRK) のフェッチが試行されているかどうかを確認します。 **注:** ファイルキーストアは、オンプレミスインスタンスと開発者インスタンスで使用されるキーセキュアのオフラインの代替手段です。
GlideEncrypter	GlideEncrypter インスタンスレベルの暗号化モジュール、仕様、およびキーが存在するかどうかを確認します。 **注:** GlideEncrypter は、キー管理フレームワーク を介して Password2 フィールドの透過的な暗号化やその他の従来の暗号化の使用を可能にするスクリプト可能なコンポーネントです。
インスタンスキー暗号化キー (IKEK)	キーをファイルキーストアと KeySecure からフェッチできるかどうかを確認します。
インスタンス HMAC キー	キーをファイルキーストアと KeySecure からフェッチできるかどうかを確認します。
ボールト PKI	ボールト接続を確認して、インスタンス非対称暗号化キー (IAEK) とインスタンス署名キー (ISK) が使用可能で、ボールトから取得できるかどうかを確認します。
EJBCA PKI	LDAP 接続を確認して、IAEK と ISK が使用可能で、キャッシュと LDAP から取得できるかどうかを確認します。
インスタンス PKI	ファイルキーストアと KeySecure でキーをフェッチできるかどうか、および証明書が存在し、対称キーと一致するかどうかを確認します。

カテゴリ	詳細
	注: インスタンス PKI は、ServiceNow データセンター内のインスタンスでのみ使用できます。

トラブルシューティングのサポートについては、カスタマーサービス & サポートにお問い合わせください。

GlideEncrypter の廃止に向けたインスタンスの準備

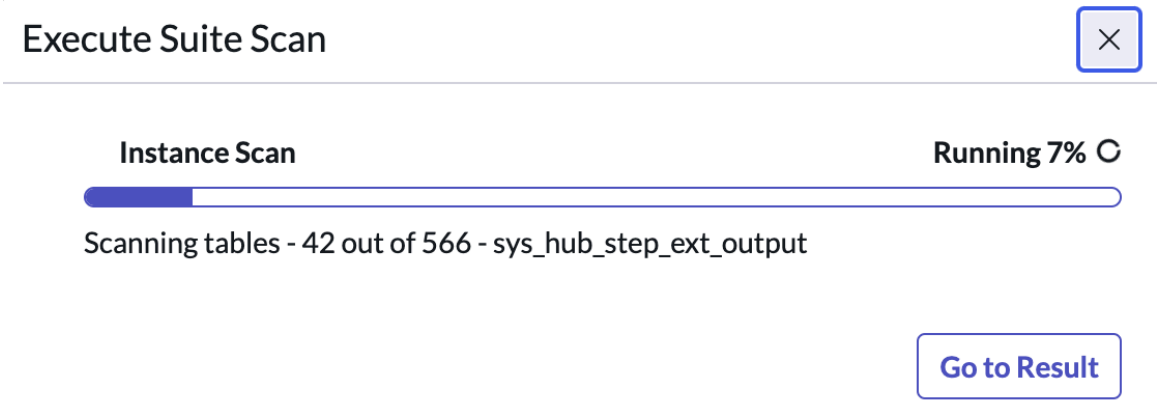
インスタンススキャンスクリプトを使用して、インスタンスで GlideEncrypter API 呼び出しを検索して削除します。これらの呼び出しの削除は、インスタンスで 3DES 暗号化を廃止するために必要なステップです。

始める前に
必要なロール：admin

GlideEncrypter API は、ServiceNow の Zurich リリースで廃止される予定です。スクリプトから GlideEncrypter 呼び出しを削除するのも、インスタンスで 3DES 暗号化を廃止する前の必要なステップです。

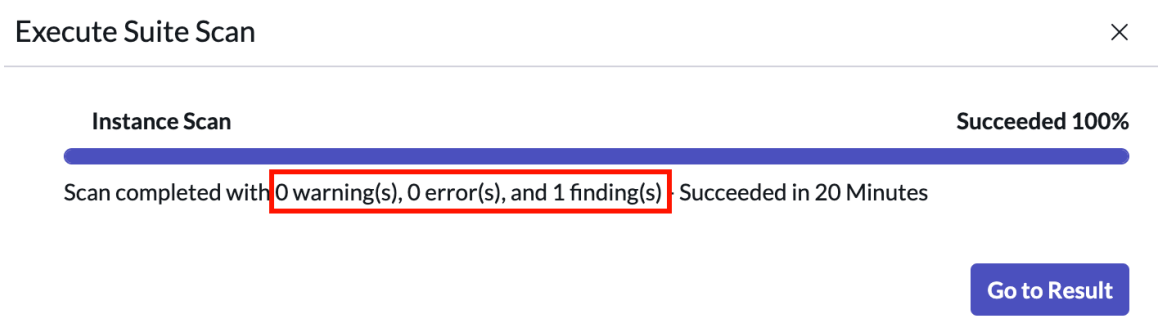
手順

1. 移動先 すべて > インスタンススキャン > スイート.
2. [スイート] リストで **[GlideEncrypter]** を選択して、GlideEncrypter が使用されているレコードを特定します。
3. [スイート] レコードで、[スイートスキャンの実行] を選択します。
4. [今すぐスイートをスキャン **(Scan Suites Now**)] ウィンドウで、[フルインスタンス **(Full Instance)**] を選択し、[スキャンの実行 **(Execute Scan)]** を選択します。
スイートスキャンが実行されます。スキャンの実行中に [テストスキャンを実行] すると、ウィンドウにスキャンの進捗状況が表示されます。

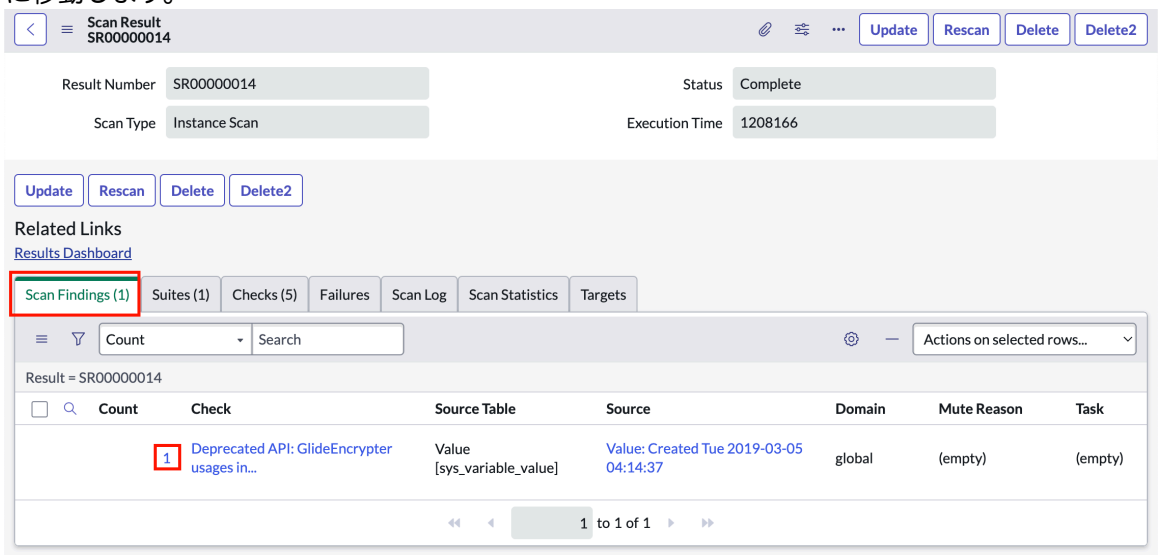


   注: このスキャンでは、顧客によって作成または変更されたレコードのみがチェックされます。

5. スキャンが完了したら、[結果に移動] を選択して [スキャン結果] レコードを表示します。



6. [スキャン結果] レコードで、[スキャン結果] リストの [カウント] フィールドを選択してレコードに移動します。



7. **GlideEncrypter** API を使用するレコード内の任意のスクリプトを変更します。**GlideEncrypter** の代替手段の詳細については、「Alternatives to deprecated GlideEncrypter APIs (廃止された GlideEncrypter API の代替手段)」を参照してください。
8. スクリプトから **GlideEncrypter** 呼び出しを削除した後、スキャンを再度実行して、API への呼び出しが残っていないことを確認します。

次のタスク
password2 フィールドの 3DES の GlideEncrypter の使用を廃止する

GlideEncrypter の廃止

インスタンスのスクリプトから従来の GlideEncrypter 呼び出しの使用を削除する方法について説明します。

Zurich リリースでの GlideEncrypter の可用性

Zurich リリースで GlideEncrypter を使用できるかどうかは、インスタンスが以前のリリースで作成されたか、以前のリリースからアップグレードされたかによって異なります。

新しいインスタンス

Zurich 以降に作成された新しいインスタンスでは、この API はデフォルトでオフになっています。すべてのベースシステムスクリプトが更新され、この API の呼び出しは使用されなくなりました。

アップグレードされたインスタンス

Zurich 以前のリリースからアップグレードされたインスタンスでは、従来の GlideEncypter API を引き続き使用できますが、API は キー管理フレームワーク を介して AES256-GCM 暗号化を使用するように更新されています。この変更により、NIST ガイドラインに準拠するために従来の 3DES 暗号化標準の使用が置き換えられ、GlideEncrypter を引き続き使用しているすべてのスクリプトが動作できるようになります。

GlideEncrypter を有効または無効にする

GlideEncrypter の可用性は、 *glide.security.glideencrypter.allow* システムプロパティによって制御されます。このシステムプロパティは、新しいZurichインスタンスではデフォルトで **false** に設定されており、更新することはできません。Zurich にアップグレードされたインスタンスでは、security_admin ロールを持つアドミニストレーターがこのプロパティを変更できます。

glide.security.glideencrypter.allow システムプロパティ

値	動作
true	true の場合、GlideEncypter は引き続きスクリプトで呼び出すことができますが、 キー管理フレームワークを介して AES256-GCM 暗号化を使用します。
false	false の場合、GlideEncrypter 呼び出しは null を返し、アドミニストレーターには次のエラーが表示されます。 Unsupported call to GlideEncrypter. Details: GlideEncrypter is deprecated and now returns null, please refer KB1320986

password2 フィールドの 3DES の GlideEncrypter の使用を廃止する

インスタンスで 3DES 暗号化標準の GlideEncrypter の使用を廃止すると、インスタンスが Password2 データの暗号化と復号化に対して、より安全な Advanced Encryption Standard (AES) を排他的に使用するようになります。

Rome 以降、password2 データは、より新しい Advanced Encryption Standard (AES) アルゴリズムを使用する キー管理フレームワーク を使用して保護されます。ただし、password2 ロジックの一部の構成と代替では、暗号化と復号化に 3DES アルゴリズムを引き続き使用できます。

Vancouver リリースでは、アドミニストレーターは 3DES アルゴリズムを完全に廃止することを選択できます。この変更を完了すると、インスタンスは password2 データに関連するすべての暗号化タスクと復号化タスクに対して AES 暗号化を排他的に使用します。この変更では 3DES 暗号化よりも優れたインスタンスセキュリティが提供され、これは、NIST 準拠を維持するために必要です。

廃止前の考慮事項

インスタンス間での password2 データの転送

password2 暗号化テキストを他のインスタンスに転送する場合は、ソースインスタンスとターゲットインスタンスの間で KMF 鍵交換 が有効になっていることを確認する必要があります。この構成により、password2 テキストの暗号化に使用されるキーが、password2 暗号化テキストを復号化するために両方のインスタンスで使用できるようになります。3DES を廃止する前に、インスタンス間の password2 データに影響を与える可能性がある次のユースケースを検討してください。

- password2 データを使用するアプリケーションがインスタンスにある場合は、そのインスタンスに KMF リソース交換 がインストールされていることを確認してください。KMF リソース交換 は、ソースインスタンスで password2 データの暗号化に使用されるインスタンスレベルのキーを、ターゲットインスタンスで復号化に使用で

自動翻訳

きるようにします。詳細については、「キー管理フレームワークリソース交換」を参照してください。

- XML またはデータソースを介して password2 データをエクスポートする場合は、ターゲットインスタンスで KMF 鍵交換 が有効になっていることを確認してください。この構成により、ソースインスタンスの password2 データの暗号化に使用されるインスタンスレベルのキーが、ターゲットインスタンスで復号化に使用できるようになります。この構成の詳細については、「キー管理フレームワークキー交換」を参照してください。

> **重要:** 上記の例はより一般的なシナリオですが、他の方法を使用して password2 暗号化テキストをインスタンス間で転送する場合は、KMF リソース交換 を設定して、ターゲットインスタンスが password2 データを復号化できるようにする必要があります。

3DES 廃止後のインスタンスのダウングレード

以下は、password2 フィールドの入力長が 125 文字を超え、3DES 暗号化が既に廃止されているインスタンスにのみ適用されます。

インスタンスのクローン作成を介して Vancouver より前のリリースにインスタンスをダウングレードするには、クローンを開始する前に次のステップを実行します。

1. データ保持が password2 フィールドデータを保持するように設定されているかどうかを確認します。
2. 設定されている場合は、クローンを要求する前に ServiceNow サポートに連絡して 3DES の廃止を無効にしてください。[理由] フィールドで、[password2 サポートに対するクローンダウングレードの前提条件 (Clone downgrade pre-requisite for password2 support)] を使用します。

従来の Password2 フィールド

インスタンスは 3DES 暗号化を使用して password2 データを従来の (Rome 前の) password2 データに変換します。3DES 暗号化の廃止後、このオプションは使用できなくなります。この機能が引き続き必要な場合は、部分的な廃止を要求します (詳細については次のセクションを参照してください)。

3DES を廃止する方法

上記のユースケースを確認した後、ナレッジベース記事のKB1704481を使用して、インスタンスで DES またはトリプル DES アルゴリズムの使用を安全に廃止するためのステップバイステップのプロセスを実行します。詳細については、「 KB1704481 」を参照してください。

> **重要:** セキュリティコンプライアンスモジュールを参照して次の手順を実行するには、security admin に昇格する必要があります。このプロセスの詳細については、「特権ロールへの昇格」を参照してください。

GlideEncrypter の廃止後

廃止プロセスが完了すると、次の情報がインスタンスに適用されます。

- password2 フィールドでは 3DES 暗号化データの復号化が引き続きサポートされます (ただし、暗号化はサポートされません)。
- password2 フィールドの既存の 3DES 暗号化データは、ユーザーまたはワークフローによってフィールド値が更新されるまで変更されません。
- password2 フィールドの値を更新すると 3DES 暗号化テキストが削除され、AES を使用して KMF で暗号化されたテキストに置き換えられます。

- 場合によっては、パスワードデータの保存時にインスタンスでエラーが表示されます。

  アクションが中止されました：技術的な問題により、パスワード値を保存できません。サポートについては、KB1296997 を参照してください。

  このエラーが表示された場合は、ナレッジベース記事 KB1296997 のサポート情報を参照してください。

キー管理フレームワークリソース交換

ServiceNow® リソース交換 は、安全な方法でインスタンス間でリソースを交換する機能を提供する KMF の機能です。

用語

リソース交換を使用する場合は、次の用語を参照してください。

リソース交換 用語

名前	説明
リソース交換	インスタンス間でリソースを交換するプロセス。
鍵交換 (KE)	インスタンス間でキーを交換するプロセス。
キーソースインスタンス (キーソース)	キーを所有するインスタンス。
キーターゲットインスタンス (キーターゲット)	キーを要求するインスタンス。

概要

リソース交換 は KMF の暗号化 API を使用して、機密性、完全性、認証、および否認防止を提供します。現在、リソース交換は鍵交換機能をサポートしています。詳細については、「キー管理フレームワークキー交換」を参照してください。

キー管理フレームワークキー交換

KMF 鍵交換 は KMF リソース交換 のサブセット関数です。鍵交換 は複数のインスタンス間で暗号化されたデータを安全に転送します。

鍵交換 の概要

鍵交換 はインスタンス間でキーを安全に転送します。

KMF 鍵交換 は、顧客がインスタンス間で KMF キーを交換するための安全な方法を提供します。アプリケーションのユースケースの 1 つにデータクローンプロセスがあります。鍵交換を使用すると、KMF コンポーネントのデータクローン中に暗号化モジュールキーがコピーされます。暗号化モジュール、モジュールキー仕様、およびモジュールアクセスポリシーはクローンプロセスに含まれます。キーの転送は含まれません。

この機能は、ServiceNow AI Platform Encryption サブスクリプションバンドルに含まれるキー管理フレームワークに含まれています。この製品の詳細については、 キー管理フレームワークを参照してください。

鍵交換 の使用

フィールド暗号化にKMFを使用するアドミニストレーターは、データのクローン作成を実行するときに、鍵交換を使用して本番インスタンス間でキーをクローンできます。データのクローン作成では、アドミニストレーター/ KMF 暗号化マネージャーは次のことを実行できます。

- すべてのキーを他のインスタンスに交換する。
- 特定のキーを 1 回または定期的に他のインスタンスと交換する。
- ターゲットインスタンスからキーソースインスタンスにオンデマンド要求を送信する。
- 暗号テキストをリキーするためにソースからターゲットにキーを交換する。
  - 要求の有効期限を管理し、要求が期限切れになった場合は、キーを削除するか、鍵交換要求を却下します。
  - 要求が完了してキーがインポートされると、使用したキーは期限切れになり、タイムスタンプが付与されます。
  - ソースのキーで暗号化されたターゲットインスタンスで暗号テキストをリキーします。

サポートされているモード

鍵交換は、暗号化モジュールの暗号化仕様レベルでいくつかのモードをサポートしています。

モード	説明
自動 (設定なし、デフォルトの動作)	すべてのキーは、追加の設定なしに、データクローンプロセス中に自動的に送信されます。
設定可能 (1 回限りの設定セットアップ)	アドミニストレーターは、データクローンプロセス中に送信されるキーを設定します。
手動 (ループ内のユーザー)	アドミニストレーターは、ターゲットインスタンスのオンデマンド要求をソースに送信します。キーソースインスタンスでアドミニストレーターが要求を承認する必要があります。
リキー (自動要求)	アドミニストレーターは、クローンセットアッププロセス中にリキーのオプションを選択します。

鍵交換の設定

キー管理フレームワーク (KMF) は、インスタンスの新規インストールまたはアップグレード中に、サポートされている暗号化モジュールの自動鍵交換要求を生成します。KMF はインスタンスのデータ暗号化キーをローカルに管理します。

始める前に

キーを持つ KMF 暗号化モジュールは、鍵交換 を使用する前にターゲットとソースの両方のインスタンスで作成する必要があります。

必要なロール：sn_kmf.cryptographic_manager

このタスクについて

鍵交換要求はターゲットインスタンスから開始されます。

自動鍵交換は、プロパティがターゲットインスタンスにクローンされるインスタンスをクローンするときにデフォルトで有効になります。KMF とともに、システムプロパティを設定して、インスタンスのクローン中のキーの処理方法を管理します。

- 自動鍵交換のオフ：繰り返しクローン要求の *glide_encryption.auto_key_exchange.enabled* プロパティを **false** に設定します。
- 自動鍵交換要求の送信：このプロパティを **true** に設定します。

重要: ベースシステムプロパティはデフォルトで **true** に設定されています。これは、インスタンスのクローン作成時に自動鍵交換がアクティブ化されることを意味します。鍵交換によるリキー暗号テキスト または繰り返し鍵交換機能を利用する場合は、この値を **false** に設定する必要があります。詳細については「繰り返し鍵交換のチュートリアル」を参照してください。

手順

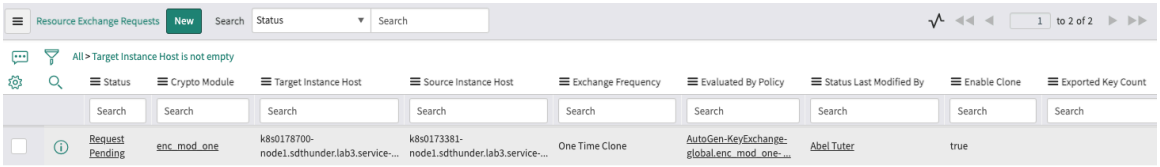
1. 移動先 すべて > キー管理 > リソース交換要求 > 新規.
2. フォームで、フィールドに入力します。

リソース交換要求フォームのフィールド

名前	説明
交換頻度	○ アドホック：キーターゲットインスタンスからソースインスタンスへ要求を送信します。ソースのインスタンスの sys_id とホスト情報を入力します。鍵交換のリキーではサポートされていません。 ○ ワンタイムクローン：ソース暗号化仕様からターゲットインスタンスへのキーの 1 回限りの交換。 ○ 繰り返しクローン：定義された繰り返しクローンで、選択したソース暗号化仕様からターゲットインスタンスにキーを交換します。
<Source or Target> インスタンス sys_id	○ アドホック：キーを要求するソースインスタンスの sys_id を入力します。 ○ **1** 回限りのクローン、繰り返しクローン：要求を送信するターゲットインスタンスの sys_id を入力します。 **ヒント:** アプリケーションナビゲーターに「stats.do」と入力してインスタンス ID を見つけます。
<Source or Target> インスタンスホスト	ソースまたはターゲットインスタンスのホストの場所または名前を入力します。 **ヒント:** 例：instanceA.service-now.com
暗号化仕様	暗号化モジュールの暗号化仕様のキーによって、クローンするキーが定義されます。1 回限りのクローン要求と繰り返しのクローン要求の両方で、インスタンスは自動的にリソース交換モジュールのアクセスポリシーを作成します。ポリシーを手動で設定する必要はありません。 **注:** [リストから参照] アイコン (🔍) を選択して使用可能な暗号化仕様を参照します。
キーのインポート後にリキーを有効にする	自動リキーを有効にするオプション。

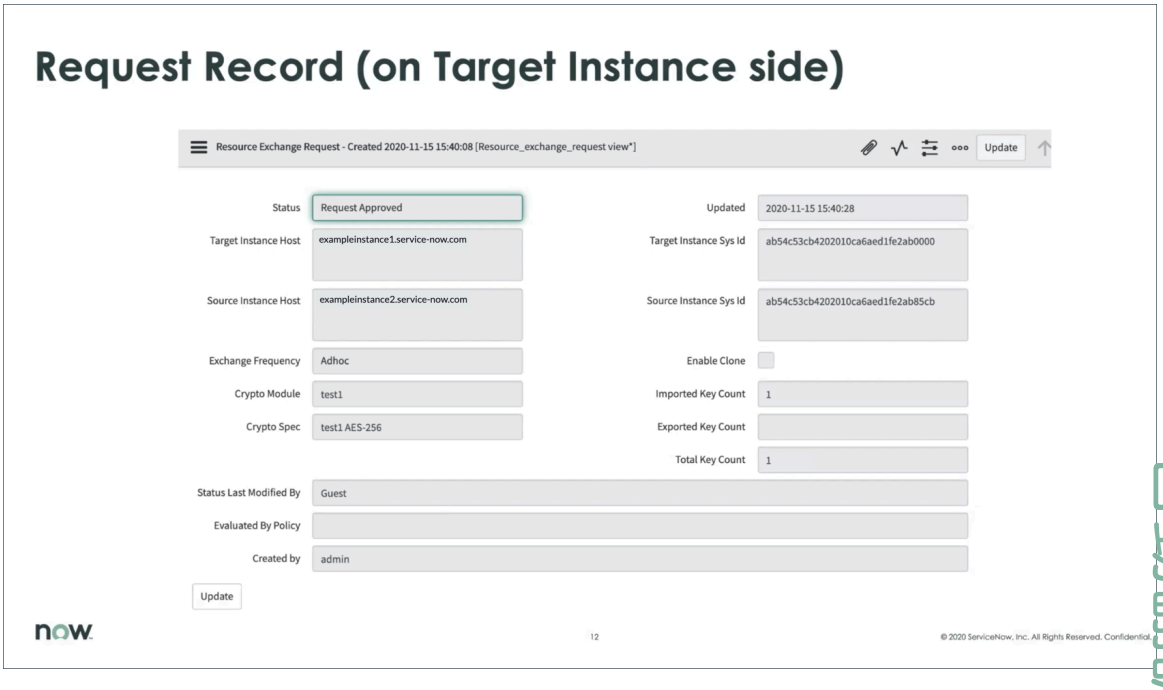
3. [送信] を選択します。
成功すると、フォームの上部に確認が表示されます。要求テーブルは、ソースインスタンスとターゲットインスタンスの両方において、処理待ちの要求のエントリで更新されます。要求レコードを開くと、要求のステータス、インポートされたキー数、およびターゲットまたはソースホストの合計キー数が表示されます。

自動翻訳



4. 処理待ちの要求がソースインスタンスで受け入れられ、交換が完了します。

   クローン時に、ソースインスタンスのモジュールアクセスポリシーが呼び出され、要求が自動的に承認されて、新しくクローンされたターゲットにキーが送信されます。



結果
鍵交換が試行された後、非本番インスタンスで *protected.script.values.kmf.rekeyed* システムプロパティが更新されます。このプロパティは、鍵交換の試行後にシステムプロパティ [sys_properties] テーブルに表示されます。交換されたキーを使用した暗号化が成功した場合、このプロパティの値は **true** です。それ以外の場合、プロパティの値は **false** です。値が false の場合、インスタンスは翌日に暗号化を再試行します。

鍵交換によるリキー暗号テキスト

リソース交換 は、ソースのキーで暗号化されたターゲットインスタンスで暗号テキストのリキーをサポートします。リキーアクティビティはキーライフサイクルで追跡されます。

概要

KMF for フィールド暗号化 を使用するアドミニストレーターは、データのクローン作成を実行するときに、鍵交換を使用して本番インスタンス間で暗号キーをリキーできます。リキーにはアクティブなキーが必要であるため、リキーを行うにはまずアクティブなキーをターゲットインスタンスで使用できるようにする必要があります。暗号化ジョブが自動的に作成されて実行されると、ソースキーのローテーションとリキーが行われ、暗号テキストが再暗号化されます。

鍵交換を使用して、次の操作を行います。

- リキーの有効期限を設定します。

  要求の有効期限が切れている場合、要求は却下され、キーは削除されます。

- ソースインスタンスのキーで暗号化された暗号テキストを自動的にリキーします。

  新しいクローンされた暗号化キーを使用して、ターゲットインスタンスの暗号テキストを再暗号化します。

- リキーの目的はクローンプロセス中に設定され、クローンの一部として自動化されます。
- リキーのアクティビティは、暗号化モジュールの [モジュールキー] タブで追跡されます。キーアクティビティのキーライフサイクル状況とキーバージョンにアクセスします。詳細については、「キーのローテーション」を参照してください。

鍵交換を設定し、[キーのインポート後にリキーを有効にする] チェックボックスを選択してアクティブ化します。詳細については、「鍵交換の設定」を参照してください。

繰り返し鍵交換のチュートリアル

このチュートリアルでは、リソース交換 を使用してインスタンスで繰り返し鍵交換を設定します。

始める前に

必要なロール：sn_kmf.cryptographic_manager

このタスクについて

この例は、ターゲットインスタンスがホストインスタンスからキーを要求する方法を示しています。

- この手順を実行する前に、インスタンスをクローンする必要があります。詳細については、「システムクローン」を参照してください。
- 自動鍵交換：ベースシステムプロパティ *glide_encryption.auto_key_exchange.enabled* はデフォルトで **true** に設定されています。これは、インスタンスのクローン作成時に自動鍵交換が有効になることを意味します。プロパティがターゲットインスタンスにクローンされます。
- 自動鍵交換をオフにするには、プロパティを **false** に設定します。

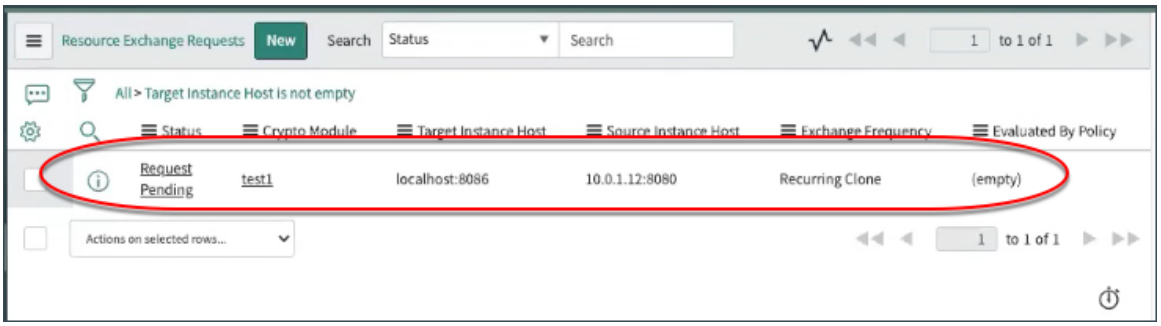
手順

1. ソースインスタンスで、column_level_encryption を使用して暗号化モジュールを作成するか、既存の暗号化モジュールにアクセスして、鍵交換の暗号テキスト暗号化の暗号化フィールド構成を設定します。
   詳細を参照してください。
   a. 暗号化モジュールでキーが生成されていることを確認します。

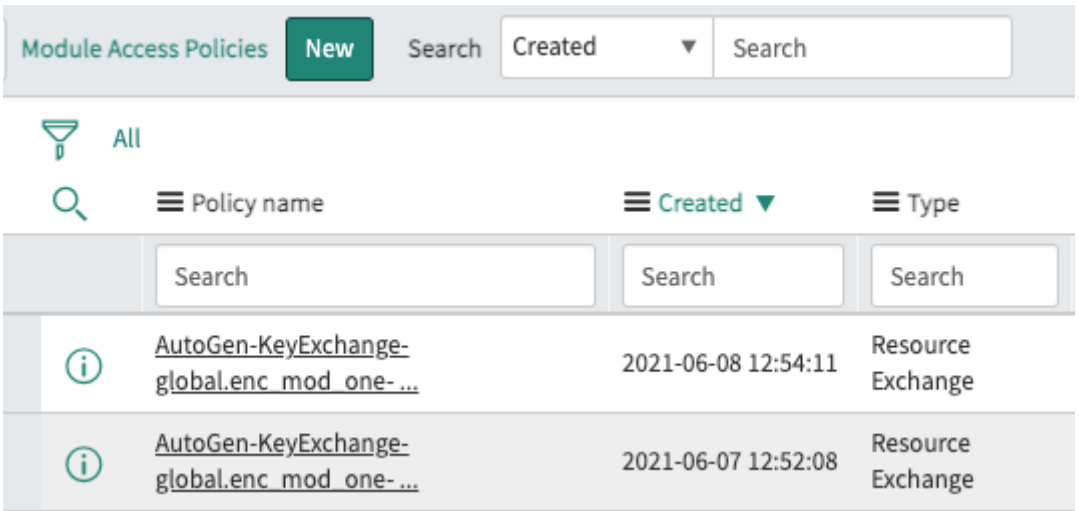
   注: インスタンスはクローン要求の実行時にモジュールアクセスポリシーを自動的に作成します。

2. クローンされたインスタンスから、 キー管理 > リソース交換要求 > 新規.
3. フォームに入力し、 交換頻度として [繰り返しクローン] を選択します。
4. クローンのターゲットインスタンスから、 キー管理 > リソース交換要求.
   ホストインスタンスからの要求がテーブルに表示されます。

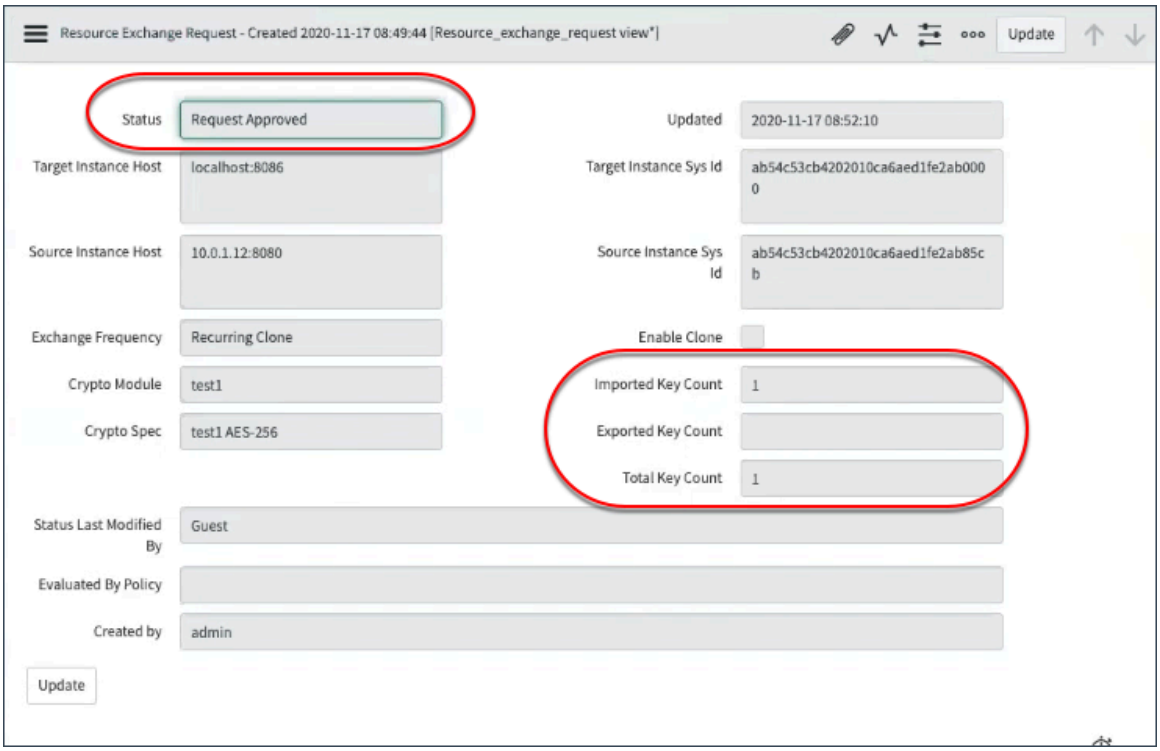
自動翻訳



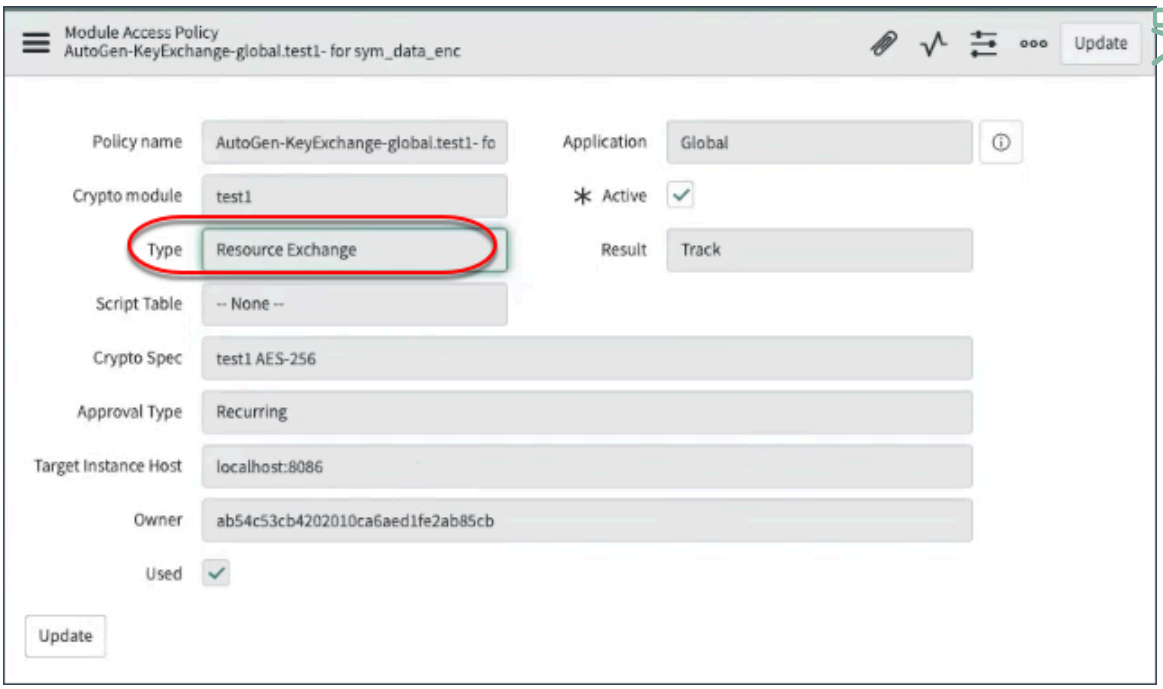
重要: 1 回限りのクローン要求と繰り返しのクローン要求の両方で、インスタンスは自動的にモジュールのアクセスポリシーを作成します。ポリシーを手動で設定する必要はありません。クローン時に、ソースインスタンスのこのポリシーが呼び出され、要求が自動的に承認され、新しくクローンされたターゲットにキーが送信されます。



要求フォームで、ステータスが [承認済み要求] に更新され、[インポートされたキーの数] フィールドがレコードに表示されます。



5. ホストインスタンスに戻ります。
6. 要求レコードを表示して、エクスポートされたキーの数を確認します。
7. モジュールアクセスポリシーレコードが表示され、[タイプ] が [リソース交換] であることを確認します。



注: リソース交換は、ターゲットインスタンスの暗号テキストのリキーもサポートしています。詳細については、「」を参照してください。

結果
鍵交換が試行された後、非本番インスタンスで *protected.script.values.kmf.rekeyed* システムプロパティが更新されます。このプロパティは、システムのプロパティ [sys_properties] テーブルに表示されます。交換されたキーを使用した暗号化が成功した場合、このプロパティの値は **true** です。それ以外の場合、プロパティの値は **false** です。値が false の場合、インスタンスは翌日に暗号化を再試行します。

Infrastructure Security

Infrastructure Security のツールを使用して、クライアントからサーバーへのトラフィックを暗号化するためにインスタンスが使用する証明書を作成、アップロード、および管理します。

Infrastructure Security プラグインには、Transport Layer Security (TLS) の暗号と証明書の管理に使用できるツールが入っています。インスタンスは TLS を使用して、クライアントからサーバーへのトラフィックを暗号化します。

インスタンスで使用する暗号を選択する

アドミニストレーターは、**[TLS]** ページを使用して、インスタンスで使用するデータセンターの暗号を構成したり、暗号を試行する順序を選択したりできます。

独自の証明書を生成してアップロードする

Infrastructure Security のツールを使用して、選択した認証局で署名可能な独自の証明書署名要求を生成します。その後、ツールを使用して署名済み証明書をインスタンスのロードバランサーにアップロードできます。

暗号と証明書のステータスを監視する

[TLS 設定] および **[SYOC** 設定] ページを使用して、暗号と証明書に加えた変更のステータスを表示します。

Infrastructure Security プラグインのインストール

これらの機能の使用を開始するための ServiceNow の Infrastructure Security Settings (com.glide.infrastructure_security) プラグインをインストールします。プラグインのアクティブ化の詳細については、「プラグインをアクティブ化する」を参照してください。

プラグインをインストールした後は、*sn_infra_sec.syoc.enabled* のシステムプロパティを true に設定して、Sign Your Own Security (SYOC) 機能を有効にします。

注: *sn_infra_sec.syoc.enabled* プロパティがインスタンスで利用できない場合は、作成する必要があります。このプロセスの詳細については、「システムプロパティを追加する」を参照してください。

証明書署名要求の生成

[証明書署名の生成 (CSR) (Generate Certificate Signing (CSR))] ページを使用して、インスタンスのロードバランサーに対する顧客の署名済みの証明書に対応する証明書署名要求を作成します。

始める前に
必要なロール：admin

これらの手順を実行するには、Infrastructure Security (com.glide.infrastructure_security) プラグインがインストールされている必要があります。このプラグインの詳細については、「Infrastructure Security」を参照してください。

ServiceNow でカスタム URL を使用する方法については、「カスタム URL をインスタンス URL として設定」を参照してください。

手順

1. 移動先 すべて > インフラストラクチャのセキュリティ設定 > **CSR** を生成.
2. 1 つ以上のドメインを要求に追加します。

   a. [ドメイン] 見出しの下にある [追加] ボタンを選択します。

   b. ポップアップウィンドウで、ドメインを入力して **[OK]** を選択します。

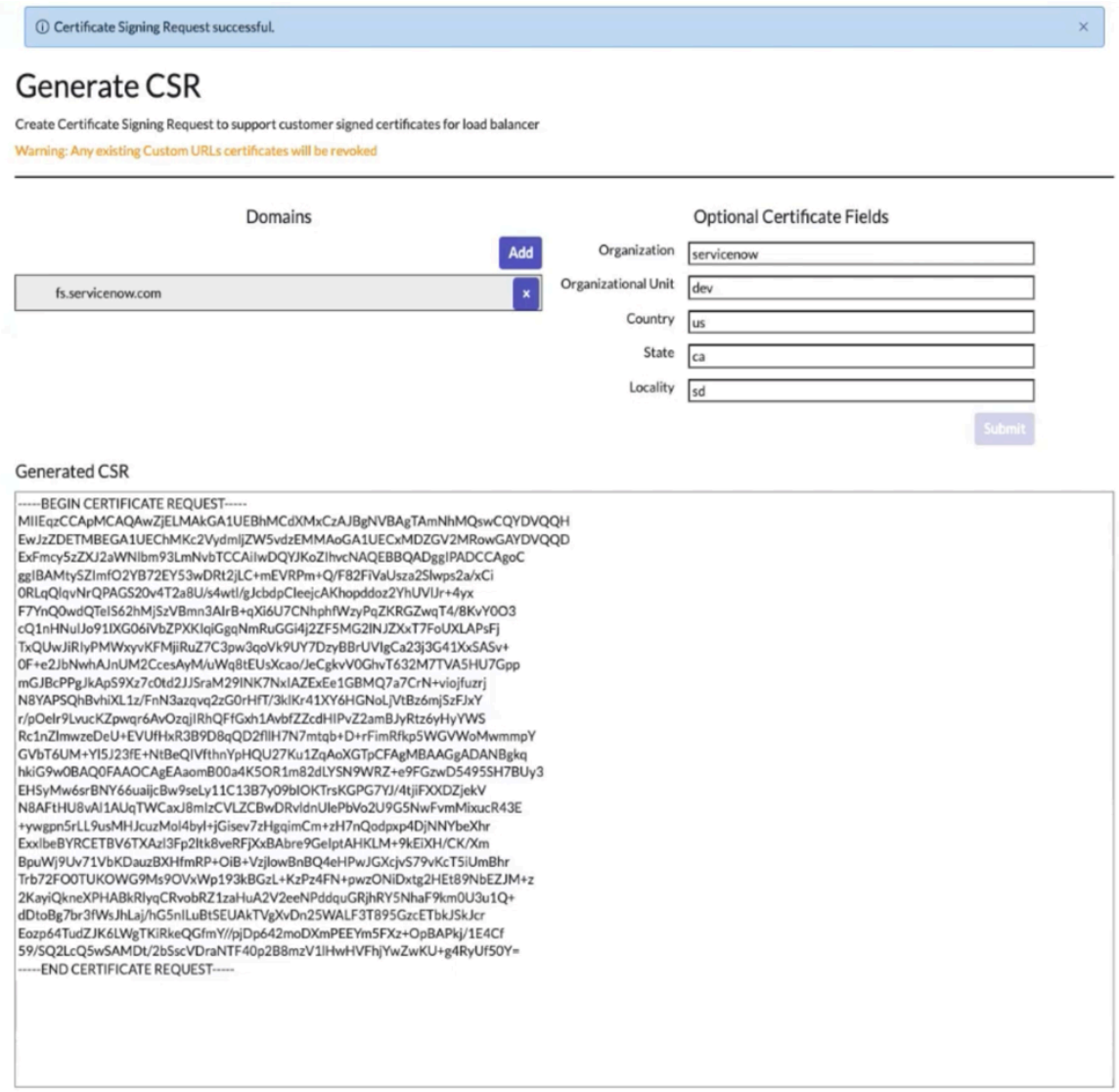
   c. 必要に応じて、上記の手順を繰り返してさらにドメインを追加します。

   注: ドメインは、各ドメインエントリの左側にある [X] ボタンを選択して削除できます。

3. [オプションの証明書フィールド **(Optional Certificate Fields)]**に、要求に含める情報を入力します。
4. [送信] を選択します。

   警告: 別の要求の生成中に要求を送信することはできません。この問題が発生すると、「リソースが競合しています (Resource in Conflict)」というエラーが表示されます。処理を続行するには、現在の要求をキャンセルするか、現在の要求が処理されるのを待ってから別の要求を送信します。

   [送信 **(Submit)]** を選択すると、インスタンスで証明書署名要求が生成されます。要求が [生成された **CSR (Generated CSR)]** フィールドに表示されます。



5. [生成された **CSR (Generated CSR)]** フィールドの内容をコピーします。

Key Management Framework (KMF) による Password2 暗号化

キー管理フレームワークでサポートされており、Password2 (双方向暗号化) フィールドタイプを使用して、職務の分離、キー保護、およびライフサイクル管理でカスタムフィールドを暗号化および復号化します。この暗号化と復号化は NIST 800-57 ガイドラインに従って行われ、FIPS 140-2-L3 保護を提供します。

Password2 は、双方向暗号化でパスワードを格納するテキストフィールドです。双方向暗号化では、インスタンス内で復号化できるセキュアな暗号化された値としてデータを格納します。

ヒント: Vancouver リリース以降、アドミニストレーターは password2 フィールドの 3DES 暗号化を廃止して、新しい Advanced Encryption Standard (AES) を導入できます。詳細については、「password2 フィールドの 3DES の GlideEncrypter の使用を廃止する」を参照してください。

アクティベーション

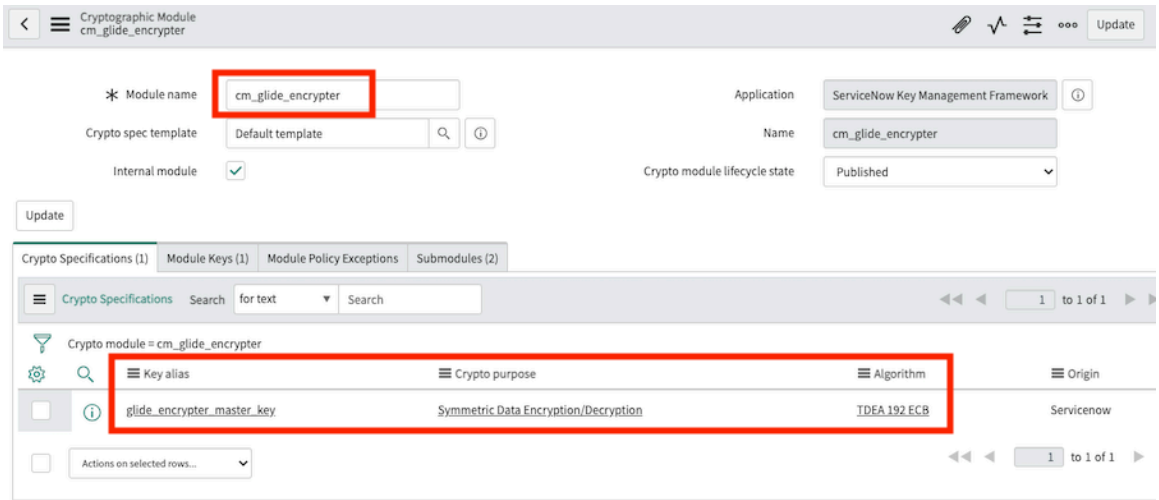
Password2 機能はデフォルトでアクティブにされています。この機能は *glide.kmf.encrypter.enabled* プロパティで制御され、すべての新しいインスタンスとアップ

グレードに対して **true** に設定されます。Password2 を使用するために フィールド暗号化エンタープライズ を有効にする必要はありません。

Password2 の仕組み

キー管理フレームワークでは、基本システムの親暗号化モジュール **cm_glide_encrypter** が提供されます。このモジュールは、暗号化仕様と が従来の Password2 フィールドを復号化できるキーを提供します。

Password2 の暗号化モジュール



この cm_glide_encrypter モジュールにはサブモジュールを含めることができ、それぞれに独自のモジュールキーと仕様があります。Password2 フィールドがあるアプリケーションと同じアプリケーションスコープにサブモジュールが存在する場合、そのサブモジュールが使用されます。たとえば、ServiceNow　カスタマーサービス アプリケーションのテーブルにサブモジュールがあり、カスタマーサービス アプリケーションスコープのテーブルの Password2 フィールドに情報を書き込むと、暗号化プロセスは カスタマーサービス サブモジュールを呼び出します。プロセスでは、一意の AES 256 GCM 暗号化キーによる暗号化と復号化にもそのサブモジュールのキーが使用されます。アプリケーションスコープごとに 1 つのサブモジュールが許可されます。親モジュールは常にグローバルスコープに使用されるわけではありません。通常、新しいフィールドでは instance_level_glide_encrypter が使用されます。

注: Zurich で独自のサブモジュールを作成することはできません。サブモジュールは、ServiceNow AI Platform のさまざまなアプリケーションプラグインで提供されます。サブモジュールのキーはローテーションできますが、親の cm_glide_encrypter モジュールのキーはローテーションできません。

ドメインセパレーションとオンプレミスの顧客

KMF Password2 ではドメインセパレーションはサポートされていません。Password2 はオンプレミスインスタンスで使用できます。

従来の **Password2** と現在の **Password2**

Zurichで、既存の Password2 フィールドがアップグレードされました。

Password2 の現在の実装：

- NIST 800-57 キーラッピングガイドラインに従って キー管理フレームワーク を使用し、キー階層全体に FIPS 140-2-L3 保護を提供します。
- 特定のアプリケーション専用の一意の KMF Password2 サブモジュールを作成し、アプリケーションスコープ全体を制御する機能が含まれています。各サブモジュールには、独自の AES 256 GCM 暗号化キーがあります。

スクリプトの **Password2** フィールド

スクリプトを使用して Password2 フィールドにアクセスする場合は、テーブルスコープと同じスコープでスクリプトを実行します。`setDisplayValue()` を使用して Password2 値を暗号化し、`getDecryptedValue()` を使用して値を復号化して読み取ります。

注: Password2 フィールドで `GlideEncrypter()` API を使用しないでください。

このサンプルスクリプトは、「table_xyz」テーブルの password2 列の `my@Password` を暗号化する方法を示しています。

```
var  gr =  new GlideRecord('table_xyz');
gr.pwd2column_name.setDisplayValue('my@Password');

gr.insert();
```

重要: Password2 フィールドに `setValue()` API を使用することはできません。

このサンプルスクリプトは、同じフィールドを復号化して値を取得する方法を示しています。

```
var  gr =  new GlideRecord('table_xyz');
gr.query();
gr.next();
var ge=gr.getElement('pwd2column_name');
var ged1 = ge.getDecryptedValue();
```

重要: `getDecryptedValue()` API はスコープ対象ではありません。グローバルに利用可能です。

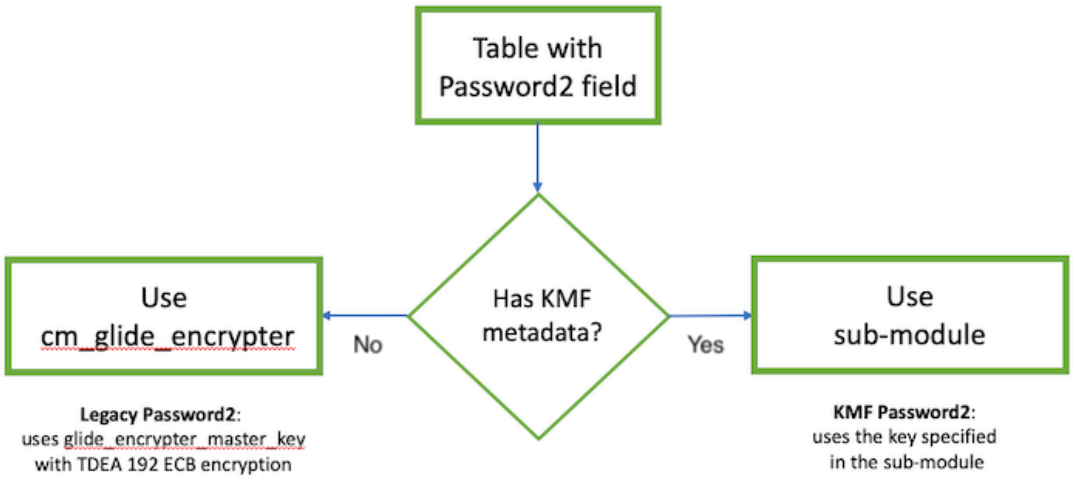
1. Password2 フィールドのデータを暗号化すると、Password2 フィールドが存在するアプリケーションのスコープが決定されます。
2. プロパティが *true* に設定されている場合、アプリケーションと同じスコープを持つ **cm_glide_encrypter** 親モジュールのサブモジュールが検索されます。

   注: 同じスコープのサブモジュールが存在する場合は、サブモジュール仕様とキーを使用して暗号化が実行されます。

この図は、インスタンスが Password2 フィールドのデータを復号化する方法を説明しています。

自動翻訳

Password2 復号化フロー



KMF Password2 移行ジョブ

以前のリリースからアップグレードする顧客向けに、移行ジョブが提供されています。従来の Password2 暗号化で暗号化されたデータを KMF Password2 サブモジュールキーで再暗号化します。再暗号化は、アプリケーションスコープ内に Password2 フィールドがあり、そのスコープ用に作成されたサブモジュールもあるテーブルにのみ適用されます。たとえば、**XYZ_example** アプリケーション (**XYZ_example** アプリケーションスコープを持つ) の従来の Password2 フィールドは、**XYZ_example** アプリケーションスコープのサブモジュールが cm_glide_encrypter 親モジュールの下に存在する場合にのみ再暗号化されます。

サブモジュール内の KMF Password2 暗号化キーは、KMF キー階層で保護されます (エンベロープ暗号化)。

証明書

インスタンスには、セキュア接続を確立して署名を検証するための証明書が必要です。

探索

証明書の主な機能とビジネス価値について学びます。

構成

コア構成を計画します。

アップロード

アプリケーションを計画して設定します。

証明書の概要

インスタンスには、セキュア接続を確立して署名を検証するための証明書が必要です。

証明書は次のような機能に使用されます。

- LDAPS
- 送信 Web サービス相互認証
- Web サービスセキュリティ
- MID サーバー

証明書を使用するには、セキュリティ保護されたサーバーまたはクライアントの証明書を生成または購入し、インスタンスにアップロードする必要があります。

LDAP 証明書

インスタンスが LDAP サーバーとの LDAP over SSL (LDAPS プロトコル) 接続を確立するには、SSL 証明書が必要です。

インスタンスは、次の 2 つのタイプの LDAP 証明書を受け入れます。

証明書	タイプ	必須
LDAP サーバー証明書	サポートされている任意のタイプ	すべての LDAP 構成
LDAP クライアント証明書	Java キーストアタイプ	相互認証

複数のサーバー証明書がある場合、LDAP サーバーが接続を許可するまで、インスタンスは各サーバー証明書を順番に試行します。複数の LDAP サーバーを使用する場合は、LDAP サーバーごとに SSL 証明書を含めてください。

相互認証では、サーバーに加えてクライアントが証明書を提示する必要があります。LDAP サーバーが相互認証を必要とする場合は、LDAP サーバーのクライアント証明書を Java キーストアタイプの証明書で指定する必要があります。

証明書の条件

有効な証明書は次の条件を満たす必要があります。

- 証明書のキーサイズは最大 2048 ビットです。
- 証明書には、次のいずれかのファイル拡張子が必要です。

拡張	説明
DER	*Distinguished Encoding Rules* 形式は、バイナリメッセージ転送構文です。この形式は、.CER および .CRT ファイル拡張子もサポートしています。
CER	Distinguished Encoding Rules 形式を使用する証明書の証明書ファイル拡張子。
CRT	Distinguished Encoding Rules 形式を使用する証明書の証明書ファイル拡張子。
PEM	*Privacy Enhanced Mail* 形式は、「-----BEGIN CERTIFICATE-----」と「-----END CERTIFICATE-----」のテキスト文字列で囲まれた、base-64 でエンコードされた DER 証明書です。

証明書の信頼

デフォルトでは、インスタンスは、Java 仮想マシン (JVM) で認識される認証局 (CA) からの証明書のみを信頼します。自己署名証明書およびエンタープライズ署名証明書は信頼されません。

注: 証明書の使用に影響するプロパティの詳細については、「インスタンスセキュリティ強化設定」の「」を参照してください。

LDAP クライアント証明書の生成

OpenSSL を使用して相互認証用の LDAP クライアント証明書を生成します。最終的な出力は、Java キーストア内に格納される PKCS#12 証明書です。

始める前に

必要なロール：admin

このタスクについて

証明書の生成の詳細については、OpenSSL のドキュメント を参照してください。これらの手順では、OpenSSL にアクセスできることを前提としています。

コマンドラインインターフェイスで次のコマンドを入力します。

手順

1. 自己署名クライアント証明書を生成します。

   Example
   たとえば、このコマンドは、test1-key.key 秘密キーに基づいてクライアント証明書 test1-cert.crt を作成します。

   ```
   openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout test1-key.key -out
    test1-cert.crt
   ```

2. 証明書ファイルと秘密キーの両方を PKCS#12 (.pfx または .p12 拡張子のファイル) に変換します。

   Example
   たとえば、このコマンドは、クライアント証明書と秘密キーを test1-certificate.pfx と呼ばれる PKCS#12 証明書に変換します。

   ```
   openssl pkcs12 -export -out test1-certificate.pfx -inkey test1-key.key -in test1-cert.crt
   ```

3. Java キーストアを生成し、pkcs12 ファイルをインポートします。

   Example
   たとえば、このコマンドは、証明書を test1.jks Java キーストアにインポートします。

   ```
   keytool -importkeystore -srckeystore test1-certificate.pfx -srcstoretype PKCS12
    -destkeystore test1.jks
   ```

4. Java キーストアファイル (test1.jks) の証明書をインスタンスにアップロードします。

   注:

   .jks 拡張子の証明書を使用してオンプレミスのインスタンスにアップロードしているときに、「アプリケーションのアップロードを処理するための有効な証明書が見つかりません」というメッセージが表示された場合は、.pfx 拡張子の証明書を代わりに使用してください。

次のタスク

インスタンスへの証明書のアップロード

サーバー証明書の生成

keytool を使用して、新しい Java キーストアファイルを生成し、証明書署名要求 (CSR) を作成してから、秘密キー、公開証明書のペア、および署名済み証明書をキーストアにインポートすることができます。

始める前に
必要なロール：admin

このタスクについて
キーと CSR の生成の詳細については、Java keytool のドキュメント を参照してください。

コマンドラインインターフェイスで次のコマンドを入力します。

手順

1. Java キーストアとキーペアを生成します。

   Example
   たとえば、このコマンドは my.keystore と呼ばれるキーストアを作成し、キーストア内に mydomain と呼ばれる秘密キーを生成します。

   ```
   keytool -genkey -alias mydomain -keyalg RSA -keystore my.keystore
   ```

2. 既存の Java キーストアの CSR を生成します。

   Example
   たとえば、このコマンドは mydomain.csr または mydomain キーと呼ばれる CSR を生成します。

   ```
   keytool -certreq -alias mydomain -keystore my.keystore -file mydomain.csr
   ```

3. ルートまたは中間認証局 (CA) 証明書を Java キーストアにインポートします。

   Example
   たとえば、このコマンドは Thawte の CA 証明書をインポートします。このコマンドでは、Thwate が CSR に署名した CA であると仮定しています。

   ```
   keytool -import -trustcacerts -alias root -file Thawte.crt -keystore my.keystore
   ```

4. 署名済みプライマリ証明書を Java キーストアにインポートします。

   Example
   たとえば、このコマンドは署名された証明書 mydomain.crt をキーストアにコピーします。

   ```
   keytool -import -trustcacerts -alias mydomain -file mydomain.crt -keystore my.keystore
   ```

5. Java キーストアファイル (my.keystore) の証明書をインスタンスにアップロードします。

次のタスク
インスタンスへの証明書のアップロード

インスタンスへの証明書のアップロード

[証明書] モジュールからインスタンスに証明書を追加します。

証明書をインスタンスにアップロードする

証明書をインスタンスにアップロードする。

始める前に
必要なロール：admin

このタスクについて
ADFS サーバーで証明書が更新されている場合は、更新された証明書をインスタンスにもアップロードする必要があります。

手順

1. 移動先 すべて > システム定義 > 証明書.
2. **[New** (新規)] を選択します。
3. フォームのフィールドに入力します。

フィールド	説明
名前	証明書の一意の名前を指定します。
有効期限通知	[オプション] 証明書の有効期限が近づいたときに通知を送信するかどうかを選択します。
アクティブ	インスタンスが安全な通信と署名要求にこの証明書を使用するかどうかを選択します。
簡単な説明	[オプション] 要求者やサーバー名などの証明書の説明を入力します。
フォーマット	証明書のフォーマットを選択します。インスタンスは、PEM および DER フォーマットをサポートしています。
タイプ	証明書コンテナを選択します。インスタンスは、トラストストア、Java キーストア、および PKCS#12 キーストアからの証明書を認識します。
PEM 証明書	DER 証明書を含む、base-64 でエンコードされた PEM 形式のテキストを入力します。インスタンスは証明書をデコードして、[有効開始日]、[有効期限]、[数日中に期限切れ]、[発行者]、および [件名] フィールドに入力します。

4. [送信] を選択します。

   アップロード中に、モジュールは証明書の読み取り専用プロパティを抽出して次のフィールドに表示します。
   - 有効開始日
   - 有効期限
   - Issuer
   - 証明書の件名

5. [ストア/証明書を検証] を選択して、証明書が正しいかどうかを確認します。
   インスタンスで証明書またはキーストアに関するエラーが発生した場合は、エラーメッセージが表示されます。

信頼できるサーバー証明書のアップロード

サービスプロバイダーの信頼できるサーバー証明書をアップロードすることで、インスタンスにより、接続しているサービスの有効性と安全性を確保できます。

始める前に
必要なロール：admin

このタスクについて
インスタンスは、サービスプロバイダーによって提供される証明書を使用して、送信 Web サービス呼び出しを検証します。

手順

1. 信頼ストア証明書タイプの新しい証明書レコードを作成します。
2. 次のいずれかのアクションを実行します。
   - サービスプロバイダーの DER 形式の証明書を添付します。
   - サービスプロバイダーの PEM 形式の証明書を **[PEM** 証明書] フィールドにコピーして貼り付けます。

フィールド暗号化

フィールド暗号化を使用して、インスタンス上の暗号化されたデータを権限のないユーザー、スクリプト、またはシステムプロセスから保護します。

フィールド暗号化 は、 ServiceNow キー管理フレームワークに基づく暗号化製品です。 フィールド暗号化 では、インスタンス内の特定のフィールドまたは添付ファイルを暗号化できます。フィールド暗号化をクラウド暗号化およびアクセス制御リスト (ACL) と組み合わせて使用すると、機密情報を表示する権限のないログインユーザーから機密情報を保護できます。

フィールド暗号化 には 2 つのバージョンがあります。

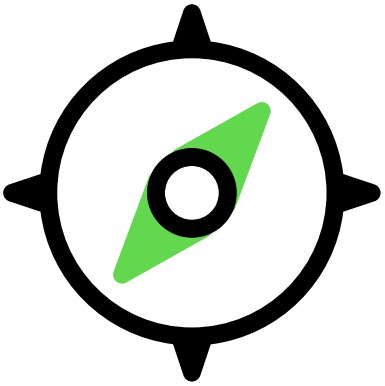
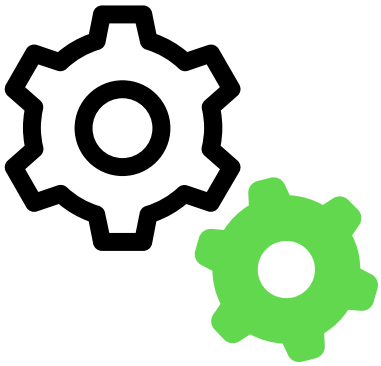
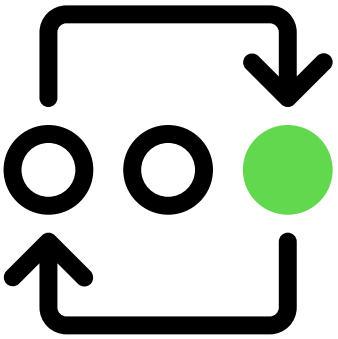
フィールド暗号化 スターター
: フィールド暗号化 スターターは ServiceNow プラットフォームに無料で含まれており、限られた数のフィールドの暗号化をサポートします。

フィールド暗号化 Enterprise
: フィールド暗号化 Enterprise は、 ServiceNow Vault、Platform Encryption、または フィールド暗号化エンタープライズ に登録しているお客様が利用できる製品のプレミアムサブスクリプションバージョンです。

これらのバージョンの違いの詳細については、「 フィールド暗号化 の探索」を参照してください。

フィールド暗号化は、Xanadu 以前のリリースで使用可能な列レベル暗号化製品に代わるものです。

開始

探索	構成	使用方法
		
フィールド暗号化 のスターターバージョンとエンタープライズバージョンのメリットについて説明します。	フィールド暗号化エンタープライズをアクティブ化して構成する方法、および暗号化のサポートまたは列レベル暗号化からの移行を管理する方法について説明します。	フィールド暗号化を使用して、インスタンス上の暗号化されたデータへのアクセスを管理します

役に立つリソース

役立つ情報を提供できる ServiceNow リソースは次のとおりです。

- ServiceNow コミュニティ
  ServiceNow コミュニティ
- Customer Success Center
  Customer Success Center
- ディベロッパー
  developer.servicenow.com
- Impact
  http://impact.servicenow.com
- ServiceNow University
  ServiceNow University
- Now Create
  Now Create
- パートナー
  https://www.servicenow.com/partners.html
- ServiceNow
  https://www.servicenow.com/

自動翻訳

ServiceNow Store

https://store.servicenow.com/

支える

https://support.servicenow.com/now

既知のエラーポータル

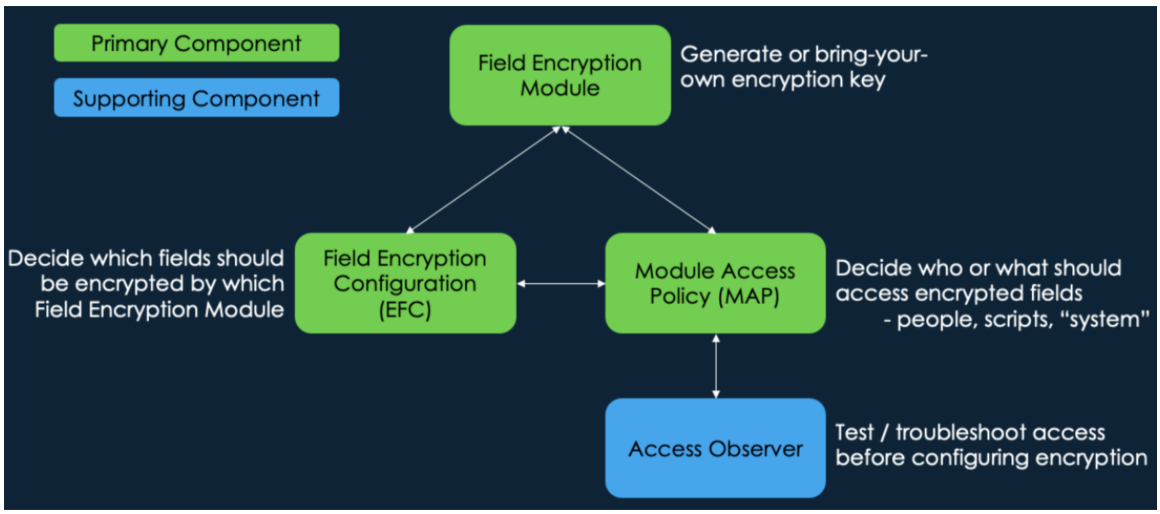
フィールド暗号化 の探索

フィールド暗号化 Starter の詳細とフィールド暗号化エンタープライズ

暗号化に裏打ちされたアクセス制御

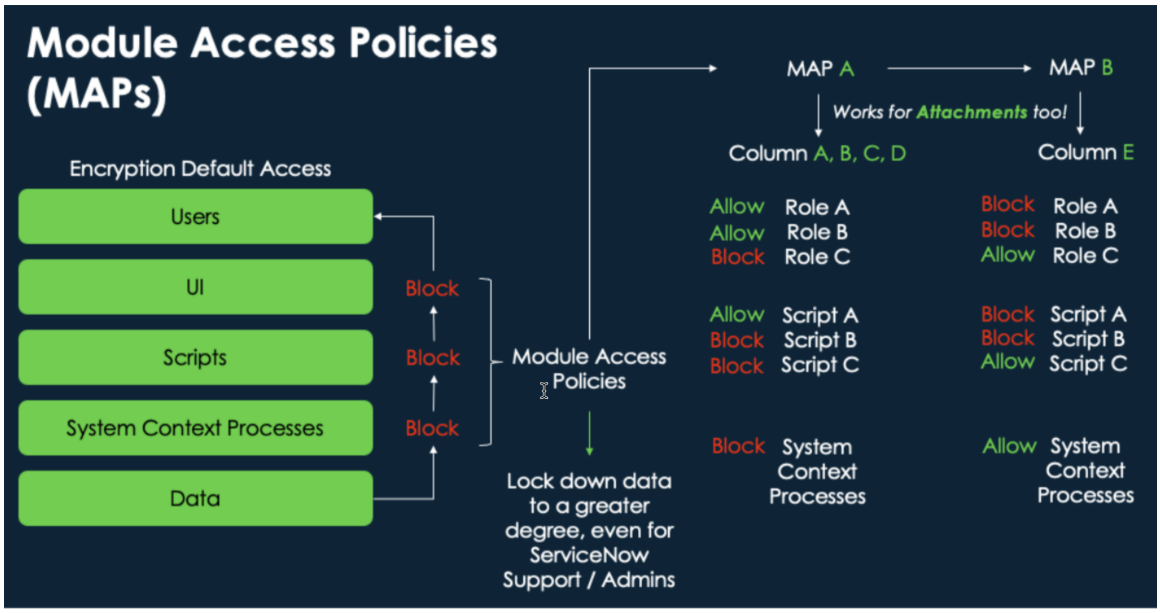
デフォルトでは、 フィールド暗号化 はすべてのユーザー、スクリプト、およびシステムプロセスが暗号化されたデータにアクセスするのをブロックします。ただし、 フィールド暗号化 には、正しいユーザー、スクリプト、またはシステムプロセスのみが暗号化されたデータにアクセスできるようにするために、アクセス制御リスト (ACL) と組み合わせて使用されるアクセス制御機能があります。

フィールド暗号化のアクセス制御機能は、フィールド暗号化モジュール、暗号化フィールド構成、およびモジュールアクセスポリシーを組み合わせて構成できます。次の画像は、これら 3 つのコンポーネントがどのように連携するかを示しています。



モジュールアクセスポリシー (次の画像を参照) は、ユーザー、スクリプト、またはシステムプロセスが暗号化されたデータへのアクセスを承認する方法です (ただし、デフォルトでは、暗号化されたデータはインスタンス内のすべてのアクセスからロックダウンされます)。

自動翻訳



フィールド暗号化スターターとの違いフィールド暗号化エンタープライズ

機能セットは、 フィールド暗号化 スターターと フィールド暗号化エンタープライズで異なります。

機能	フィールド暗号化 スターター	フィールド暗号化エンタープライズ
暗号化フィールドの数	最大 5 つの暗号化フィールド	暗号化フィールドの数に制限なし
添付ファイルの暗号化	いいえ	はい
キー管理	なし (連絡先 ServiceNow キーローテーションのサポート)	ServiceNowサポートの関与なしにインスタンスからキーを管理する
サポートされるデータ型	サポートされているすべてのデータタイプ	サポートされているすべてのデータタイプ
フィールド暗号化モジュールの数	制限なし	制限なし
モジュールアクセスポリシーの数	制限なし	制限なし

フィールド暗号化 ユーザー

ユーザー

ユーザー	説明
Key Management Framework (KMF)Admin または KMF 暗号化マネージャー	これらのロールは、 フィールド暗号化の要素を構成するために使用されます。 • フィールド暗号化 モジュールとモジュールキー • 暗号化仕様 • モジュールライフサイクルポリシー • フィールドと添付ファイルの暗号化フィールド構成

ユーザー (続く)

ユーザー	説明
	• モジュールアクセスポリシー (MAP) • 顧客指定のキー (フィールド暗号化エンタープライズ用) を構成、ラップ、およびアップロード • アクセスオブザーバーを構成し、アクセスオブザーバーログを確認します。 • 一括暗号化、復号化、またはリキーをスケジュールする
KMF 暗号化演算子	顧客指定のキーのプロパティを設定します

フィールド暗号化 およびレコード履歴

フィールド暗号化で暗号化されたフィールドへの変更は、レコードのアクティビティストリームやレコード履歴 [sys_history_set] テーブルでは追跡されません。

次に探索する内容

フィールド暗号化を構成して使用する方法の詳細については、以下を参照してください。

- フィールド暗号化の構成
- フィールド暗号化 の使用

フィールド暗号化の構成

フィールド暗号化をアクティブ化して構成し、暗号化のサポートからの移行を管理する方法について説明します。

フィールド暗号化 のアクティブ化
: いずれかまたは フィールド暗号化エンタープライズを有効にする方法について説明します。

フィールド暗号化の設定に必要なロール
: フィールド暗号化を構成するために必要なロールについて説明します。

暗号化のサポートからの移行

スケジュール済みジョブを使用して、キーと暗号化データを従来の暗号化のサポートから フィールド暗号化エンタープライズ に移行します。このプロセスの詳細については、「フィールド暗号化エンタープライズ への移行」を参照してください。

添付ファイルの暗号化設定の変更

ユーザーが暗号化されていないファイルを添付できないようにすることでセキュリティを向上します。詳細については、「ユーザーが暗号化されていないファイルを添付できないようにする」を参照してください。

フィールド暗号化 のアクティブ化

フィールド暗号化スターターまたはフィールド暗号化エンタープライズのいずれかを有効にします。

始める前に
必要なロール：admin

ServiceNowは、Yokohama リリース以降列レベル暗号化をフィールド暗号化に置き換えました。

Yokohama リリースでは、これまで 列レベル暗号化 を使用していなかった顧客が、新しいエンタイトルメント構造の下で フィールド暗号化 Starter または フィールド暗号化エンタープライズ の使用を開始できます。

以前のリリースで 列レベル暗号化 を使用していて、 フィールド暗号化の使用を開始するお客様には、次のオプションがあります。

フィールド暗号化 スターター

列レベル暗号化 Starter のお客様は、再実装の必要なく フィールド暗号化 Starter をインストールできます。フィールド暗号化 スターターは既存の構成を引き継ぎ、新しい機能をシームレスに追加します。

警告: 列レベル暗号化 Starter と フィールド暗号化 Starter ではエンタイトルメントに違いがあります。フィールド暗号化 Starter をインストールする前に、構成がエンタイトルメントに準拠していることを確認してください。フィールド暗号化 スターターのエンタイトルメントの詳細については、「フィールド暗号化の探索」を参照してください。

フィールド暗号化エンタープライズ

列レベル暗号化 エンタープライズのお客様は、アカウントチームと協力して、 フィールド暗号化エンタープライズの適切なエンタイトルメントがあることを確認する必要があります。これが発生すると、 フィールド暗号化エンタープライズ プラグインをインスタンスにインストールできるようになります。

重要: 列レベル暗号化 エンタープライズでは、 フィールド暗号化エンタープライズ は代替製品とは異なるため、エンタイトルメントは自動的に付与されません。

手順

1. 移動先 すべて > システム定義 > プラグイン
2. [ライセンス済みアプリケーションとプラグインを検索] で、 フィールド暗号化を検索します。検索により、プラグインが明らかになるはずです。フィールド暗号化エンタープライズ のサブスクリプションを購入した場合は、このプラグインも利用できます。

   重要: フィールド暗号化エンタープライズをアクティブ化するには、まずサブスクリプションを購入する必要があります。アカウントマネージャーは、通常数日以内に、組織の本番インスタンスと非本番インスタンスでプラグインがアクティブ化されるように手配できます。

3. [インストール] を選択または フィールド暗号化エンタープライズ、[インストール] を選択します。

   注: ドメインセパレーションと代理アドミンがインスタンスで有効になっている場合、管理ユーザーはグローバルドメインに含まれている必要があります。それ以外の場合、次のエラーが表示されます： 別の操作が実行されているため、アプリケーションのインストールは利用できません： <プラグイン名> のプラグインの有効化。

のロール要件 フィールド暗号化

フィールド暗号化を構成するために必要なロールについて説明します。

管理には次のロールが必要です。は キー管理フレームワークに基づいているため、両方に共通のロールがあります。

- 管理者
- セキュリティアドミン
- KMF アドミン
- KMF 暗号化マネージャー

ロールの詳細については、「 キー管理フレームワークとともにインストールされるロール」を参照してください。

アドミンとセキュリティアドミン

セキュリティアドミンロールに昇格するには、ユーザーにアドミンロールが必要です。暗号化フィールドの構成や Access Observer の構成など、高度なセキュリティタスクを実行するには、セキュリティアドミンロールが必要です。

アドミニストレーターは、この手順を使用してセキュリティアドミニストレーターに昇格できます。

1. 画面の右上にあるプロファイル画像を選択します。
2. ドロップダウンメニューで、[ロールを昇格] を選択します。
3. [セキュリティアドミン] を選択します。
4. **[Save** (保存)] を選択します。

KMF アドミン

admin ロールを持つユーザーは、次のプロセスを使用してユーザーを KMF admin ロールに割り当てることができます。

1. 移動先 すべて > システムセキュリティ > キー管理の管理.
2. [利用可能なユーザー **(Available Users**)] リストから、KMF Admin ロールを必要とするユーザーを [選択したユーザー **(Selected User(s)**)] リストに移動します。
3. **[Save** (保存)] を選択します。

> **重要:** セキュリティ上の問題を防ぐために、複数のユーザーに KMF アドミンロールを付与しないでください。より分化したロールが利用可能な場合は、このロールをユーザーに割り当てないでください。

KMF 暗号化マネージャー

KMF 暗号化マネージャーロールを持つユーザーは、暗号化モジュールおよびモジュールアクセスポリシーに対する操作を作成および更新できます。KMF 暗号化マネージャーは、キー管理とライフサイクル操作を実行することもできます。

このロールをユーザーに割り当てるには、次のプロセスを使用します。

1. 移動先 すべて > システムセキュリティ > ユーザー.
2. フィールド暗号化を構成する必要があるユーザーを選択します。
3. [ロール] 関連リストで、[編集] を選択します。
4. `sn_kmf.cryptographic manager` を検索し、選択したユーザーのロールを追加します。
5. **[Save** (保存)] を選択します。

フィールド暗号化モジュールの構成

フィールド暗号化モジュールの構成方法について説明します。

始める前に

必要なロール:KMF Admin または KMF Cryptographic Manager

手順

1. 移動先 すべて > システムセキュリティ > フィールド暗号化 > フィールド暗号化モジュール.
2. **[New** (新規)] を選択します。
3. [モジュール] フォームで、次に示すようにフィールドに入力します。

フィールド	値
モジュール名	モジュールの名前を選択します。この名前は、スクリプトの実行時に参照されます。
暗号化仕様テンプレート	デフォルトテンプレートが自動的に入力されます。このテンプレートは、暗号化仕様および推奨アルゴリズムに対する多くの暗号化目的のマッピングを含む暗号化モジュールを作成するために使用されます。
アプリケーション	このモジュールのアプリケーションスコープ。このフィールドには現在のアプリケーションが自動的に入力されます。
名前	この名前は自動的に生成されます。他のスコープ対象アプリケーションとの競合を避けるために、アプリケーションスコープ名の先頭に追加されたモジュール名です。たとえば、グローバルアプリケーションスコープで my_crypto_module という名前のモジュールを作成すると、名前は global.my_crypto_module として保存されます。
暗号化モジュールライフサイクルステータス	「ライフサイクル」という用語は、暗号化モジュールの作成、使用、非アクティブ化を指します。この値は、構成時に最初に [ドラフト] に設定します。アクティブに使用するには [公開済み] に設定します。 注: [デフォルト] テンプレートは自動的に [公開] に設定されます。
親暗号化モジュール	フィールド暗号化の場合は、この値が **column_level_encryption** に設定されていることを確認します。

4. **[Submit** (送信)] を選択します。

次のタスク

暗号化キーの目的、アルゴリズム、キーの長さ、モード、および作成元を の暗号化仕様 フィールド暗号化 で設定します。

の暗号化仕様 フィールド暗号化

暗号化仕様を使用して、暗号化キーの目的、アルゴリズム、キーの長さ、モード、および作成元を定義します。

始める前に

必要なロール:KMF Admin または KMF Cryptographic Manager

このタスクについて
この手順では、生成されたキーを構成する方法を示します。暗号化キー (モジュールキー) は、暗号化仕様を構成すると自動的に入力されます。

顧客指定のキー構成については、「 の顧客指定のキーの構成 フィールド暗号化エンタープライズ」を参照してください。

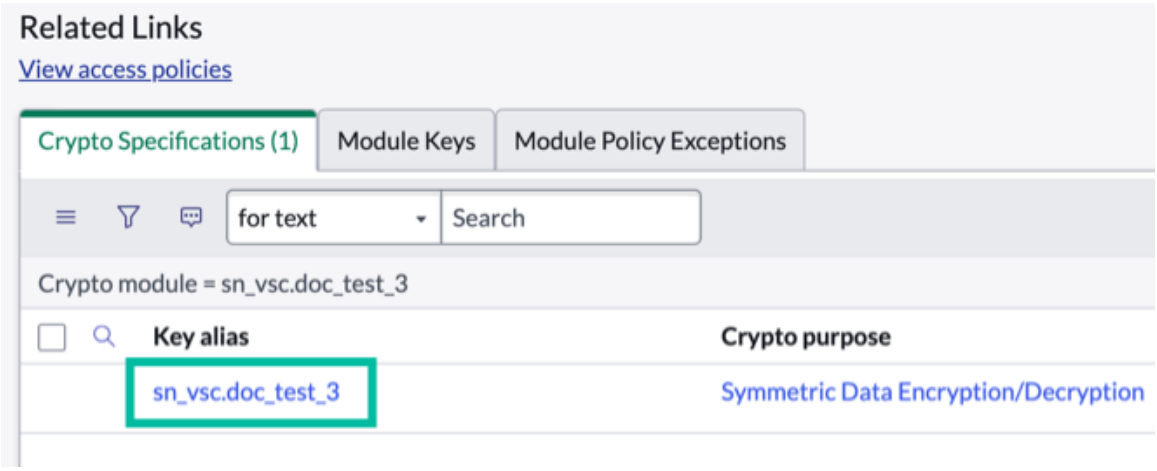
手順

1. 移動先 すべて > システムセキュリティ > フィールド暗号化 > フィールド暗号化モジュール.

2. [暗号化モジュール] リストで、設定するモジュールレコードを開きます。

> **注:** リストに表示される名前には、たとえばグローバルのようにスコープが付加されます 。フィールド暗号化モジュール名]。

3. [暗号化仕様] 関連リストで、[キーエイリアス] の下にある名前を選択して、暗号化固有のレコードを開きます。



4. 暗号化仕様フォームで、必要に応じてフィールドに入力します。

> **注:** フィールドはセクションに分割されています。セクション間を移動するには、[次へ] または [戻る] ボタンを選択します。

セクション	フィールド	説明
アルゴリズム定義	暗号化されたモジュール	選択した暗号化モジュールの名前を表示します。
	暗号化の目的	選択したアルゴリズムの目的、キー、長さ、およびモード。フィールド暗号化の場合、このフィールドは読み取り専用で、値は [対称データの暗号化/復号化] です。
	アルゴリズム	暗号化の目的を達成するために使用するアルゴリズムのタイプを選択します。利用可能なオプションは、選択した暗号化の目的に合わせてフィルタリングされます。
ライフサイクル定義	適用先	ライフサイクルが適用される、選択したキーを表示します。
	フィールド用	ライフサイクルに適用するキーのコントロールタイプを選択します。 ○ 有効期限 ○ 今後のアクティブ化日

セクション	フィールド	説明
		○ 今後の破棄日 ○ 今後の更改日 ○ 今後のローテーション日
	キータイプ	
	ライフサイクルのデフォルト	
	順序	暗号化仕様のキーライフサイクル状況を処理する順序。小さい値は大きい値の前に実行されます。
	相対期間	キーが有効である年数、月数、または日数。
	相対期間タイプ	ライフサイクルの期間タイプ。年、月、または日から選択します。
	相対操作	[前] または [後] を選択します。
	相対的	期間が相対するフィールドを選択します。 ○ アクティブ化日 ○ 侵害日 ○ 非アクティブ化日 ○ 破棄日 ○ 有効期限 ○ 生成日 ○ 前回の更改日 ○ 前回のローテーション日 ○ 失効日 **注:** このフィールドは、[相対操作] フィールドで選択した値と連動します。
	タイプ	キーライフサイクルの値が相対値か絶対値かを選択します。 相対指定 キーの生成、アクティブ化、非アクティブ化など、システム内の他のデータエントリに依存する値を入力します。 絶対 日付などの正確な値を入力します。
キーの作成元	暗号化されたモジュール	選択した暗号化モジュールの名前を表示します。
	作成元	キーが顧客から発信されたか、顧客によって提供されたか。 ○ まず、**ServiceNow** を選択します ○ Enterprise の場合は、[**Servicenow**] または [顧客指定] を選択します。

セクション	フィールド	説明
	キーのエイリアス	名前の先頭にスコープが付加された暗号化モジュールの名前。
	暗号化の目的	選択したアルゴリズムの目的、キー、長さ、およびモードを表示します。フィールド暗号化の場合、このフィールドは読み取り専用で、値は [対称データの暗号化/復号化] です。
	アルゴリズム	暗号化の目的を達成するために使用されるアルゴリズムを表示します。
キーの作成	暗号化されたモジュール	選択した暗号化モジュールの名前を表示します。
	キーのエイリアス	暗号化モジュールの名前を、名前の先頭にスコープを追加して表示します。
	キーの生成	顧客指定のキーではなく、生成されたキーを使用している場合は、このリンクを選択してデータ暗号化キーを生成します。
	キーの自動生成	[キーの生成] リンクを選択しない場合、暗号化モジュールを使用して初めてデータを暗号化するときに、データ暗号化キーが自動的に生成されます。
	暗号化の目的	選択したアルゴリズムの目的、キー、長さ、およびモードを表示します。フィールド暗号化の場合、このフィールドは読み取り専用で、値は [対称データの暗号化/復号化] です。
	作成元	[キーの作成元] セクションで選択された値を表示します。
	アルゴリズム	暗号化の目的を達成するために使用されるアルゴリズムを表示します。

5. [暗号化モジュールに移動] を選択して、モジュールレコードに戻ります。
   モジュールレコードの [モジュールキー] 関連リストにエントリが追加されました。キーエイリアスが新しいモジュールキーに移動されたため、[暗号化仕様] 関連リストの [キーエイリアス] フィールドは空になりました。

次のタスク
顧客指定のキーの使用については、「 の顧客指定のキーの構成 フィールド暗号化エンタープライズ」を参照してください。

のモジュールキー フィールド暗号化

[モジュールキー] タブには、フィールド暗号化のデータ暗号化キーに関するサマリーレベルの情報が表示されます。キーのエイリアス、キータイプ、アルゴリズム、キーのライフサイクルステータス、およびキーバージョンを表示できます。

モジュールキー情報へのアクセス

暗号化モジュール [sys_kmf_crypto_module] テーブルのレコードからモジュールキーにアクセスします。暗号化モジュールレコードから、[モジュールキー] 関連リストのレコードの [キーエイリアス] フィールドを選択できます。

利用可能な情報は、インスタンスでアクティブにしている フィールド暗号化 のバージョンによって異なります。

フィールド暗号化 スターター

フィールド暗号化 スターターを使用して、主要な使用状況監査データを表示できます。

フィールド暗号化エンタープライズ

キー使用状況の監査データを表示することに加えて、更新、取り消し、ローテーション、一時停止などの手動キー操作を実行できます。

モジュールキー情報

モジュールキーに関する情報は、ここにリストされているフィールドのモジュールキーレコードに表示されます。これらのフィールドは読み取り専用であり、情報提供のみを目的として使用されます。

フィールド	説明
生成日	モジュールキーが生成されたタイミング
アクティブ化日	モジュールキーがアクティブ化されたとき。
前回の更改日	モジュールキーが最後に更新された日時。
前回のローテーション日	モジュールキーが最後にローテーションされた日時。
非アクティブ化日	モジュールキーが非アクティブ化されたとき。
破棄日	モジュールキーが破棄されたとき。
今後のアクティブ化日	モジュールキーがアクティブになるタイミング。
今後の更改日	モジュールキーが更新されるタイミング。
今後のローテーション日	モジュールキーがローテーションされるタイミング。
今後の破棄日	モジュールキーが破棄されるタイミング。
キーのライフサイクルステータス	現在のモジュールキーのステータス。

のモジュールライフサイクルポリシーの例外 フィールド暗号化

モジュールライフサイクルポリシー例外を使用して、モジュールキーのライフサイクルをカスタマイズします。

始める前に

必要なロール:KMF Admin または KMF Cryptographic Manager

このタスクについて

モジュールライフサイクルポリシー例外は、 フィールド暗号化 モジュールのライフサイクルポリシーを標準のインスタンスレベルのライフサイクルポリシーから変更します。たとえば、インスタンスレベルで対称キーを 1 年間に制限するように設定した場合は、特定の フィールド暗号化 モジュールに対してモジュールライフサイクルポリシーの例外を作成して、そのキーを 2 年間アクティブにしておくことができます。

手順

1. 移動先 すべて > システムセキュリティ > フィールド暗号化 > フィールド暗号化モジュール.
2. モジュールライフサイクルポリシーの例外が必要なフィールド暗号化モジュールレコードを選択します。

自動翻訳

3. フィールド暗号化モジュールレコードで、[モジュールポリシー例外] 関連リストの [新規] を選択します。
4. 必要に応じて、キーライフサイクルポリシーフォームのフィールドに入力します。

フィールド	説明
暗号化されたモジュール	このポリシー例外を使用するフィールド暗号化モジュールの名前を表示します。
適用先	指定されたキーが自動入力されます。
キータイプ	キータイプを選択します。例外ポリシーは特定のキーに関連しています。モジュールごとに複数の例外ポリシー フィールド暗号化 作成できます。
ポリシー条件	ドロップダウンメニューから認定条件を作成し、追加の制約条件を入力します。
結果	キーの使用を却下する場合は [却下] を、条件が満たされた場合にキーの使用を許可する場合は [トラック] を選択します。

5. **[Submit** (送信)] を選択します。

の顧客指定のキーの構成 フィールド暗号化エンタープライズ

ServiceNowが生成するキーを使用するのではなく、独自のデータ暗号化キーをプラットフォームに使用します。

始める前に

必要なロール:KMF Admin または KMF Cryptographic Manager

このタスクについて

フィールド暗号化エンタープライズを使用している場合は、ServiceNowによって生成されたキーではなく、独自のデータ暗号化キーをプラットフォームに使用できます。

ServiceNow の外部で生成された対称キーが必要です。このドキュメントの例は、OpenSSL に依存しています。OpenSSL の詳細については、 https://www.openssl.org で詳細を参照してください。LibreSSL や GnuTLS などの他の暗号化ツールを使用している場合は、それらの製品のドキュメントで類似した手順について参照してください。

手順

1. マシンのコマンドライン(例:ターミナル)で、次のコマンドを実行します: `openssl rand -hex 32 -out mykey.bin`。
   次の手順で使用する `mykey.bin` ファイルを保存します。
2. インスタンスで、 すべて > システムセキュリティ > フィールド暗号化 > フィールド暗号化設定.
3. [キーソース] フィールドを **[ServiceNow** によって生成されたキー] から [顧客指定のキー] に変更します。
4. **[Submit** (送信)] を選択します。

次のタスク

次の手順に従って、インスタンスに作成した対称キーを使用します。

1. 顧客指定のキーのプロパティの構成
2. 顧客指定のキーをラップする
3. 顧客指定のキーをアップロードする

顧客指定のキーのプロパティの構成

顧客指定のキーのラップ解除にインスタンスが使用する短期公開ラッピングキーのシステムプロパティを確認します。

始める前に
必要なロール:KMF admin または KMF 暗号化オペレーター

このタスクについて
対称データ暗号化キーをインスタンスにアップロードする前に、 ServiceNow エフェメラル公開ラッピングキーでラップする必要があります。

キーがインスタンスにアップロードされると、インスタンスは公開鍵のプライベート側を使用してキーをラップ解除します。

インスタンスのシステムプロパティを使用して、キーのパディング、短期キーペアのサイズ、およびこの短期公開キーのキー有効期間を定義できます。

システムプロパティ	説明	デフォルト値
glide.kmf.ephemeral_key.key_padding	短期キーペアのパディングスキーム。	`OAEPWithSHA256AndMGF1Padding` OAEP SHA256 ですが、SHA1 はサポートされています。
glide.kmf.ephemeral_key.key_size	短期キーペアのキーサイズ。	`4096` 4096 ビットですが、2048 ビットもサポートされています。
Glide.kmf.ephemeral_key.key_validity_period(有効期間)	短期キーペアが有効な期間。	`02:00:00` 2 時間

手順

これらのプロパティのいずれかを変更する必要がある場合は、 ServiceNow サポートにお問い合わせください。

> **注:** これらのシステムプロパティはアドミニストレーターには表示されず、システムプロパティ [sys_properties] リストにも表示されません。上のテーブルを使用して、デフォルト値を確認してください。

次のタスク
プロパティをニーズに合わせて構成したら、 顧客指定のキーをラップするに進みます。

顧客指定のキーをラップする

対称データ暗号化キーを、インスタンスにアップロードする前に、エフェメラル公開ラッピングキーでラップします。

始める前に
必要なロール:KMF admin または KMF 暗号化オペレーター

これらの手順を使用するには、 .bin に対称データ暗号化キーが必要です。このプロセスの手順については、「 の顧客指定のキーの構成 フィールド暗号化エンタープライズ」を参照してください。

重要: 対称データ暗号化キーはバイナリ形式 (.BIN)。別の形式が使用されている場合、次のエラーメッセージが表示されます。

トークンの検証に失敗しました。変更されていないトークンを再添付してください。

このタスクについて
キーのサイズ、パディングアルゴリズム、および有効期間を制御するオプションのプロパティを変更するには、「 顧客指定のキーのプロパティの構成」を参照してください。

キーをラップするには、暗号化ツールが必要です。このドキュメントの例では、OpenSSL 1.1 を使用しています。OpenSSL の詳細については、 https://www.openssl.org で詳細を参照してください。LibreSSL や GnuTLS などの他の暗号化ツールを使用している場合は、それらの製品のドキュメントで同様の手順について参照してください。

手順

1. 移動先 すべて > システムセキュリティ > フィールド暗号化 > フィールド暗号化モジュール.
2. 以前に作成したフィールド暗号化モジュールを開きます。

   注: フィールド暗号化モジュールをまだ作成していない場合は、「 フィールド暗号化モジュールの構成」の手順を使用して作成できます。

3. [モジュール] 関連リストで、[キーのエイリアス] の下にある名前を選択して、暗号化固有のレコードを開きます。
4. [キーの作成元] セクションが表示されるまで [次へ] ボタンを選択します。
5. [作成元] フィールドの値が [顧客指定のキーをアップロードする] であることを確認します。
   設定されていない場合や、その値を選択できない場合は、 の顧客指定のキーの構成 フィールド暗号化エンタープライズの手順 3 ～ 5 を参照してください。
6. [キーエイリアス] フィールドで、エイリアスを作成します。
   キーは、アップロードされるとこのエイリアスを使用します。
7. **[Next** (次へ)] をクリックします。
8. [ラッピングキーのダウンロード] フィールドのリンクを選択します。

   token_publickeyファイルがコンピューターにダウンロードされます。このファイルの名前は変更しないでください。

9. ローカルマシンで、 token_publickey フォルダーを解凍して開きます。
   インポートトークンファイル (.txt) と公開鍵ファイル (.PEM) をこのフォルダーに格納します。
10. 生成した対称データ暗号化キーをこのフォルダーに移動します。
11. token_publickeyファイルの名前をクリップボードにコピーします。
12. ターミナルセッションを開き、 token_publickey フォルダーに移動します。
13. 次のコマンドを入力します。

重要: 括弧で囲まれたテキスト (<>) を特定のファイル名と情報に置き換えます。次のキーラッピングコマンドの例の表を参考にしてください。

自動翻訳

openssl pkeyutl -encrypt -pubin -inkey publickey_<keyname>。PEM -in <keyname.bin> -out wrapped_key_material -pkeyopt rsa_padding_mode: oaep -pkeyopt rsa_oaep_md:sha<128 または 256>

キーラッピングコマンドの例

方向	コマンド	例
publickey_<keyname> を入力します。エンティティタイプ:	openssl pkeyutl -encrypt -pubin -inkey publickey_<keyname>。エンティティタイプ:	openssl pkeyutl -encrypt -pubin -inkey publickey_567898643ffff。エンティティタイプ:
対称データ暗号化キーの名前を入力します	-in <keyname.bin>	-mykey.bin中
ラップされたキーマテリアルが 128 ビットか 256 ビットかを指定する <-out> コマンドを入力します	-out wrapped_key_material -pkeyopt rsa_padding_mode:oaep -pkeyopt rsa_oaep_md:sha256	NA

次のタスク
キーがラップされたので、「 顧客指定のキーをアップロードする」の手順を使用してインスタンスにアップロードできます。

顧客指定のキーをアップロードする

ラップされた対称データ暗号化キーをインスタンスにアップロードして、暗号化の使用を開始します。

始める前に
必要なロール:KMF Admin または KMF Cryptographic Manager

このタスクについて

注: 独自のキーを指定しない場合は、「 フィールド暗号化モジュールの構成 」の手順を使用して ServiceNowを使用できます。顧客指定のキーを取り消すことはできません。

手順

1. 移動先 すべて > システムセキュリティ > フィールド暗号化 > フィールド暗号化モジュール.
2. キーを使用するフィールド暗号化モジュールを開きます。
3. [モジュール] 関連リストで、[キーのエイリアス] の下にある名前を選択して、暗号化固有のレコードを開きます。
4. [キーの作成元] セクションが表示されるまで [次へ] ボタンを選択します。
5. [作成元] フィールドの値が [顧客指定のキーをアップロードする] であることを確認します。
   設定されていない場合や、その値を選択できない場合は、 の顧客指定のキーの構成 フィールド暗号化エンタープライズの手順 3 ～ 5 を参照してください。
6. [キーエイリアス] フィールドに値があることを確認します。
7. **[Next** (次へ)] をクリックします。
8. [顧客指定のキーをアップロード] リンクを選択します。
   このリンクは、キーのラッピングの一環として選択した [ラッピングキーのダウンロード] リンクの下に表示されます。

9. [参照] を選択し、次の 2 つのファイルを選択します。
   a. wrapped_key_materialファイル
   b. 「インポートトークン」ファイル
10. **[OK]** を選択します。

結果
顧客指定のキーが正常にアップロードされたことを示す確認メッセージが表示されます。キーは、顧客指定のキーの作成元とともに [モジュールキー] 関連リストにも一覧表示されます。

暗号化キーが構成されたので、暗号化するフィールドと添付ファイルの指定を開始できます。詳細については、「フィールドまたは添付ファイルの暗号化フィールドの構成」を参照してください。

フィールドまたは添付ファイルの暗号化フィールドの構成

暗号化フィールド構成を作成して、テーブルでどのフィールドを暗号化するか、およびそのテーブルの添付ファイルを暗号化するかどうかを指定します。

始める前に
必要なロール:KMF Admin または KMF Cryptographic Manager、Security Admin

ServiceNowまたは顧客指定のキーを使用してフィールド暗号化モジュールを構成する必要があります。モジュールをまだ構成していない場合は、「 フィールド暗号化モジュールの構成」を参照してください。

手順

1. 暗号化するテーブルと同じアプリケーションスコープ内にいることを確認してください。
2. 移動先 すべて > システムセキュリティ > フィールド暗号化 > 暗号化フィールドの設定.
3. **[New** (新規)] を選択します。
4. 暗号化フィールドの構成フォームで、必要に応じてフィールド内のフィールドを入力します。

フィールド	値
タイプ	列または添付ファイルのいずれかを選択 注: 添付ファイルの暗号化は、 フィールド暗号化エンタープライズでのみ使用できます。
テーブル	フィールドまたは添付ファイルを暗号化するテーブルを選択します。
列	[タイプ] フィールドで [列] を選択した場合は、暗号化するフィールドを選択します。

フィールド	値
	注: 暗号化するフィールドが利用できない場合は、サポートされているタイプではない可能性があります。サポートされているフィールドタイプは次のとおりです。 ○ 文字列 (完全な UTF-8 を含む) ○ 日付 ○ 日付/時刻 ○ URL ○ HTML ○ ジャーナル ○ 翻訳済み ○ メール ○ 電話
アクティブ	構成がアクティブかどうか。 **重要:** アクティブな場合、インスタンスは選択したフィールドまたは添付ファイルの新しいデータをアクティブに暗号化しています。ユーザーは、関連するモジュールアクセスポリシーによる権限がない限り、このデータにアクセスできません。モジュールアクセスポリシーの暗号化と適用を開始する準備ができていないフィールドをオフにしてください。 暗号化フィールド構成がアクティブになった後に履歴データが暗号化されるようにするには、列で一括暗号化ジョブを実行する必要があります。詳細については、「一括暗号化、復号化、またはリキーのスケジュール」 を参照してください。
暗号化されたモジュール	この暗号化フィールド構成で使用されるフィールド暗号化モジュール。
メソッド	すべてのフィールドまたは添付ファイルが単一フィールド暗号化モジュールで暗号化されるようにするには、[単一モジュール] を選択します。 列内の異なる行または異なる添付ファイルに異なるフィールド暗号化モジュールを使用できるようにするには、[マルチモジュール] を選択します。マルチモジュール構成の詳細については、「 」を参照してください。
アルゴリズム等価性保存	[暗号化モジュール] フィールドで選択したフィールド暗号化モジュールで等価性保存が有効になっているかどうかを表示します。

5. **[Submit** (送信)] を選択します。

マルチモジュール暗号化フィールド構成を構成する

複数の暗号化モジュールを使用する暗号化フィールド構成を作成します。

始める前に

必要なロール:KMF Admin または KMF Cryptographic Manager、Security Admin

ServiceNowまたは顧客指定のキーを使用してフィールド暗号化モジュールを構成する必要があります。モジュールをまだ構成していない場合は、「 フィールド暗号化モジュールの構成」を参照してください。

このタスクについて

単一の暗号化フィールド構成に複数の暗号化モジュールを使用し、異なるモジュールキーを使用して列内の異なる行 (または同じテーブルの異なる添付ファイル) を暗号化します。たとえば、異なるロールを持つユーザーは同じテーブルのデータを暗号化できますが、互いの暗号化されたデータを復号化することはできません。

⚠ 警告:

続行する前に、マルチモジュール暗号化フィールドの構成に関する次の制限に注意してください。

- 一括暗号化は、マルチモジュール暗号化フィールド構成ではサポートされていません。
- フィールド構成をマルチモジュールから単一モジュールに変更することはできません。代わりに、マルチモジュールフィールド構成を無効にして、新しい単一モジュールフィールド構成を作成する必要があります。
- マルチモジュールフィールド構成で使用するモジュールキーは、フィールドにデータを入力する最初のユーザーによって決定されます。フィールド暗号化モジュールはレコードごとに設定されるため、リスト内のフィールドはさまざまなフィールド暗号化モジュールで暗号化できます。ただし、1 つのレコード内では、フィールドは 1 つのフィールド暗号化モジュールでのみ暗号化できます。

手順

1. 暗号化するテーブルと同じアプリケーションスコープ内にいることを確認します。
2. 使用するフィールド暗号化モジュールが作成されていることを確認します。
   まだ行っていない場合は、 フィールド暗号化モジュールの構成を参照してください。
3. 各モジュールにモジュールアクセスポリシーがあることを確認します。
   まだ行っていない場合は、 フィールド暗号化のモジュールアクセスポリシーの構成を参照してください。
4. 移動先 すべて > システムセキュリティ > フィールド暗号化 > 暗号化フィールドの設定.
5. 暗号化フィールド設定レコードを開くか作成します。
6. [メソッド] フィールドで、[複数のモジュール] を選択します。
7. 必要に応じて、[タイプ] フィールドで [列] または [添付ファイル] を選択します。
8. 該当する場合は、[テーブル] フィールドでテーブルを選択し、[列] フィールドで列を選択します。
9. **[Submit** (送信)] を選択します。

結果

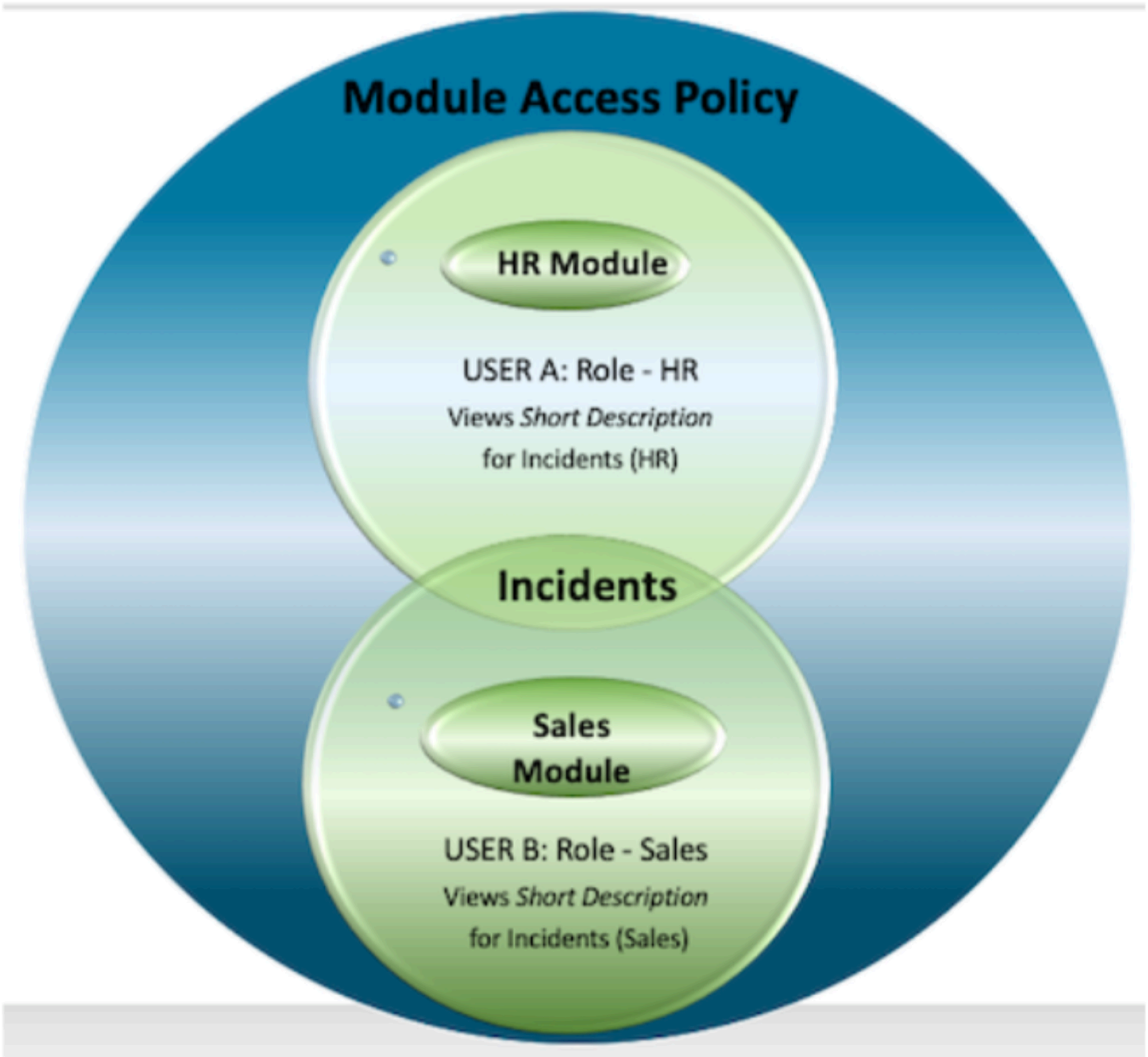
レコードが保存され、[アクティブ] フィールドが有効になると、指定されたフィールドに作成された新しいデータは、関連するフィールド暗号化モジュールのモジュールキーで暗号化されます。モジュールアクセスポリシー「A」のロールを持つユーザーが指定されたテーブルに書き込むと、デー

タはフィールド暗号化モジュール「A」のモジュールキーで暗号化されます。この場合、同じロールを持つユーザーのみがそのデータを復号化できます。

Example: マルチモジュール暗号化フィールド構成を使用したインシデントテーブルの [簡単な説明] 列の暗号化

1. この例では A と B という 2 つのフィールド暗号化モジュールを作成します。
2. フィールド暗号化モジュールごとに、モジュールアクセスポリシー (MAP A および B) を作成します。
   a. フィールド暗号化モジュール A の場合、モジュールアクセスポリシー A への HR ロールアクセス権をユーザーに付与します。
   b. フィールド暗号化モジュール B の場合、セールスロールを持つユーザーにモジュールアクセスポリシー B へのアクセス権を付与します。
3. インシデントテーブルの [簡単な説明] 列を指定する暗号化フィールド設定レコードを作成し、[メソッド] フィールドで [複数のモジュール] を選択していることを確認します。
4. 2 人のユーザーがいる:
   - マップ A に関連する HR ロールとフィールド暗号化モジュール A (ユーザー A) を持つ 1 つ
   - マップ B に関連する営業ロールを持つ 1 つとフィールド暗号化モジュール B (ユーザー B) を持つ 1 つ

   簡単な説明の値を使用してインシデントレコードを作成します。両方のユーザーにインシデントのリストを見てもらいます。
   a. ユーザー A によって作成されたインシデントレコードの簡単な説明は、フィールド暗号化モジュール A のキーで暗号化されます。
   b. ユーザー B によって作成されたインシデントレコードの簡単な説明は、フィールド暗号化モジュール B のキーで暗号化されます。
5. HR ロールと営業ロールを持つユーザーは、インシデントにアクセスできます。HR ロールを持つユーザーのみが、(HR ロールを持っていた) ユーザー A によって作成されたインシデントの簡単な説明を復号化して表示できます。営業ロールを持つユーザーのみが、ユーザー B (営業ロールを持つ) によって作成されたインシデントの簡単な説明を復号化して表示できます。



フィールド暗号化のモジュールアクセスポリシーの構成

モジュールアクセスポリシーを作成して、フィールド暗号化モジュールで暗号化されたデータを暗号化または復号化できるユーザー、スクリプト、またはシステムプロセスを制御します。

始める前に

必要なロール:KMF Admin または KMF Cryptographic Manager、Security Admin

このプロセスを使用するには、公開されたフィールド暗号化モジュールが必要です。まだ行っていない場合は、 フィールド暗号化モジュールの構成を参照してください。

このタスクについて

モジュールアクセスポリシー (MAP) は、フィールド暗号化モジュールに適用して、データを暗号化または復号化できるユーザー、スクリプト、またはシステムプロセスを定義するアクセス制御です。「システム」コンテキストで実行されているユーザー、スクリプト、またはプロセスの MAP を (ロールを介して) 構成します。MAP がないと、ユーザー、スクリプト、またはシステムプロセスはデータを暗号化または復号化できないため、エンドツーエンドのワークフロープロセスが正しく機能しなくなる可能性があります。

MAP はアクセス制御リスト (ACL) とは別ですが、ACL(Access Control List)と組み合わせて使用できます。MAP の目的の詳細については、 フィールド暗号化 の探索 を参照してください。

フィールド暗号化エンタープライズについては、を確認して、マップを必要とするユーザー、スクリプト、またはシステムプロセスを計画します。

手順

1. 移動先 すべて > キー管理 > モジュールアクセスポリシー > すべて.
2. **[New** (新規)] を選択します。
3. 必要に応じて、[モジュールアクセスポリシー] フォームのフィールドに入力します。

フィールド	説明
ポリシー名	マップの名前
暗号化されたモジュール	このマップによって管理されるフィールド暗号化モジュールを選択します。
暗号化仕様	オプション。この MAP の暗号化仕様を選択または新規作成します。 このフィールドは、[目的を指定] フィールドが有効になっている場合にのみ表示されます。
タイプ	データを暗号化または復号化するために、この MAP へのアクセス権を持つユーザーまたは対象を決定します。 スコープ 指定されたアプリケーションスコープ内のすべてのものがこのマップにアクセスできます。 ロール 特定のロールを持つユーザーのみがこのマップにアクセスできます。 スクリプト 指定されたスクリプトがこのマップにアクセスできることを確認します。 システムアクセス 「システムコンテキスト」で実行されているプロセスがこの MAP にアクセスできるようにします。 リソース交換 このマップへのリソース交換機能アクセスを許可します。 これらのさまざまなタイプの MAP の仕組みの詳細については、「フィールド暗号化 の探索」を参照してください。
ターゲットスコープ	このマップが適用されるスコープを選択します。 このフィールドは、[タイプ] フィールドが [スコープ] に設定されている場合にのみ表示されます。
目的を指定	オプション。有効にすると、フォームに [暗号化仕様] フィールドが表示されます。一部のユーザーが暗号化はできるが復号化できないなど、詳細な操作を構成するには、このオプションを有効にします。
詳細な操作	オプション。暗号化仕様の暗号化の目的を選択します。使用可能な値は、選択した暗号化仕様のタイプによって異なります。 たとえば、この MAP では、ユーザーに暗号化のみを許可し、復号化は許可しないように、またはその逆、またはその両方を指定できます。

フィールド	説明
	このフィールドは、[暗号化仕様] フィールドに値がある場合にのみ表示されます。 ○ ユーザーが暗号化アクセス権を持っているが、復号化アクセス権がない場合、フィールドは編集モードで表示され、入力されたデータはアスタリスクとして表示されます。 ○ ユーザーが復号化アクセス権を持っているが、暗号化アクセス権がない場合、フィールドには復号化されたデータが読み取り専用モードで表示されます。 ○ ユーザーが暗号化と復号化のアクセス権を持っている場合、暗号化フィールドでは読み取りと書き込みの両方の機能を使用できます。
ターゲットロール	このマップにアクセスするロールを選択します。 このフィールドは、[タイプ] フィールドが [ロール] に設定されている場合にのみ表示されます
スクリプトテーブル	このマップに適用するスクリプトのタイプを選択します。 ○ アクセス制御 ○ アクティビティデザイナー ○ ビジネスルール ○ 受信メールアクション ○ レコードプロデューサー ○ スクリプトの実行を予定 ○ スクリプトインクルード ○ UI アクション ○ ウィジェット ○ ワークフローアクティビティ このフィールドは、[タイプ] フィールドが [スクリプト] に設定されている場合にのみ表示されます。
ターゲットスクリプト	このマップにアクセスする必要がある [スクリプトテーブル] フィールドで選択したテーブルタイプの特定のスクリプトを選択します。 このフィールドは、[タイプ] フィールドが [スクリプト] に設定されている場合にのみ表示されます。
スクリプトバージョンのチェック	選択すると、実行されるスクリプトのバージョンが [ターゲットスクリプト] フィールドで指定されたバージョンでチェックされます。バージョンが異なる場合は、アドミニストレーターに通知されます。 このフィールドは、[タイプ] フィールドが [スクリプト] に設定されている場合にのみ表示されます。
承認タイプ	[**1** 回] または [繰り返し] を選択します。 ワンタイム

フィールド	説明
	関連付けられたフィールド暗号化モジュールの対称データ暗号化キーを、ターゲットインスタンスと 1 回だけ安全に共有できます。 繰り返し 関連付けられたフィールド暗号化モジュールの対称データ暗号化キーを、繰り返しターゲットインスタンスと安全に共有できるようにします。 このフィールドは、[タイプ] フィールドが [リソース交換] に設定されている場合にのみ表示されます。
ターゲットインスタンスホスト	関連するフィールド暗号化モジュール内の対称データ暗号化キーの送信先のターゲットインスタンスの URL を入力します。 このフィールドは、[タイプ] フィールドが [リソース交換] に設定されている場合にのみ表示されます。
代理操作	有効にすると、別のユーザーの代理操作を行うユーザーは、両方のユーザーから MAP 権限を取得します。無効にすると、別のユーザーの代理操作を行うユーザーには、代理操作の前から付与された MAP 権限のみが付与されます。
アクティブ	この MAP を有効にすることができます。
結果	次のいずれかを選択します。 トラック MAP へのアクセスを許可し、使用を監視します。 却下 別の MAP がアクセスを許可しない限り、アクセスを拒否します。 StrictReject 別の MAP がアクセスを許可する場合でも、すべての状況下でアクセスを拒否します。

4. **[Submit** (送信)] を選択します。

フィールド暗号化エンタープライズ への移行

スケジュール済みジョブにより、キーと暗号化データが暗号化のサポートから フィールド暗号化エンタープライズ に移行します。

スケジュール済みジョブを確認するには、次に移動します: システムセキュリティ > 高セキュリティ設定 > セキュリティジョブ:

- **autoKeyMigration**：暗号化コンテキストキーを キー管理フレームワーク (KMF) 暗号化モジュールキーに移行します。
- **autoDataMigration**：既に暗号化されているデータを KMF 暗号化モジュールキーを使用するように移行します。

スケジュール済みジョブをいつ実行するかを変更できます。また、いつでも一時停止または再開できます。

次の場所に移動して、暗号化フィールドの構成で新しく移行したモジュールキーが使用されていることを確認します。 システムセキュリティ > フィールド暗号化 > 暗号化フィールドの設定. 次のアイテムを探します。

- [メソッド] フィールドは単一モジュールです。
- [暗号化モジュール] フィールドには、自動的に作成される暗号化モジュールの名前が入力されます。そのモジュールとモジュールアクセスポリシーを確認できます。どちらもアクティブで、公開されています。

フィールド暗号化エンタープライズ およびシステムクローン

フィールド暗号化エンタープライズ がインスタンスにインストールされている場合、クローンプロセスの一環として、ターゲットのクローンインスタンスに新しいフィールド暗号化モジュール暗号化キーが自動的に生成されます。これらのキーは、ユーザーがアクセス権を持ち、まだキーを持っていないすべてのモジュールに対して生成されます。

このため、ターゲットクローンインスタンスのフィールド暗号化モジュールには、次の 2 つのモジュール暗号化キーが存在する場合があります。

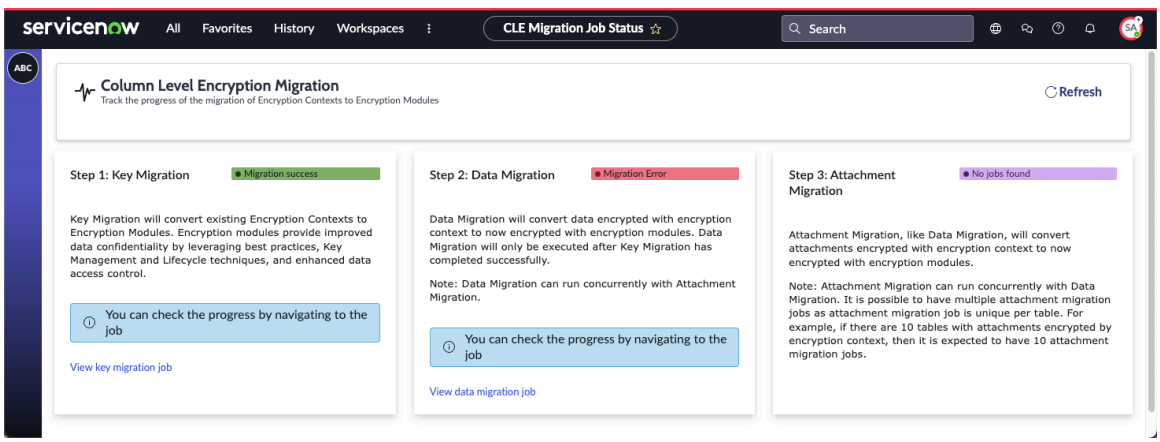
- アクティブなモジュール暗号化キー。これは、ユーザーがモジュールにアクセスできて、以前のキーがない場合に、クローン後に生成される新しいキーです。
- 非アクティブ化された暗号化モジュールキー (自動キー交換転送から)

  アクティブなモジュール暗号化キーは、必要に応じてターゲットクローンインスタンスで挿入されたデータを暗号化するために使用されます。非アクティブ化されたモジュールは、システムクローンの一部としてクローン作成された既存のデータを復号化するために使用されます。

  単一のキーを使用してすべてのデータを復号化および暗号化するには、モジュールのリキージョブを実行します。モジュールのリキージョブの詳細については、「一括暗号化、復号化、およびリキージョブをスケジュールする」を参照してください。

フィールド暗号化 移行ステータスページ

移行ステータスページを使用して、暗号化コンテキストから暗号化モジュールへの移行を追跡します

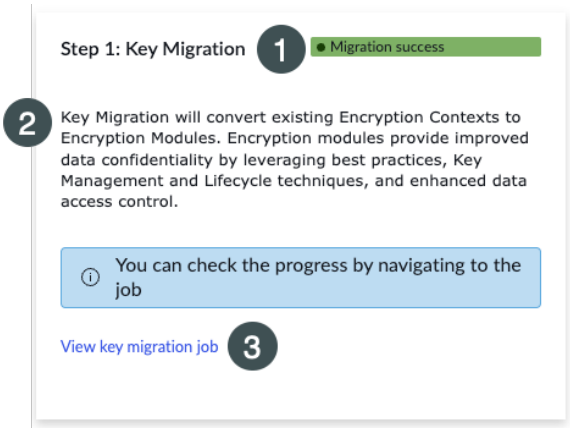


フィールド暗号化 移行ページには、暗号化コンテキストの暗号化モジュールへの移行に関連するステップのステータスが表示されます。3 つのセクションにはそれぞれ、プロセス内の特定のステップのステータスが表示されます。

ページセクションカード

このページには、移行のステップを表す 3 つのカードが含まれています。これらのカードには次のものが表示されます。

- (1) 現在のステップのステータス。このステータスには、ステップが正常に完了したかどうか、または処理するジョブがないかどうかが表示されます。
- (2) リストされたステップの説明。
- (3) 関連する暗号化ジョブ [sys_mass_encryption_job] レコードへのリンク。



エッジ暗号化からフィールド暗号化への移行

エッジ暗号化から フィールド暗号化 に移行して、最新のセキュリティ機能を活用します。

移行プロセス

このトピックでは、次の手順について説明します。

1. 列と添付ファイルをエッジ暗号化から フィールド暗号化に移行します。
2. エッジ暗号化プロキシサーバーを停止して無効化します。
3. データのトークン化を解除
4. サービスカタログアイテム変数を復号化します。

制限事項

- フィールド暗号化 は現在、エッジ暗号化と同じ方法でのデータのトークン化をサポートしていません。エッジ暗号化によってトークン化されたデータは、 フィールド暗号化暗号化フィールド構成に含める必要があります。
- フィールド暗号化 は現在、サービスカタログアイテム変数の暗号化をサポートしていません。

エッジ暗号化フィールドの フィールド暗号化 を構成する

既存のエッジ暗号化フィールドを フィールド暗号化に移行する前に、これらのフィールドのフィールド暗号化を構成する必要があります。

始める前に
必要なロール：admin

このプロセスでは、 フィールド暗号化に移行する既存のエッジ暗号化フィールドがあることを前提としています。

手順

1. 移動先 すべて > キー管理 > 暗号化モジュール > すべて > .
2. [暗号化モジュール] リストで、[新規] を選択します。
3. フォームのフィールドに入力します。

フィールド	値
名前	わかりやすい名前を選択
暗号化仕様テンプレート	デフォルトテンプレート
暗号化モジュールライフサイクルステータス	公開済み
親暗号化モジュール	column_level_encryption

4. フォームヘッダーを右クリックし、**[Save** (保存)] を選択します。
5. [暗号化仕様] 関連リストで、リスト内のレコードを開きます。
6. 暗号化仕様レコードで、[次へ] を 3 回選択して、アルゴリズム定義、ライフサイクル定義、およびキー作成元のセクションを完了します。
これらのセクションのどのフィールドも変更する必要はありません。
7. [キーの生成] フィールドで、[キーの生成] リンクを選択してキーを生成します。
8. 移動先 すべて > システムセキュリティ > フィールド暗号化 > 暗号化フィールドの構成.
9. 暗号化フィールドの構成リストで、[新規] を選択します。
10. フォームのフィールドに入力します。

フィールド	値
タイプ	暗号化する内容に応じて、[列] または [添付ファイル] を選択します。
暗号化されたモジュール	前のステップで作成した暗号化モジュールを選択します。
テーブル	暗号化するデータを含むテーブルを選択します。
メソッド	`単一モジュール`を選択
列	暗号化するテーブルの列 (フィールド) を選択します。

注: 暗号化するデータは、[テーブル] フィールドと [列] フィールドによって決まります。これらのフィールドは、エッジ暗号化を現在使用しているテーブルと列 (フィールド) である必要があります。

11. **[Submit** (送信)] を選択します。
12. 移動先 すべて > キー管理 > モジュールアクセスポリシー > すべて.
13. **[New** (新規)] を選択します。
14. フォームのフィールドに入力します。

フィールド	値
ポリシー名	わかりやすい名前を選択
暗号化されたモジュール	前のステップで作成した暗号化モジュールを選択します。
タイプ	`ロール`を選択

フィールド	値
ターゲットロール	使用するロールを選択します。このロールは、列のデータを暗号化および復号化できる必要があります。
結果	目的のアクションを選択します。

15. **[Submit** (送信)] を選択します。
16. 構成を確認するには、 フィールド暗号化 で暗号化するテーブルに移動し、レコードを開きます。

    たとえば、ユーザー [sys_user] テーブルにアクセスするには、ナビゲーションフィルターに「sys_user.list 」と入力します。

    これまでの手順で暗号化対象として選択したフィールドに、フィールドラベルの横にロックアイコンが表示されるようになりました。

結果
エッジ暗号化フィールドをフィールド暗号化に移行する準備ができました。さらにフィールドを設定するには、フィールドごとに前の手順を繰り返します。

ユーザーが暗号化されていないファイルを添付できないようにする

com.glide.encryption.enable_attachment_key_uii プロパティを変更すると、暗号化モジュールキーにアクセスできるユーザーが、暗号化されていない添付ファイルを添付できなくなります。

始める前に
必要なロール：security_admin

次の手順を実行するには、security_admin ロールに昇格する必要があります。詳細については、「特権ロールへの昇格」を参照してください

デフォルトでは、暗号化モジュールキーにアクセスできるユーザーは、暗号化されていない添付ファイルをアップロードできます。この動作を変更するには、*com.glide.encryption.enable_attachment_key_ui* システムプロパティを使用します。

ファイル添付時、マルチモジュール暗号化フィールド構成を持つレコードで UI ピッカーが表示されます。このプロパティを false に設定すると、添付ファイルを暗号化しないオプションがユーザーに表示されなくなります。

手順

1. 移動先 すべて > システムプロパティ > すべてのプロパティ.
2. [システムプロパティ] リストで、当該のシステムプロパティを検索して開きます。
3. プロパティの値を false に設定します。

フィールド暗号化 の使用

フィールド暗号化を使用して、インスタンス上の暗号化データへのアクセスを管理します。

フィールドデータを暗号化するには、次の 2 つの方法があります。

- 単一モジュール:単一の暗号化モジュールを確定的な方法で使用してデータ暗号化を許可します。
- マルチモジュール:非決定的な方法で複数の暗号化モジュールを使用したデータ暗号化を許可します。行条件は、フィールドに複数のモジュールを適用するための新しい推奨される方法です。行条件は、マルチモジュール機能を確定的な方法で適用します。

関連リンクを使用して、一般的な フィールド暗号化 タスクに関する情報を検索します。

関連トピック

- の暗号化モジュールを作成 フィールド暗号化
- の暗号化仕様の作成 フィールド暗号化
- の詳細アルゴリズムの構成 フィールド暗号化エンタープライズ
- 顧客指定のキーのプロパティを設定する
- フィールドと添付ファイルの暗号化
- フィールド暗号化エンタープライズ の例

の暗号化モジュールを作成 フィールド暗号化

暗号化操作に使用するメカニズムを定義する フィールド暗号化 暗号化モジュールを作成します。

始める前に

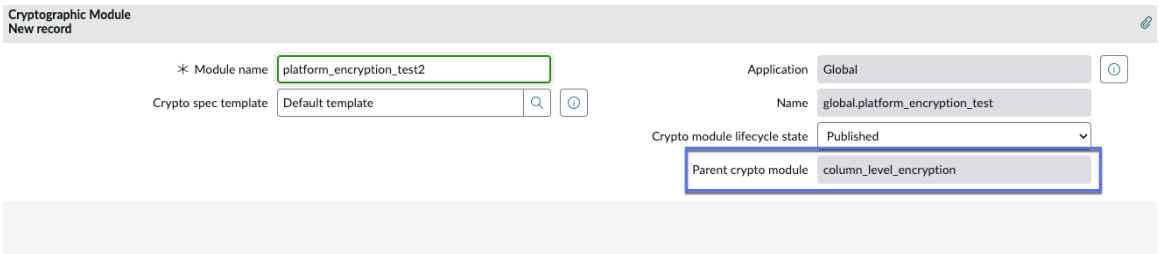
必要なロール：sn_kmf.cryptographic_manager または sn_kmf_admin、security_admin、admin

このタスクについて

この手順では、ベースシステムとの フィールド暗号化 で使用できるオプションと、 フィールド暗号化エンタープライズ 機能で利用可能になる追加の設定オプションについて説明します。 フィールド暗号化エンタープライズ は有料サブスクリプションで利用できます。サポート対象機能と各製品で使用できるオプションについては「暗号化とキー管理のサブスクリプションバンドル」を参照してください。フィールド暗号化エンタープライズ の取得についての詳細は「フィールド暗号化 のアクティブ化」を参照してください。

手順

1. 移動先 すべて > システムセキュリティ > フィールド暗号化モジュール > 新規.



2. フォームのフィールドに入力します。

暗号化モジュールフォーム

フィールド	説明
モジュール名	スクリプトの実行時に参照される英数字の文字列。
暗号化仕様テンプレート	暗号化の仕様および推奨されるアルゴリズムに対し、多くの暗号化の目的のマッピングを含む、暗号化モジュールの作成に使用されるデフォルトのテンプレート。
アプリケーション	選択したアプリケーションスコープ。

フィールド	説明
名前	モジュールの作成時に他のスコープ対象のアプリケーションとの競合を避けるために、暗号化モジュール名の先頭にアプリケーションスコープ名が追加されます。たとえば、グローバルアプリケーションスコープで my_crypto_module という名前のモジュールを作成した場合、その名前は global.my_crypto_module として保存されます。
暗号化モジュールライフサイクル状況	ライフサイクルという用語は、暗号化モジュールの作成、使用、非アクティブ化を指します。設定時に最初は [ドラフト] に設定されます。モジュールを使用する場合は、[公開] に設定します。[デフォルト] テンプレートは自動的に [公開] に設定されます。
親暗号化モジュール	親は **column_level_encryption** として自動的に入力されます。

3. [送信] をクリックします。

   正常に送信すると、暗号化モジュールが暗号化モジュールテーブルにリストされます。

   ⚠ 警告:

   従来の暗号化サポートユーザーの場合：

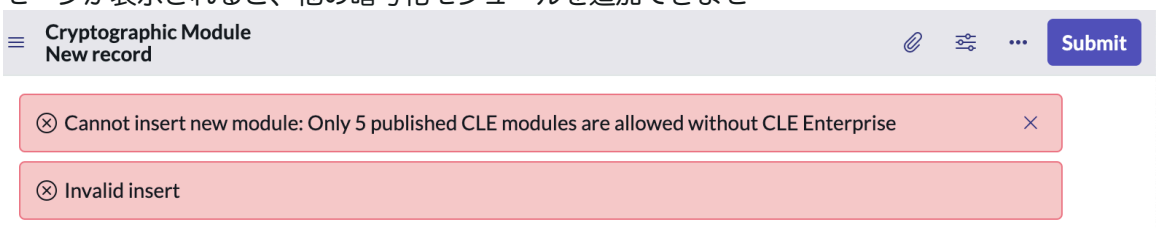
   エンタープライズ以外のバージョンの フィールド暗号化 を使用している場合は、5 つの フィールド に制限されます。この上限を超えると、次の警告が表示されます。

   この挿入は、サブスクリプション製品のエンタイトルメントが付与されているフィールド暗号化の公開フィールド 制限の数を超えています。追加の フィールドには、フィールド暗号化 のエンタープライズサブスクリプションが必要です。アカウントチームにお問い合わせください。

   デフォルトの暗号化仕様は、暗号化の目的が [対称データ暗号化/復号化] に設定され、アルゴリズムが [AES 256 CBC] に設定されて作成されます。更新のアルゴリズムを選択します。

4. 設定オプションを開くには、新しく作成された暗号化モジュールをクリックします。

   ℹ 注: フィールド暗号化エンタープライズ にアップグレードする前に、最大 5 つの フィールド暗号化 フィールド を設定できます。エラーメッセージが表示されると、他の暗号化モジュールを追加できません。



次のタスク

の暗号化仕様の作成 フィールド暗号化。

複数暗号化モジュールの使用

複数の暗号化モジュールでは、複数の暗号化モジュールでデータを暗号化できます。たとえば、各モジュールにロールに基づく独自のアクセスポリシーがある場合、異なるロールを持つユーザーは同じテーブル上のデータを暗号化できますが、互いの暗号化されたデータを表示できないようにするために使用されます。

プラットフォームで複数の暗号化モジュールを使用してデータを暗号化するには、次の 2 つの方法があります。

- 行条件 - アドミニストレーターは、使用する暗号化暗号化モジュールを定義できます。このオプションは確定的なアプローチを提供し、1 つの列に複数の暗号化フィールド構成 (EFC) を許可します。特定のフィールドに個別の暗号化キーを設定することで、フィールドまたは添付ファイルに対して異なるレコードを暗号化できます。これは、複数の暗号化フィールド構成 (EFC) を割り当てることで実現されます。行条件は、複数のモジュールを使用してデータを暗号化する場合に適した方法です。「行条件を使用したデータの暗号化」を参照してください。
- 複数モジュール:ユーザーがデータの暗号化に使用する暗号化モジュールを使用できるようにします。複数のモジュールは、ユーザーがデータを暗号化する正しい暗号化モジュールを選択することに依存するため、この機能は非決定的であり、機能がまだ使用可能であっても、レガシーと見なされ、推奨されません。「複数モジュール機能を使用してデータを暗号化する」を参照してください。

行条件を使用したデータの暗号化

複数のフィールド暗号化モジュールでフィールドを暗号化し、行条件を使用して暗号化するデータと関連する暗号化キーを定義します。行条件を使用して、条件ビルダーを使用してアクセス権を持つユーザーを定義することもできます。

始める前に

必要なロール：sn_kmf.cryptographic_manager または sn_kmf.admin

このタスクについて

注: 行条件を使用する場合は、次のガイドラインが適用されます。

- 行条件は、暗号化された列と添付ファイルでのみサポートされます。
- 複数の暗号化モジュールメソッドを使用する場合は、一括暗号化を使用できません。
- 複数の暗号化モジュールを使用してフィールドを変更し、単一の暗号化モジュールを使用することはできません。
- 行条件を使用すると、メインレコードでデータを暗号化できます。条件ビルダーではドット連結はサポートされていません。
- 行条件は、次のサービスカタログテーブルではサポートされていません。
  - オプション [sc_item_option]
  - 質問と回答 [question_answer]
  - 複数行の質問と回答 [sc_multi_row_question_answer]

このフィールドは、最初にデータを入力するユーザーの暗号化モジュールで暗号化されます。暗号化モジュールはレコードごとに設定されるため、リストのフィールドにはさまざまな暗号化モジュールが含まれる可能性があります。1 つのレコード内では、フィールドは 1 つのモジュールでのみ暗号化できます。

手順

1. 複数の暗号化モジュールとそれぞれのアクセスポリシーを作成します。

   アクセスポリシーを通じて、さまざまな暗号化モジュールにさまざまなロールを付与するようにします。

2. 移動先 システムセキュリティ > フィールド暗号化 > 暗号化フィールドの設定 > 新規.

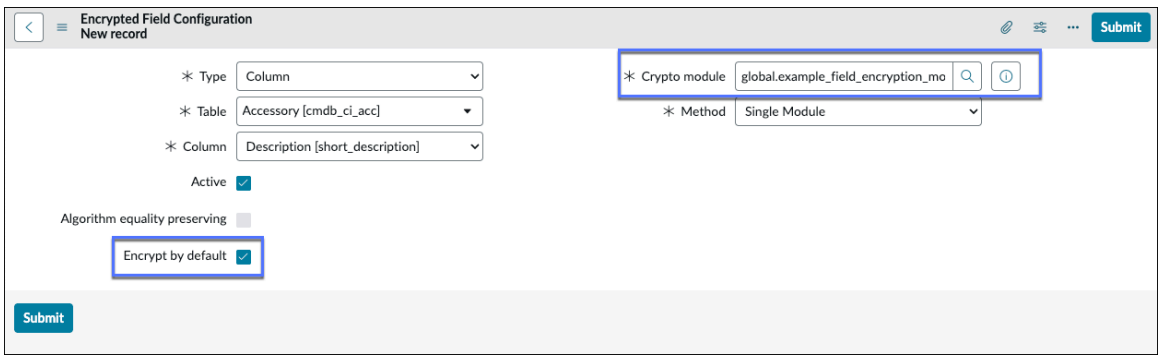
   暗号化フィールドの設定の詳細については、「暗号化フィールド構成を設定する」を参照してください。

3. 暗号化するテーブルの [テーブル] と [列] を選択します。
4. 定義された行の条件を満たさない保存済みレコードを暗号化するには、[デフォルトで暗号化] を選択します。

   このオプションを選択する場合は、デフォルトの暗号化モジュールとして使用する 暗号化モジュール を選択してください。

   このボックスが選択されておらず、追加されたレコードが行の条件を満たさない場合、レコードは暗号化されません。

   たとえば、レコードを暗号化する行条件を定義して、レコードの「部門」フィールドが「IT」と等しいことを宣言できます。その場合、[部門] フィールドが [IT] と等しくないレコードは、[デフォルトで暗号化] が選択されていない限り暗号化されません。



5. **[Submit** (送信)] を選択します。
6. 条件ビルダーに移動します すべて > システムセキュリティ > フィールド暗号化 > 暗号化フィールドの設定 をクリックし、新しい暗号化フィールド構成 (EFC) を選択します。
7. **[New** (新規)] を選択します。
8. 必須フィールドに入力して、行条件を定義します。
9. **[Submit** (送信)] を選択します。

結果

注: 必要に応じてこのプロセスを繰り返し、行条件の数が必要な暗号化暗号化モジュール (ECM) を満たしていることを確認します。ECM セットアップで [デフォルトで暗号化] を選択した場合は、行条件も設定して、選択されているデフォルトの ECM を定義する必要があります。

指定されたフィールドの新しく作成されたデータは、関連するモジュールのキーで暗号化されます。モジュール A のアクセスポリシーで指定されたロールを持つユーザーが、指定されたテーブルに書き込むと、データはモジュール A のキーで暗号化されます。同じロールを持つユーザーのみがデータを読み込むことができます。

Example:
この例では、インシデントテーブルの [簡単な説明] フィールドを暗号化する方法について説明します。この例は、テーブル内の別のフィールドを暗号化する場合にも同様に機能します。
[簡単な説明] を暗号化するには、次の操作を行います。

1. 2 つのフィールド暗号化モジュール A と B を作成します。
2. モジュールごとにモジュールアクセスポリシー (MAP) を作成し、次のようにアクセスを定義します。
   a. モジュール A – HR ロールを持つユーザー向け。
   b. モジュール B – 営業ロールを持つユーザー向け
3. 暗号化フィールド構成 (EFC) レコードを作成します。
   a. 移動先 システムセキュリティ > フィールド暗号化 > 暗号化フィールドの設定 > 新規.
   b. [テーブル] フィールドで [インシデント] を選択します。
   c. [列] フィールドで [簡単な説明] を選択します。
   d. 条件ビルダーの基準に当てはまらないレコードが、デフォルトのフィールド暗号化モジュールで暗号化されていることを検証する必要がある場合は、[デフォルトで暗号化] ボックスを選択します。このオプションを選択しないと、条件ビルダー基準にないレコードは暗号化されません。

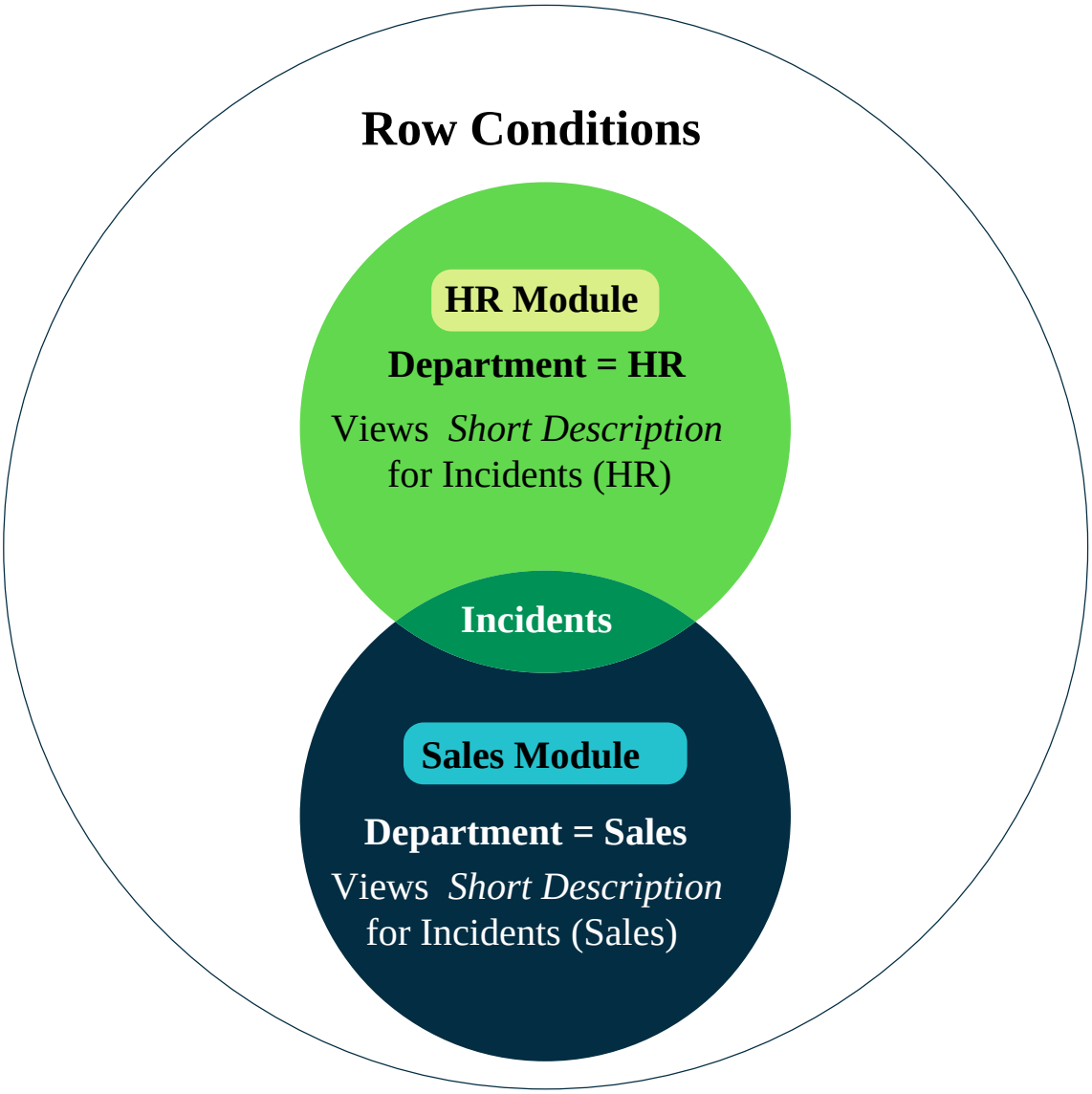
      関連する 暗号化モジュール フィールドにデフォルトの暗号化モジュールを入力します。

   e. **[Submit** (送信)] を選択します。
4. 行条件を作成します。
5. 適切な暗号化ジョブを実行します。
   - 一括暗号化:新しい行条件が作成されたときにこのジョブを実行します。
   - 一括リキー:既存の行条件が変更されたときにこのジョブを実行します。

   「一括暗号化、復号化、およびリキーのジョブをスケジュールする」を参照してください。
6. モジュール A のユーザーとモジュール B のユーザーに、簡単な説明を記載したインシデントを作成します。両方のユーザーにインシデントのリストを表示してもらいます。

HR ロールを持つユーザーが作成したインシデントの簡単な説明は、モジュール A のキーで暗号化されます。営業ロールを持つユーザーが作成したインシデントの簡単な説明は、モジュール B のキーによって暗号化されます。

HR ロールと営業ロールを持つすべてのユーザーに、インシデントへのアクセス権があります。ただし、HR ロールを持つ別のユーザーによって作成されたインシデントの簡単な説明を復号化して表示できるのは、HR ロールを持つユーザーのみです。同様に、営業ロールを持っていたユーザー B が作成したインシデントの簡単な説明を復号化して表示できるのは、営業ロールを持つユーザーのみです。



次のタスク
次のいずれかの操作を実行します。

- 条件フィールドの暗号化を更新する 一括暗号化 ジョブをスケジュールします。
- 既存の行条件を変更する場合は、 一括リキー ジョブを実行して、更新された暗号化モジュールで必要なデータを暗号化します。

「一括暗号化、復号化、およびリキーのジョブをスケジュールする」を参照してください。

複数モジュール機能を使用してデータを暗号化する

複数の暗号化モジュールでデータを暗号化し、暗号化されたデータ内の特定の行に使用するキーをユーザーが判断できるようにします。

始める前に
必要なロール：sn_kmf.cryptographic_manager または sn_kmf.admin

このタスクについて
[複数モジュール] オプションは非決定的と見なされ、特定のレコードに使用するキーをユーザーが決定するため、推奨される方法ではありません。列に複数のモジュールを使用する機能は、行条件に置き換えられます。「複数暗号化モジュールの使用」を参照してください。この非決定論的実装は、最

初に作成され、現在も使用されているため、引き続きサポートされていますが、新しい複数モジュールのユースケースには行条件を使用することをお勧めします。

注: 列の暗号化のみが複数のモジュールをサポートしています。添付ファイルの暗号化は行いません。複数の暗号化モジュールメソッドを使用する場合は、一括暗号化を使用できません。

複数の暗号化モジュールを使用してフィールドを変更し、単一の暗号化モジュールを使用することはできません。

このフィールドは、最初にデータを入力するユーザーの暗号化モジュールで暗号化されます。暗号化モジュールはレコードごとに設定されるため、リストのフィールドにはさまざまな暗号化モジュールが含まれる可能性があります。1 つのレコード内では、フィールドは 1 つのモジュールでのみ暗号化できます。

手順

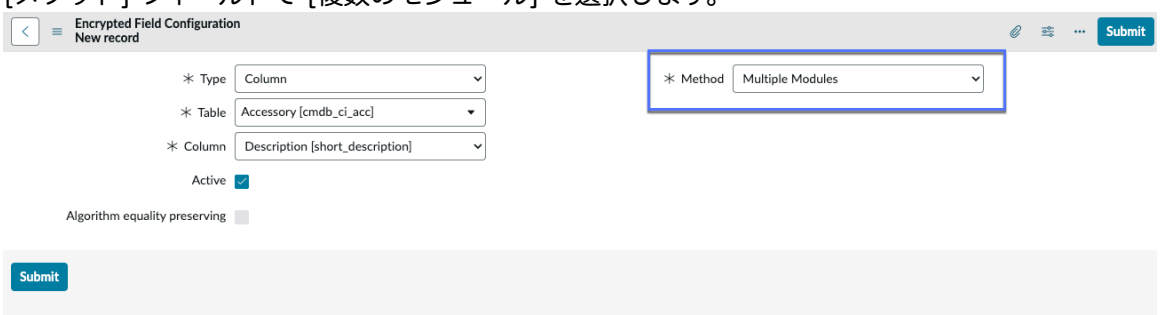
1. 複数のフィールド暗号化モジュールと、それぞれにモジュールアクセスポリシー (MAP) を作成します。

   アクセスポリシーを通じて、さまざまな暗号化モジュールにさまざまなロールを付与するようにします。

2. 移動先 システムセキュリティ > フィールド暗号化 > 暗号化フィールドの設定 > 新規.

   暗号化フィールドの設定の詳細については、「暗号化フィールド構成を設定する」を参照してください。

3. [タイプ] フィールドで、[列] を選択する必要があります。
   添付ファイルの暗号化は複数のモジュールをサポートしていません。
4. [メソッド] フィールドで [複数のモジュール] を選択します。



5. 暗号化するテーブルの [テーブル] と [列] を選択します。
6. **[Submit** (送信)] を選択します。

結果
指定されたフィールドの新しく作成されたデータは、関連するモジュールのキーで暗号化されます。モジュール A のアクセスポリシーで指定されたロールを持つユーザーが、指定されたテーブルに書き込むと、データはモジュール A のキーで暗号化されます。同じロールを持つユーザーのみがデータを読み込むことができます。

Example:
インシデントテーブルの [簡単な説明] 列を暗号化します。次の操作を行います。

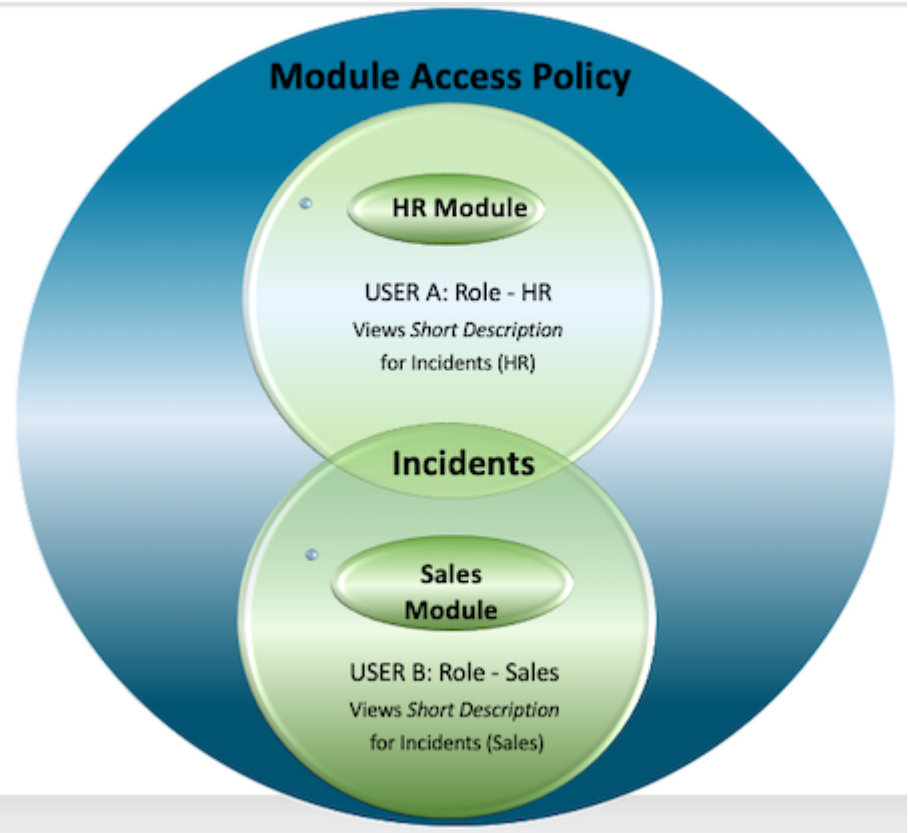
1. 2 つの暗号化モジュール A と B を作成します。
2. 各モジュールに、モジュールアクセスポリシーを作成します。

   モジュール A の場合、人事ロールアクセス権をユーザーに付与します。モジュール B の場合、営業ロールアクセス権をユーザーに付与します。

3. インシデントテーブルの [簡単な説明] 列を指定する暗号化フィールドの設定レコードを作成し、[メソッド] フィールドで [複数のモジュール] を選択していることを確認します。
4. 人事ロールのユーザー (ユーザー A) と営業ロールのユーザー (ユーザー B) の 2 人のユーザーに、簡単な説明のあるインシデントを作成させ、両方のユーザーにインシデントのリストを表示させます。

   人事ロールのユーザーが作成したインシデントの簡単な説明は、モジュール A のキーで暗号化されます。同様に、営業ロールのユーザーが作成したインシデントの簡単な説明は、モジュール B のキーで暗号化されます。

   人事ロールと営業ロールを持つすべてのユーザーがインシデントにアクセスできますが、人事ロールを持つユーザー A が作成したインシデントの簡単な説明を復号化して表示できるのは、人事ロールを持つユーザーのみです。同様に、営業ロールを持っていたユーザー B が作成したインシデントの簡単な説明を復号化して表示できるのは、営業ロールを持つユーザーのみです。



の暗号化仕様の作成 フィールド暗号化

暗号化モジュールを作成してから、対応する暗号化仕様にアクセスし、アルゴリズムを定義します。

始める前に

必要なロール：sn_kmf.cryptographic_manager または sn_kmf_admin および security_admin または admin

このタスクについて

この手順では、ベースシステムとの フィールド暗号化 で使用できるオプションと、 フィールド暗号化エンタープライズ 機能で利用可能になる追加の設定オプションについて説明します。 フィールド暗号化エンタープライズ 機能は有料サブスクリプションで利用できます。サポート対象機能と各製品で使用できるオプションについては「暗号化とキー管理のサブスクリプションバンドル」を参照してください。フィールド暗号化エンタープライズ の取得についての詳細は「フィールド暗号化 のアクティブ化」を参照してください。

フィールド暗号化エンタープライズの暗号化モジュールを作成すると、暗号化仕様が作成されます。

手順

1. 移動先 システムセキュリティ > フィールド暗号化モジュール > すべて.
2. 暗号化モジュールを選択して設定オプションを開きます。
   暗号化モジュール情報が画面の上部に表示されます。対称データの暗号化/復号化の暗号化仕様は、AES 256 CBC アルゴリズムで自動的に作成されます。
3. テーブルから暗号化仕様を選択して、アルゴリズム定義を開きます。
   フィールド暗号化エンタープライズ については、「の詳細アルゴリズムの構成 フィールド暗号化エンタープライズ」を参照してください。
4. [次へ] をクリックしてキーライフサイクルにアクセスします。

次のタスク

次のいずれかの操作を実行します。

- キーライフサイクルテーブルのエントリを選択して、キーライフサイクルの動作を定義します。
  キーライフサイクル定義の完了に関する詳細については、「キーライフサイクル状況の設定」を参照してください。
- [次へ] をクリックして暗号化キーを作成します。このプロセスの詳細については、「ServiceNow 暗号化キーを生成する」を参照してください。

の詳細アルゴリズムの構成 フィールド暗号化エンタープライズ

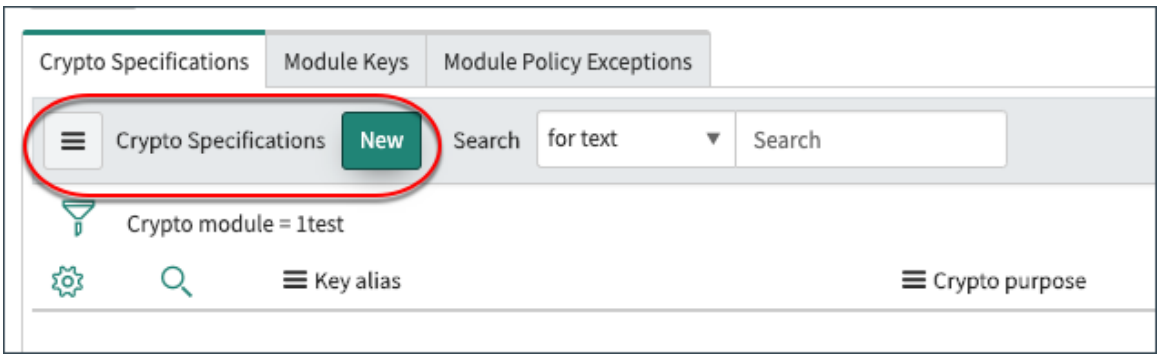
暗号化モジュールのアルゴリズムを定義する暗号化仕様を作成します。フィールド暗号化エンタープライズ で利用可能な詳細オプションを使用して、暗号化の仕様をカスタマイズします。

始める前に

必要なロール：admin

手順

1. [暗号化仕様 **(#)**] タブで、[新規] をクリックします。



2. フォームのフィールドに入力します。

アルゴリズム定義フォーム

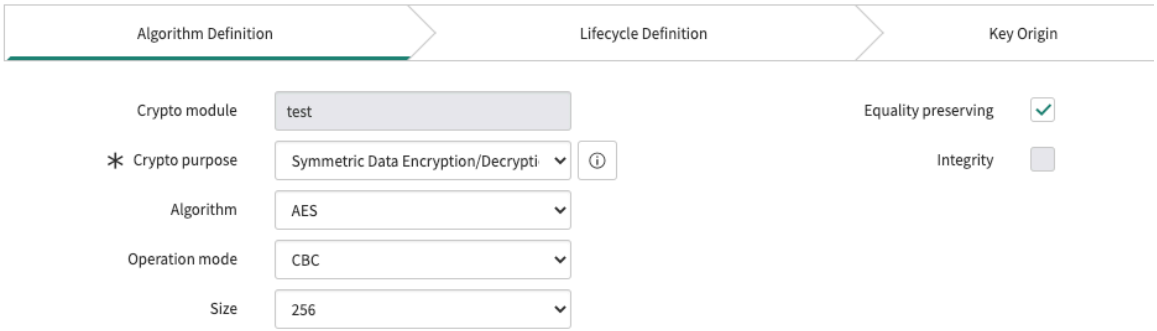
フィールド	説明
暗号化モジュール	選択した暗号化モジュールの名前が入力されます。

フィールド	説明
暗号化の目的	フィールド暗号化エンタープライズ の場合、値は対象データの暗号化/復号化です
アルゴリズム	フィールド暗号化エンタープライズ の場合、値は **AES** です。
操作モード	フィールド暗号化エンタープライズ の場合、値は **CBC** です。
サイズ	可能な値は **256** と **128**です。 注: 256 ビットサイズが暗号化に最も安全で、フィールド暗号化エンタープライズ の対称データの暗号化/復号化に使用されます。
等価性保存	決定的暗号化を有効にするオプション。 注: このオプションを選択すると、フィールドの値が変わらなければフィールド暗号化された値も同じになるはずです。 暗号ブロックチェーン **(CBC)** モードで **AES** を使用した対称データの暗号化/復号化を有効にするオプション。
完全性	GCM 操作で完全性を提供するオプションであり、フィールド暗号化エンタープライズ 機能には適用されません。

3. [送信] をクリックします。

次の例は、AES CBC-256 暗号化を示しています。フィールド暗号化エンタープライズ がアクティブで、親モジュールが column_level_encryption の場合、対称データ暗号化/復号化 AES

CBC-256 のみが暗号化の目的として適用されます。詳細については、「暗号化仕様の概要」を参照してください。



次のタスク

次のいずれかの操作を実行します。

- キーライフサイクルテーブルのエントリを選択して、キーライフサイクルの動作を定義します。キーライフサイクル定義の完了に関する詳細については、「キーライフサイクル状況の設定」を参照してください。
- [次へ] を選択して暗号化キーを作成します。キーの生成については、次のいずれかのタスクを参照してください。
  - ServiceNow 暗号化キーを生成する。
  - 顧客指定のキーのプロパティを設定する。
  - ラッピング/ラップ解除キーペアのインポート。

での顧客指定のキーの使用 フィールド暗号化エンタープライズ

ServiceNow® システム生成キーを使用する代わりに、顧客が指定した独自のキーを使用できます。

重要: これらのトピックは、`com.glide.now.platform.encryption` プラグインでのみ利用可能な フィールド暗号化エンタープライズ を使用するインスタンスのみが対象になります。このプラグインの取得についての詳細は「フィールド暗号化 のアクティブ化」を参照してください。

フィールド暗号化エンタープライズ® では、暗号化に独自のキーを使用できます。アドミニストレーターは、ServiceNow® 指定のキーまたは独自の顧客指定のキー (CSK) を ServiceNow AI Platform の暗号化に使用できます。

重要: 顧客指定のキーオプションを使用するには、独自の暗号化キーが必要です。

キーを取得したら、次の手順に従ってインスタンスで使用できます。

1. 顧客指定のキーのプロパティを設定する

   ラッピング RSA キーペアのサイズ、パディングアルゴリズム、および有効期間は、3 つのシステムプロパティによって定義されます。これらのプロパティを確認し、デフォルトがニーズに合わない場合は値を調整します。

2. 顧客指定のキーをラップする

   OpenSSL のような暗号化ツールを使用してキーをラップし、ダウンロードした公開鍵で暗号化に使用する対称キーをラップします。

顧客指定のキーの構成とアップロード

ラップされた顧客指定のキーをアップロードし、インスタンスでの暗号化にキーの使用を開始するように暗号化モジュールを設定します。

顧客指定のキーのプロパティを設定する

フィールド暗号化エンタープライズ プラグインが有効になっている場合は、システムプロパティを使用して、顧客指定のキーのキーパディング、短期キーペアのサイズ、およびキーの有効期間を定義できます。

フィールド暗号化エンタープライズ をキー管理とともに使用すると、データ暗号化キーのキーライフサイクル全体を管理できます。必要に応じて、環境内で生成されたデータ暗号化キーを安全に交換できます。

Platform Encryption をキー管理とともに使用すると、データ暗号化キーのキーライフサイクル全体を管理できます。必要に応じて、環境内で生成されたデータ暗号化キーを安全に交換できます。

キーペア属性を定義するためのシステムプロパティ

独自のキーを指定する場合は、RSA 公開キーでラップする必要があります。ラッピング RSA キーペアのサイズ、パディングアルゴリズム、および有効期間は、次の 3 つのプロパティで定義されます。

- `glide.kmf.ephemeral_key.key_padding` は、短期キーのキーパディングスキームを制御します。デフォルトのスキームは OAEP SHA256 ですが、SHA1 もサポートされています。
- `glide.kmf.ephemeral_key.key_size` は、短期キーペアのキーサイズを制御します。デフォルトは 4096 ビットですが、2048 ビットもサポートされています。
- `glide.kmf.ephemeral_key.key_validity_period` は、短期キーペアの有効期間を定義します。デフォルト値は 2 時間です。

データ暗号化キーがインスタンスにインポートされた後、セキュアなラッピングキーによってインスタンスの新しいモジュールキーが保護されます。ラッピングキーは、SafeNet KeySecure のハードウェアセキュリティモジュール (HSM) で生成されたインスタンスキー暗号化キーです。キータイプの詳細については、「キー管理フレームワークのインスタンスレベルのキー」を参照してください。

顧客指定のキーをラップする に続きます。

顧客指定のキーをラップする

ダウンロードした公開鍵で、暗号化に使用する対称キーをラップします。

始める前に

注: この手順では、KMF ベースシステムで使用できるオプションと フィールド暗号化エンタープライズ 機能で使用するオプションについて説明します。フィールド暗号化エンタープライズ 機能は、`com.glide.now.platform.encryption` プラグインがアクティブな場合にのみ使用できます。 フィールド暗号化エンタープライズ の取得についての詳細は「フィールド暗号化 のアクティブ化」を参照してください。

このドキュメントの一部の手順では、ローカルデバイスにインストールされている暗号化ツールを使用する必要があります。このタスクの例では、OpenSSL ツールを使用します。このツールの詳細については、「 https://www.openssl.org 」を参照してください。LibreSSL や GnuTLS などの他の暗号化ツールを使用している場合は、それらの製品のドキュメントで類似した手順について参照してください。

- キーのサイズ、パディングアルゴリズム、および有効期間を制御するオプションのプロパティを変更します。「顧客指定のキーのプロパティを設定する」を参照してください。
- 暗号化には対称キー (.BIN) が必要です。

重要: キーはバイナリ形式である必要があります。別の形式が使用されている場合、「トークンの検証に失敗しました。変更されていないトークンを再添付してください。」というエラーメッセージが表示されます。

- キーをラップするには、暗号化ツールが必要です。この例では、OpenSSL 1.1 を使用します。

必要なロール：sn_kmf.cryptographic_manager または sn_kmf.admin

手順

1. 移動先 すべて > キー管理フレームワーク > 暗号化モジュール > すべて.
2. 暗号化仕様関連リストから、顧客指定のキーに対して作成した暗号化モジュールを選択します。
3. キーの作成手順に移動します。
4. 以前にラッピングキーをダウンロードしていない場合は、リンクをクリックして *token_publickey<id>.zip* ファイルをダウンロードし、キーと同じ場所に保存します。

   注: ダウンロードした *token_publickey<id>* ファイルの名前は変更しないでください。

5. ローカルネットワークにファイルを展開します。
   zip ファイルには、インポートトークンと公開鍵 .PEM 証明書の 2 つのファイルが含まれています。対称キーを公開鍵でラップして暗号化します。
6. *token_publickey* ファイルの名前をクリップボードにコピーします。
7. コマンドラインから、コピーした *token_publickey* ファイル名を使用して、ラップされたキーのプレースホルダーとして展開されたファイルのフォルダーを開きます。
8. 例を実際の暗号化ファイルの名前に置き換えて、このスクリプトを編集します。

```
"downloads user.name$ cd token_publickey_<token>
openssl pkeyutl -encrypt -pubin -inkey publickey_<keyname>.PEM
-in <keyname.bin>
-out wrapped_key_material -pkeyopt rsa_padding_mode:oaep -pkeyopt
 rsa_oaep_md:sha<128 or 256> "
```

詳細については、次の表のキーラッピングコマンドを確認してください。

キーラッピングコマンド

方向	コマンド	例
ラッピングトークンをダウンロードしたファイルディレクトリを開きます。	cd	cd token_publickey123456789
publickey.PEM 証明書の名前を貼り付けます。	openssl pkeyutl -encrypt -pubin -inkey	publickey_586798643ffff.PEM
ここにキーの名前を貼り付けます。	-in	mykey.bin
<-out> コマンドを入力し、キーが 128 ビットか 256 ビットかを指定します。	-out wrapped_key_material -pkeyopt	適用外

自動翻訳

方向	コマンド	例
	rsa_padding_mode:oaep -pkeyopt rsa_oaep_md:sha256	

9. コマンドを実行します。
   システムメッセージに token_publickey_<keynumber> が表示されます。キーが生成され、「*wrapper_key_ material*」ファイルがディレクトリに追加されます。
10. ラップされたキーをアップロードします。

次のタスク
ラップされたキーをアップロードするには 顧客指定のキーの構成とアップロード に戻ります。

顧客指定のキーの構成とアップロード
ServiceNow® システム生成キーを使用する代わりに、顧客が指定した独自のキーを使用できます。

始める前に
必要なロール：security_admin、sn_kmf.cryptographic_manager

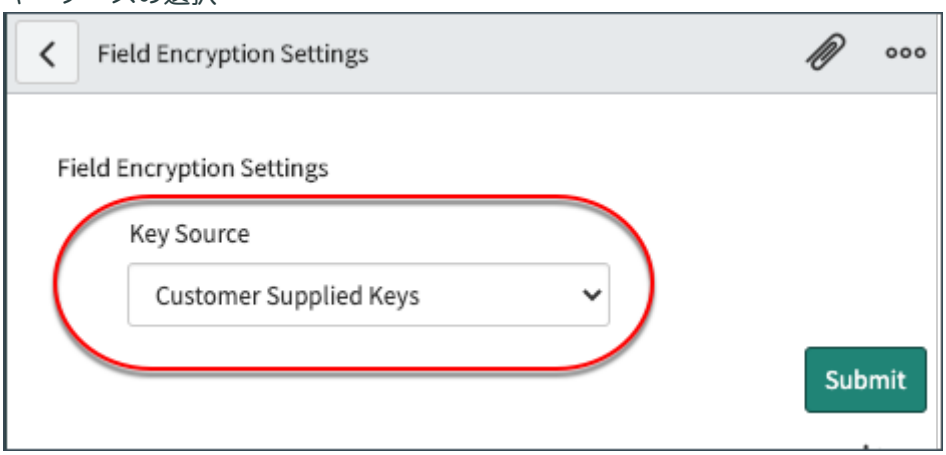
独自のキーを提供していない場合は、この手順を実行する必要はありません。ServiceNow® キーで暗号化モジュールを作成するには、暗号化モジュールを作成する または の暗号化モジュールを作成 フィールド暗号化 に移動します。

注: この手順は、フィールド暗号化エンタープライズ 機能にのみ適用されます。詳細については、「フィールド暗号化 のアクティブ化」を参照してください。

重要: 顧客指定のキーを取り消すことはできません。

手順
1. 移動先 すべて > システムセキュリティ > フィールド暗号化設定 をクリックし、[顧客指定のキー] が選択されていることを確認します。

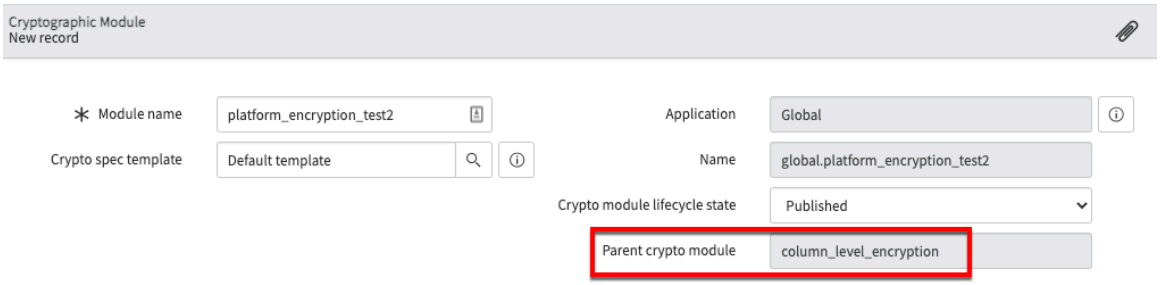
キーソースの選択



2. **[Submit** (送信)] を選択します。
3. 戻る システムセキュリティ > フィールド暗号化モジュール > > 新規作成.

自動翻訳

新しい暗号化モジュールの作成



4. 暗号化モジュールフォームに次のように入力します。

暗号化モジュールのフィールド

フィールド	説明
モジュール名	モジュールの名前を入力します。
暗号化仕様テンプレート	デフォルトの暗号化テンプレートが選択されています。
名前	モジュール名に基づいて自動入力され、どのアプリケーションが適用されているかを確認するためにスコープが名前の先頭に追加されます。この場合、グローバルスコープが適用されます。
暗号化モジュールライフサイクルステータス	選択 公開済み 暗号化モジュールをアクティブ化します。
親暗号化モジュール	顧客指定のキーと暗号化モジュールを使用すると、親モジュール **column_level_encryption** が自動的に選択されます。

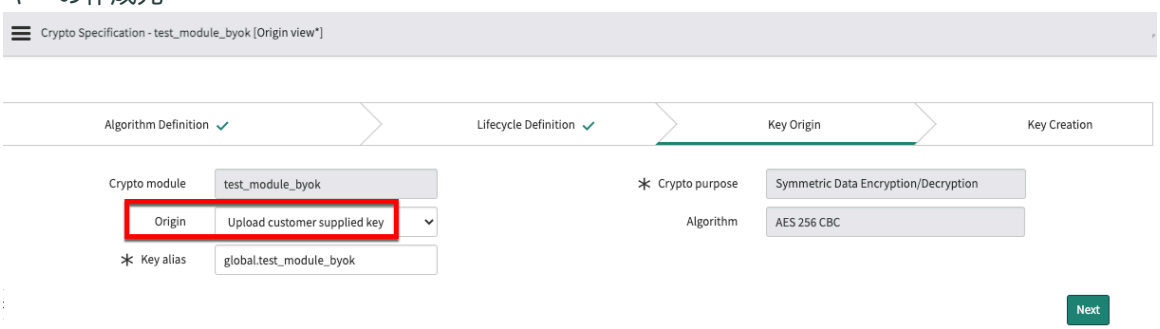
5. [送信] を選択します。

6. テーブルから新しく作成された暗号化モジュールを選択します。
[暗号化仕様] 関連リストで、AES 256 CFB アルゴリズムを使用して自動生成されたキーエイリアスを選択します。

暗号化の目的と フィールド暗号化 アルゴリズムが自動的に入力され、[キーの作成元] ステージにジャンプします。

7. [顧客指定のキーをアップロード] が [作成元] であり、[キーエイリアス] が既に入力されていることに注意してください。

キーの作成元

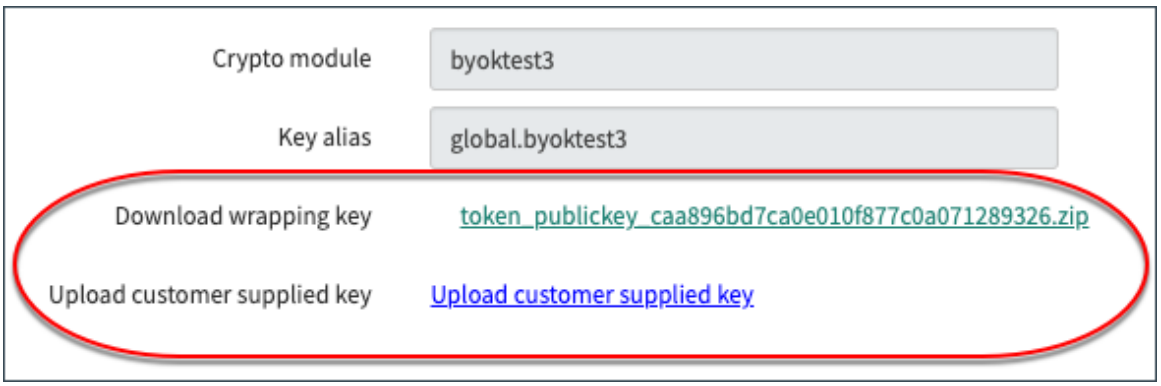


8. [次へ] を選択して [キーの作成] ステージに移動します。

   次の 2 つのリンクがあります。

   - [ラッピングキーをダウンロード] では、インポートトークンと公開鍵証明書 (.PEM ファイル) を含む zip ファイルでキーがダウンロードされます。インポートトークンを使用して、インスタンスのセキュリティ仕様に従ってキーラッピングが成功したことを確認します。公開鍵証明書の .PEM ファイルを使用して、顧客指定のキーを安全にラップしてから、トークンとともにアップロードします。
   - [顧客指定のキーをアップロード] を選択すると、ファイルブラウザーが開き、ラップしたトークンと暗号化キーが選択されます。

   キー作成アップロードリンク

9. [ラッピングキーをダウンロード] を選択してトークンを保存します。
   キーがシステムに保存されているのと同じ場所にトークンを保存します。ダウンロードしたトークンの名前を変更しないでください。
10. 端末で BYOK コマンドを実行してキーをラップします。
    詳細については、「顧客指定のキーをラップする」を参照してください。
11. [顧客指定キーのアップロード] を選択します。
12. [参照] を選択して、ラップされたキーとトークンファイルの 2 つのファイルを選択します。
    [添付ファイル] ウィンドウに 2 つのファイルが表示されます。

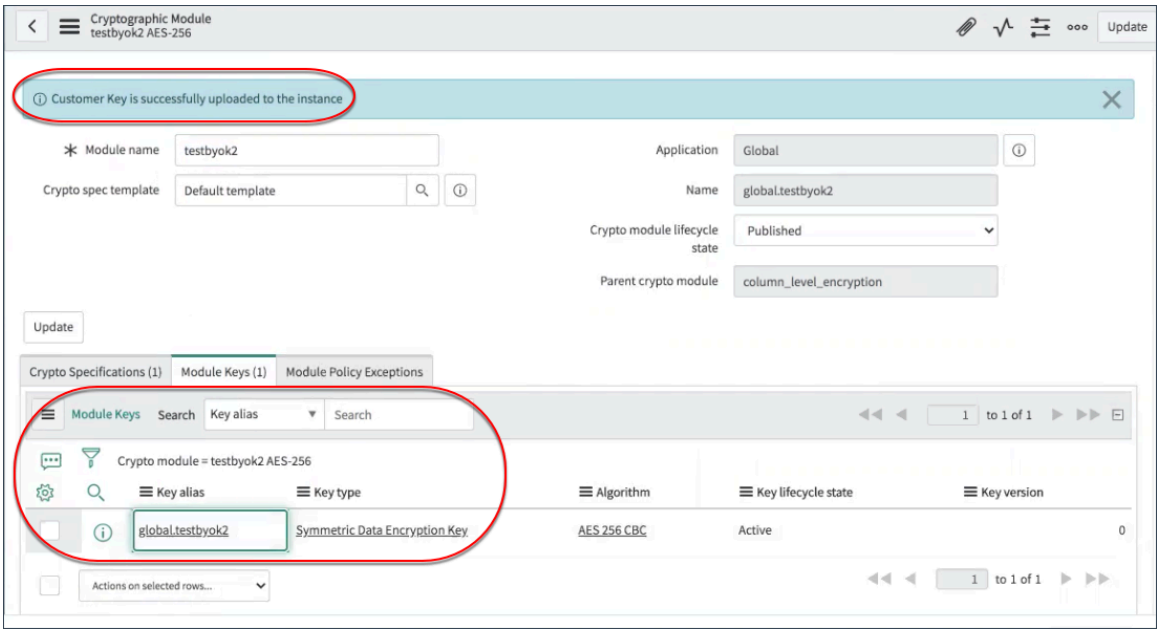
    ラップされたキー添付ファイルのアップロード

    必要に応じて、削除するファイルを選択して再アップロードします。
13. **[OK]** を選択します。

[暗号化モジュール] 画面に戻ります。顧客キーの正常なアップロードを確認するメッセージが表示されます。キーはモジュールキー関連リストにも表示されます。

キーのアップロードの確認



次のタスク
顧客指定のキーを使用した暗号化モジュールの設定が完了したので、「モジュールアクセスポリシーを作成する」に進みます。

フィールドと添付ファイルの暗号化

暗号化モジュールが作成されると、セキュリティアドミニストレーターは暗号化フィールド構成 (EFC) を定義し、テーブルのフィールドまたは添付ファイルを暗号化することを選択できます。

フィールド暗号化方法

注: 暗号化されたフィールドは設計上、監査されません。この動作は設定できません。

1. で生成されたキーまたは顧客指定のキー (自分のキーを使用する) ServiceNow キーソースを指定します。 システムセキュリティ > フィールド暗号化設定.
2. キーソースを指定したら、暗号化モジュールを作成するか、既存の暗号化モジュールを使用します。手順については、「暗号化モジュールを作成する」を参照してください。

   注: 顧客指定のキーを使用する場合は、「の暗号化モジュールを作成 フィールド暗号化」および「顧客指定のキーのプロパティを設定する」の指示に従ってください。

3. 暗号化フィールドの構成を作成して、暗号化が適用される場所を定義します。ここでは、ターゲットテーブルを指定し、列を暗号化するか、テーブル内の添付ファイルを暗号化するかを選択します。「暗号化フィールド構成を設定する」を参照して開始してください。

   注: 顧客指定のキーを使用してフィールドと添付ファイルを暗号化する方法については、「フィールド暗号化エンタープライズ の例」を参照してください。

暗号化フィールド構成を設定する

事前設定された暗号化モジュールを使用して、暗号化するテーブルの列または添付ファイルを設定します。

始める前に
必要なロール：sn_kmf.cryptographic_manager および security_admin、または security admin へのロールの昇格

このタスクについて
そのスコープ内のテーブルを表示できるように、正しいアプリケーションスコープ内にいることを確認してください。

この構成で使用した暗号化モジュールへのアクセス権を持つユーザーのみが、暗号化されたテーブル列のデータを読み取ったり、添付ファイルにアクセスしたりできます。

- ユーザーが書き込みアクセス権を持っているが、読み取りアクセス権がない場合、フィールドは編集モードで表示され、入力されたデータはアスタリスクとして表示されます。
- ユーザーが読み取りアクセス権を持っているが、書き込みアクセス権がない場合、フィールドには復号化されたデータが読み取り専用モードで表示されます。
- ユーザーがすべてのアクセス権を持っている場合、暗号化フィールドで読み取り/書き込み機能を使用できます。

開始するには、「暗号化モジュールを作成する」または「の暗号化モジュールを作成 フィールド暗号化」を参照してください。

重要:

列を暗号化した後、列に挿入された新しいデータはすべて自動的に暗号化されます。ただし、暗号化がアクティブになる前に列に存在していたデータは、自動的に暗号化されません。

列が暗号化される前に存在していたデータを暗号化するには、別途一括暗号化ジョブを実行する必要があります。一括暗号化の詳細については、「一括暗号化または復号化を実行する」を参照してください。

手順

1. 移動先 すべて > システムセキュリティ > フィールド暗号化 > 暗号化フィールドの設定 > 新規.
2. **[New** (新規)] を選択します。
3. フォームに入力します。

フィールド	説明
タイプ	テーブル列を暗号化する列、またはテーブルのすべての添付ファイルを暗号化する添付ファイル。 暗号化されるデータのタイプは次のとおりです。 ○ 文字列テキスト (完全な UTF-8) ○ 添付ファイル ○ 日付、日付/時間： 注: 暗号化フィールドの設定を作成して、既存の [日付] および [日付/時刻] フィールドを暗号化することができます。新しい暗号化構成は、親テーブルのみに追加できます。新しい暗号化構成は、子テーブルに追加できません。 ○ URL

フィールド	説明
	○ HTML ○ ジャーナル ○ 翻訳済み
テーブル	フィールドまたは添付ファイルを暗号化するテーブルを選択します。
列	タイプとして列を選択した場合は、暗号化する列 (フィールド)。
有効	構成をアクティブとしてマークする場合に選択します。構成がまだ使用されていない場合は選択を解除します。
暗号化モジュール	暗号化フィールド構成が適用される暗号化モジュール。
メソッド	1 つのモジュール全体のフィールド構成を設定するには、単一モジュールを選択します。複数の暗号化モジュールにまたがるロールベースのアクセスの場合は、複数のモジュールを選択します。 単一モジュール 単一モジュールを使用してすべての添付ファイルを暗号化するには、このオプションを使用します。ユーザーにはこのモジュールへのアクセス権が必要です。そうでないと添付ファイルをアップロードできません。 複数モジュール 添付ファイルのアップロード時にユーザーがモジュールを選択できるようにするには、このオプションを使用します。1 つ以上のモジュールへのアクセス権を持つユーザーが、暗号化に使用するモジュールを選択できるようになります。モジュールへのアクセス権のないユーザーは、暗号化されていない添付ファイルをアップロードできます。
アルゴリズム暗号化保存 [読み取り専用]	選択した暗号化モジュールが非決定的暗号化をサポートするように既に設定されているかどうかを示します。これは、同じデータが複数回暗号化されている場合、それぞれの暗号化が毎回異なることを意味します。

4. [送信] を選択します。

暗号化モジュールのスクリプトアクセス

スクリプトを実行して、暗号化の目的のために暗号化モジュールポリシーにアクセスできます。

キー管理フレームワーク の場合、ポリシーはスクリプトに基づくことができます。スクリプトアクセスに対してアクセスポリシーがトリガーされると、バックエンドスクリプトはスクリプトからモジュールポリシーアクションを実行できます。

暗号化モジュールは、非対称データ復号化や対称データ復号化など、1 つ以上の暗号化の目的をサポートできます。暗号化の目的ごとに、暗号化キーと定義された暗号化の目的を生成する必要があります。

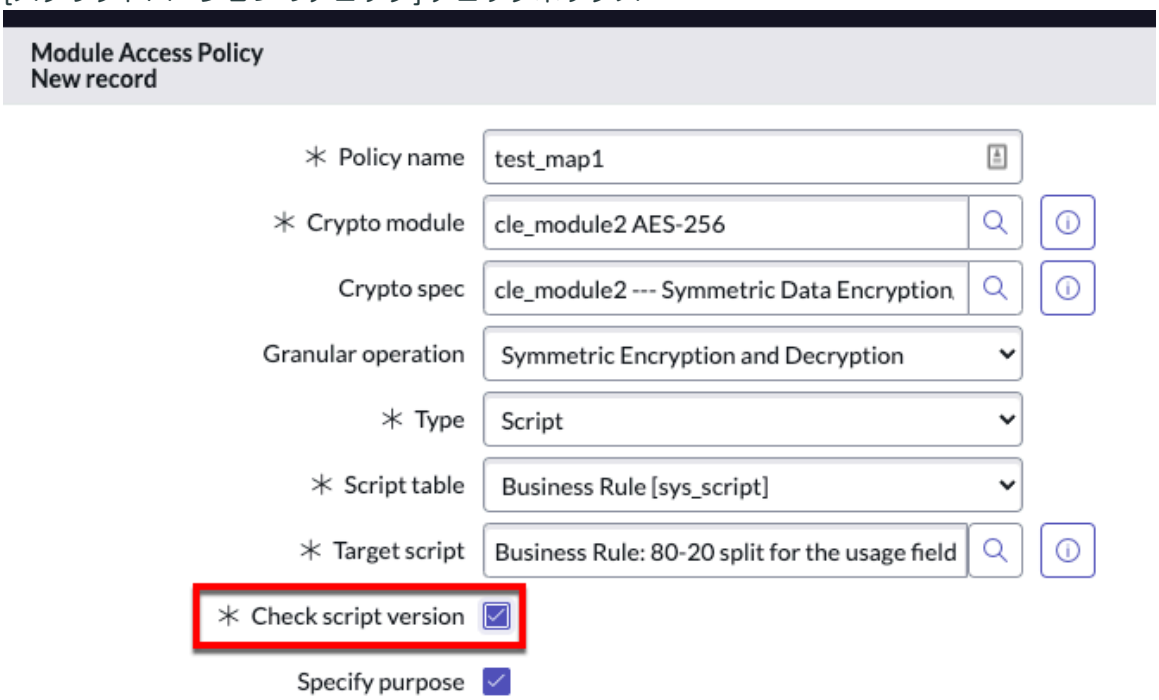
暗号化スクリプト要求を実行するときは、次の点を考慮してください。

- 参照される暗号化の目的は、暗号化モジュールで定義する必要があります。
- 暗号化モジュールに対して生成されたアクティブなキーが存在する必要があります。
- モジュールアクセスポリシータイプはスクリプトに設定する必要があります。

スクリプトバージョンのチェック

スクリプトタイプに設定されたモジュールアクセスポリシーを作成するときに、アクセスするスクリプトバージョンの整合性を検証するオプションがあります。割り当てられたバージョンのスクリプトのみが暗号化モジュールへのアクセスを許可されます。モジュールアクセスポリシーで [スクリプトバージョンのチェック] チェックボックスをオンにすると、スクリプトが実行されるたびにバージョンの比較が実行されます。スクリプトが変更された場合は、ユーザーに通知されます。

[スクリプトバージョンのチェック] チェックボックス



暗号化されたデータへのスクリプトアクセスを設定する

スクリプトを実行して、暗号化の目的のために暗号化モジュールポリシーを実行します。特定の読み取り (復号化/ラップ解除) または書き込み (暗号化/ラップ) アクセスは、モジュールアクセスポリシー操作の粒度に基づいて定義できます。

始める前に

必要なロール：sn_kmf.cryptographic_manager

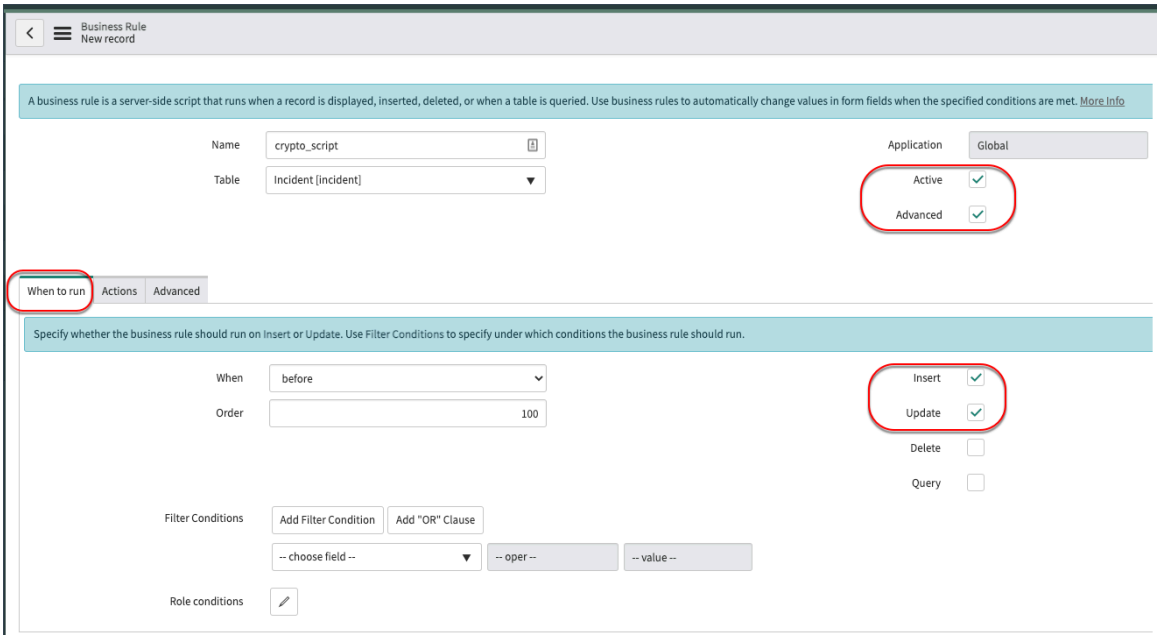
このタスクについて

使用例は、ビジネスルールおよびスクリプトインクルードです。この手順では、ビジネスルールのスクリプトを使用します。

手順

1. 対称データ暗号化/復号化アルゴリズムを使用して暗号化モジュールを作成します。
   詳細については、「暗号化モジュールを作成する」を参照してください。データまたは添付ファイルへの特定のアクセスは、次の特性を持つモジュールアクセスポリシーで制御されます。
   - 対称暗号化：スクリプトはデータを暗号化できますが、復号化することはできません。
   - 対称復号化：スクリプトはアップロードされた暗号化データまたは添付ファイルを復号化できますが、データまたは添付ファイルを暗号化できません。
   - 対称暗号化と復号化：スクリプトはデータまたは添付ファイルの暗号化と復号化の両方を行えます。
2. 移動先 システム定義 > ビジネスルール.

3. [**New**] をクリックします。



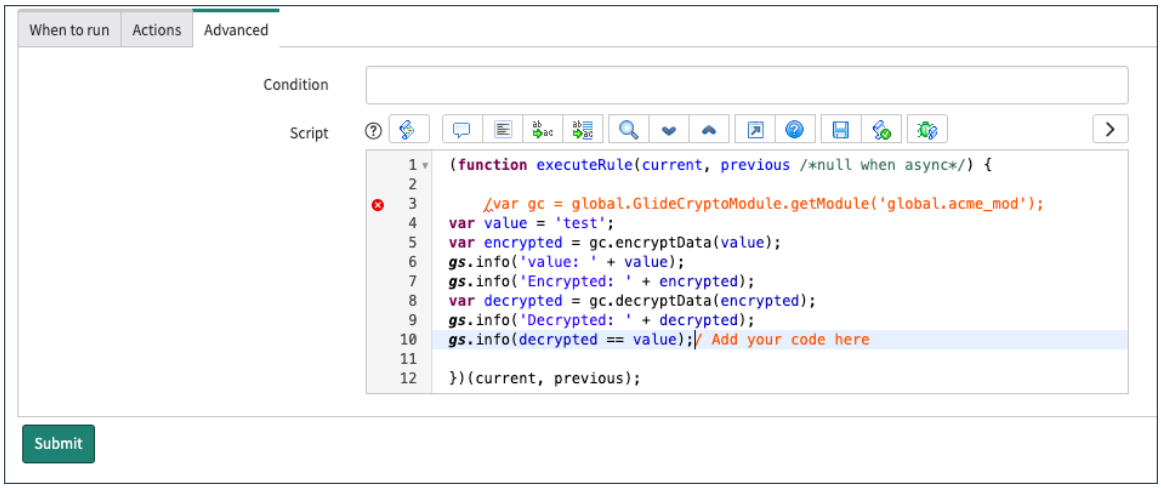
4. [実行タイミング] タブでフォームに入力し、[詳細] タブでスクリプトを入力します。

ビジネスルールのフィールド

フィールド	説明
名前	ビジネスルールの名前を入力します。
テーブル	ドロップダウンリストからインシデント [インシデント] を選択します。
アプリケーション	デフォルトでグローバルが選択されています。
有効	ルールをアクティブとしてマークします。
詳細	このチェックボックスをオンにして、詳細オプションを表示します。
[実行タイミング] タブ	[実行タイミング] タブで、[挿入] および [更新] フィールドを有効にします。

[詳細] タブ	[詳細] タブで、次のスクリプトテキストを 3 行目に貼り付けます。
	`//`
	`var gc =`
	`global.GlideCryptoModule.getModule('global.acme_mod');`
	`var value = 'test';`
	`var encrypted = gc.encryptData(value);`
	`gs.info('value: ' + value);`
	`gs.info('Encrypted: ' + encrypted);`
	`var decrypted = gc.decryptData(encrypted);`
	`gs.info('Decrypted: ' + decrypted);`
	`gs.info(decrypted == value);`
	注: 詳細については、「ビジネスルール詳細タブ」の画像を参照してください。

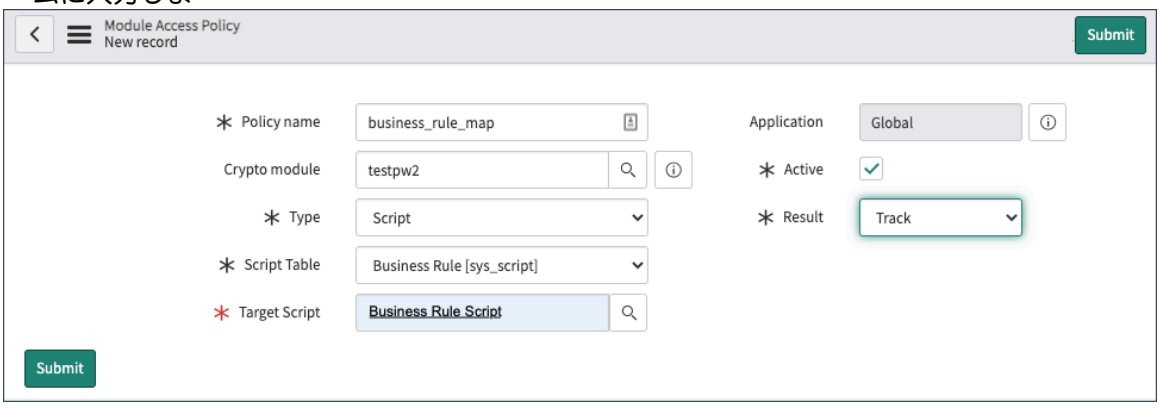
自動翻訳



5. [**Submit** (送信)] を選択します。
6. 移動先 キー管理 > モジュールアクセスポリシー > すべて。

   注: 詳細については、「モジュールアクセスポリシーを作成する」を参照してください。

7. [新規] をクリックします。
8. フォームに入力しま



す。

[モジュールアクセスポリシー] フィールド

フィールド	説明
ポリシー名	ポリシーの名前を入力します。
暗号化モジュール	検索アイコンをクリックして、対称データの暗号化/復号化アルゴリズムを使用するモジュールを選択します。
タイプ	[スクリプト] を選択して、スクリプトでアクセスを制御します。
スクリプトテーブル	スクリプトテーブルのドロップダウンリストから値を選択します。この例では、ビジネスルール **[sys_script]**を選択します。
ターゲットスクリプト	ポリシーのスクリプトドキュメントを選択します。[テーブル名] を選択してから、ポリシーの関連ドキュメントを選択します。この例では、

フィールド	説明
	前のステップで作成したビジネスルールを選択します。
有効	ポリシーを有効にする場合に選択します。
結果	スクリプトにモジュールへのアクセス権を付与するには、[結果] フィールドで [追跡] を選択します。

9. [送信] をクリックします。
   スクリプトのモジュールアクセスポリシーがシステムで利用できるようになりました。

拒否された暗号化モジュール使用要求を表示する

サポートされていない暗号化メカニズムが原因で、スクリプトによる暗号化要求が却下された暗号化モジュールが表示されます。

始める前に
必要なロール：sn_kmf.cryptographic_manager

このタスクについて
暗号化モジュールは、非対称データ復号化や対称データ復号化など、1 つ以上の暗号化の目的をサポートできます。モジュールアクセスポリシーに基づいてのみ、暗号化されたデータにアクセスできます。スクリプトがモジュールで定義されていない目的で暗号化モジュールを使用しようとすると、スクリプトは暗号化されたデータにアクセスできません。

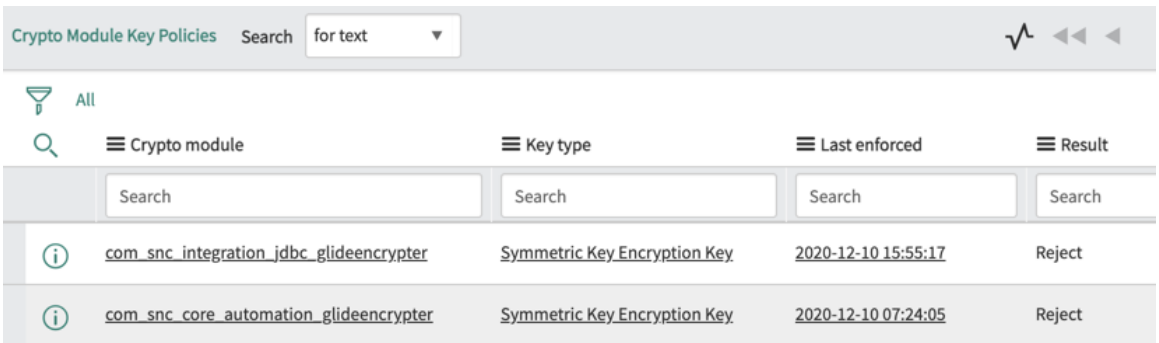
以下の例では、暗号化の目的が暗号化モジュールに割り当てられましたが、そのためのキーは生成されませんでした。

手順

移動先 すべて > キー管理 > モジュールキーポリシー > モジュールキーの拒否.

要求を却下した暗号化モジュールのリストが、対応するスクリプトで使用されている暗号化キーとともに表示されます。

モジュールキーの拒否



注: 異なるスクリプトが同じキータイプを使用して同じ暗号化モジュールを使用しようとすると、前回の強制の値が更新されます。別の行は生成されません。

この例では、モジュール 1 のキーが侵害されたため、2020-02-10_15:55:17 に最初のモジュールが要求を却下しました。2 番目のモジュールのキーが一時停止されているため、2020-02-10_07:24:05 に 2 番目のモジュールが要求を却下しました。

次回実行時に暗号化モジュールを使用する権限をスクリプトに付与するには、スクリプト暗号化のモジュールアクセスポリシーを作成します。詳細については、「暗号化されたデータへのスクリプトアクセスを設定する」を参照してください。

一括暗号化、復号化、およびリキーのジョブをスケジュールする

インスタンスに最適な時間に実行する暗号化、復号化、およびリキーのジョブをスケジュールします。

始める前に

暗号化、復号化、およびリキーのジョブは多くの時間とリソースを必要とするため、ピーク時間外にスケジュールするよう検討してください。また、ジョブをスケジュールしているユーザーが各ジョブに対して適切なアクセス権を持っていることも確認してください。

必要なロール：sn_kmf.cryptographic_manager

このタスクについて

一括暗号化と復号化は、暗号化フィールドの設定フォームからも使用できます。手順については、「一括暗号化または復号化を実行する」を参照してください。

手順

1. 移動先 すべて > システムセキュリティ > セキュリティジョブ.
2. **[New]** をクリックします。
3. スケジューリングフォームに入力します。

フィールド	説明
名前	暗号化、復号化、またはリキージョブの名前
タイプ	ジョブタイプ： ○ モジュールへのキー移行コンテキスト：暗号化モジュールへと暗号化コンテキストキーを一括で移行します (暗号化モジュールでアクセス制御するためのモジュールアクセスポリシーレコードの作成を含む)。 ○ モジュールへのデータ移行コンテキスト：暗号化コンテキストによって暗号化されたデータを暗号化モジュールに移行します。 ○ 一括復号化添付ファイル：[テーブル] フィールドで定義した単一のテーブルのレコード内のすべての暗号化された添付ファイルを復号化します。 ○ 一括暗号化添付ファイル：[テーブル] フィールドで定義した単一のテーブルのレコード内のすべての添付ファイルを暗号化します。 ○ 一括暗号化:フィールド暗号化設定で使用される、定義された列/フィールドの既存の値を暗号化します ○ 一括復号化モジュール：単一モジュールのフィールド暗号化設定で使用されている、定義された列/フィールドの既存の値を復号化します。 ○ 一括復号化マルチモジュール：複数モジュールのフィールド暗号化設定で使用されている、定義された列/フィールドの既存の値を復号化します。 ○ 一括リキー:モジュールの現在アクティブなキーを使用して、フィールド暗号化設定で使用されている定義された列/フィールドの既存の値を再暗号化します。 ○ 添付ファイルコンテキストのモジュールへの移行：フィールド暗号化設定で定義されたテーブルの既存の添付ファイルを暗号化します。以前にコンテキストで暗号化された添付ファイルは、 モジュールで再暗号化されます。

フィールド	説明
状況	初期ジョブのステータスは新規です。ジョブがスケジュールどおりに実行された後、それに応じてステータスが更新されます。
期間開始	24 時間形式のジョブの開始時間。
期間終了	24 時間形式のジョブの終了時間。
テーブル	暗号化または復号化するテーブル。
フィールド	暗号化または復号化するフィールド。
サマリー	ジョブが実行中、完了した、またはエラーがある場合のジョブのステータス情報。

注: システムオーバーヘッドのために、一括暗号化、復号化、およびリキーのジョブはピーク時間外に実行するようにスケジュールする必要があります。ServiceNow AI Platform は、期間開始と期間終了の間でジョブを実行します。ジョブが 1 つの処理ウィンドウで完了しなかった場合、すべての処理が完了するまで、次の指定された処理ウィンドウでジョブが続行されます。

4. [送信] をクリックします。
5. ジョブをスケジュールすると、次のことができます。
   - 実行中のジョブをキャンセルするには、[ジョブをキャンセル] をクリックします。
   - [開始] をクリックしてジョブをすぐに開始します。
   - [更新] をクリックして、スケジュール済みジョブの変更を保存します。
   - [削除] をクリックして、スケジュールしたジョブを削除します。

一括暗号化または復号化を実行する

暗号化設定で一括暗号化を実行したり、一括復号化を実行して以前に暗号化された値を復号化したりすることができます。

始める前に
必要なロール：security_admin

このタスクについて
一括暗号化と一括復号化のためにスケジュール済みジョブを作成することもできます。手順については、「一括暗号化、復号化、およびリキーのジョブをスケジュールする」を参照してください。

一括暗号化と一括復号化は、暗号化フィールドの構成で単一暗号化モジュールを使用する場合にのみ使用できます。一括復号化は、単一暗号化方法と複数暗号化方法の両方で使用できます。

注: 一括暗号化と一括復号化は、リソースと時間がかかるため、ピーク時間外にのみ実行してください。

手順

1. 移動先 すべて > システムセキュリティ > フィールド暗号化 > 暗号化フィールドの設定.
2. 一括暗号化または一括復号化するフィールド暗号化フィールド構成を開きます。

3. [関連リンク] で、利用可能なオプションを選択します。
   - 一括復号化ジョブをスケジュール
   - 一括暗号化ジョブをスケジュール
4. ダイアログで選択を確認します。

結果
一括暗号化を実行する場合、すべての値は暗号化フィールド設定レコードで定義された暗号化モジュールで暗号化されます。一括復号化を実行すると、アクセスできる暗号化モジュールで暗号化されたフィールドのみが復号化されます。

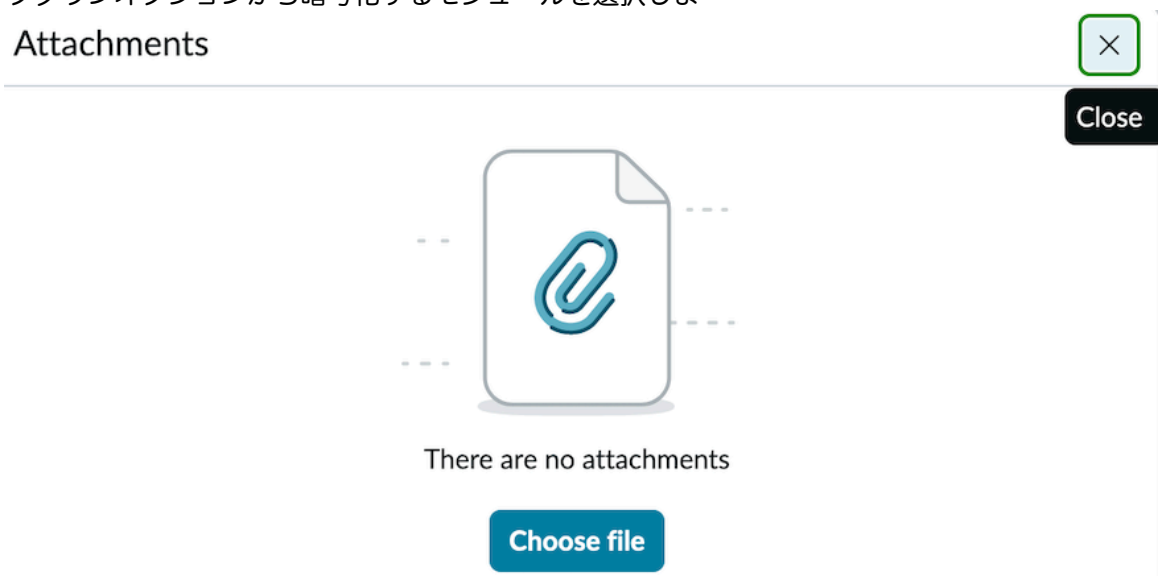
暗号化のために添付ファイルをアップロードする

フィールド暗号化と行条件を使用してレコードの添付ファイルを暗号化することで、機密ファイルを保護します。

始める前に
必要なロール:アドミンによって作成されたモジュールアクセスポリシー (MAP) に準拠する任意のロール。

手順

1. レコードに移動します。
2. [添付ファイルを管理] ペーパークリップアイコンを選択します。
3. [添付ファイル] ウィンドウで、[モジュールで暗号化] ドロップダウンオプションから暗号化するモジュールを選択します。



注:

行条件が暗号化フィールド構成に追加されている場合、このオプションは表示されません。添付ファイルは、行条件で定義されたモジュールで自動的に暗号化されます。同様に、このドロップダ

ウンオプションは、[添付ファイルを管理] アイコンを使用している場合にのみ表示され、添付ファイルをドラッグアンドドロップして追加する場合は表示されません。

4. [ファイルを選択] を選択して添付ファイルを見つけ、[開く] を選択して添付します。

結果
添付ファイルはフォームの上部に表示されます。暗号化された添付ファイルはロックアイコンで示されます。暗号化されたファイルを表示するための暗号化モジュールを持つユーザーのみに一覧表示されます。

フィールド暗号化エンタープライズ の例

これらの例では、顧客指定のキーを使用したフィールドと添付ファイルの暗号化について説明します。

フィールド暗号化エンタープライズ チュートリアル

このチュートリアルでは、キー管理フレームワーク (KMF) がある フィールド暗号化エンタープライズ を使用してインスタンスのフィールドを暗号化する方法を示します。また、独自のキーを使用する方法も示します。

始める前に

注: この手順は、フィールド暗号化エンタープライズ 機能にのみ適用されます。フィールド暗号化エンタープライズ の取得についての詳細は「フィールド暗号化 のアクティブ化」を参照してください。

必要なロール：admin または security_admin

注: security_admin は権限ロールです。権限ロールの使用に関する詳細については、「特権ロールへの昇格」を参照してください。

このタスクについて
このチュートリアルは、個人暗号化キーを既に作成してアップロードしたインスタンスから開始します。ServiceNow キーを使用できますが、この例では顧客指定のキーを使用します。

キーが暗号化モジュールに格納されたら、給与や社会保障番号など、特定のユーザーのアクセスが制限されるフィールドの設定を開始できます。暗号化フィールド構成で、機密データにアクセスできる権限を持つ担当者を指定します。

このタスクでは、2 つのシナリオを示します。1 つの例では、機密データを表示する権限のないユーザーのために、インシデントの [簡単な説明] フィールドを暗号化します。

添付ファイルを暗号化して、アクセス権が付与されたユーザーのみに表示するか、データの表示が制限されていないすべてのユーザーに表示することもできます。添付ファイルを暗号化するには、「添付ファイルの暗号化のチュートリアル」を参照してください。

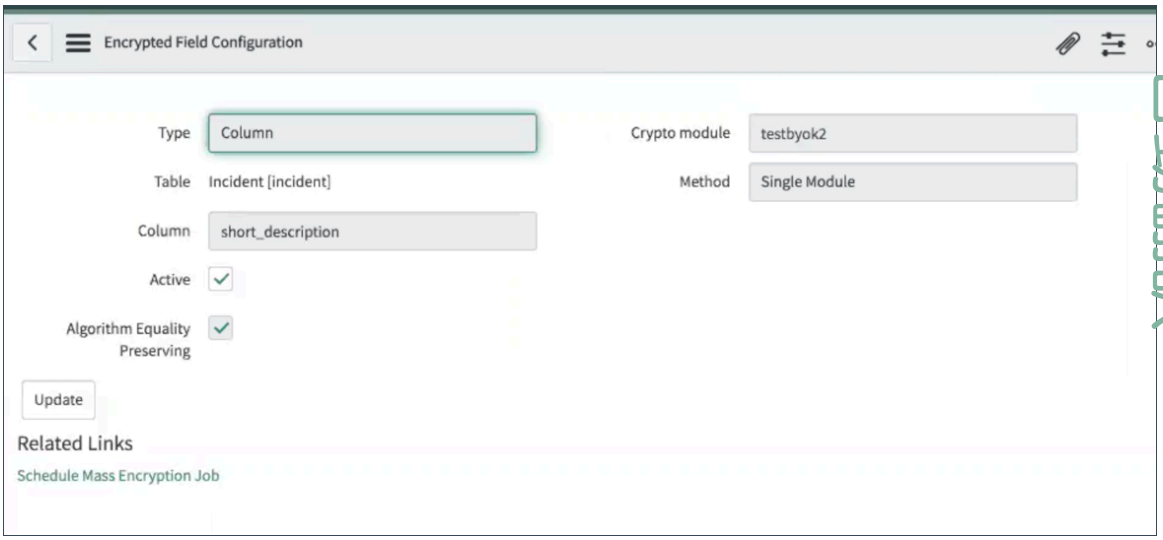
手順

1. フィールド暗号化エンタープライズ が有効になっていることを確認します。
2. column_level_encryption に暗号化モジュールを作成します。
   詳細については、「の暗号化モジュールを作成 フィールド暗号化 暗号化モジュールを作成する」を参照してください。
3. 移動先 システムセキュリティ > 暗号化フィールドの設定.
4. [**New**] をクリックします。
5. フォームのフィールドに入力します。

暗号化フィールドの構成フォーム

フィールド	説明
タイプ	個人キーを使用するには、列が必要です。
テーブル	機密情報を格納するテーブル。この例では、インシデント [インシデント] を選択します。
列	暗号化する機密データを表す列または特定の情報。この例では、**short_description** を選択します。
有効	フィールド構成を使用するためにアクティブをマークするオプション。
アルゴリズム等価性保存	このオプションは自動的に選択されます。
暗号化モジュール	個人キーで使用するために作成したモジュール。
メソッド	単一モジュールオプションは、1 つのモジュールにポリシーを適用するために使用されます。複数モジュールは、複数のモジュールにまたがってポリシーを適用するために使用されます。

暗号化フィールドの構成の例



6. [送信] をクリックします。

   モジュールアクセスポリシーを確立して、暗号化モジュールへのアクセスをアサインします。詳細については、「モジュールアクセスポリシーを作成する」を参照してください。

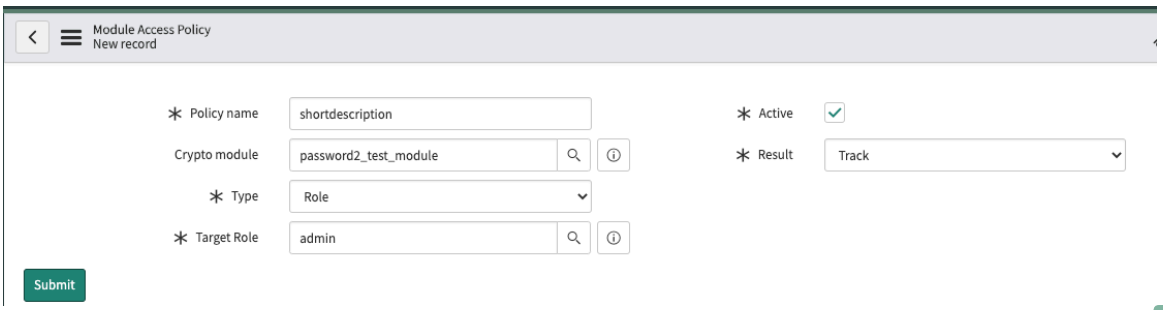
7. 移動先 キー管理 > モジュールアクセスポリシー > > 新規作成 > .
8. フォームで、フィールドに入力します。

モジュールアクセスポリシーフォーム

フィールド	説明
ポリシー名	簡単な説明などのポリシーの名前。
暗号化モジュール	キーを暗号化するために作成した暗号化モジュール。

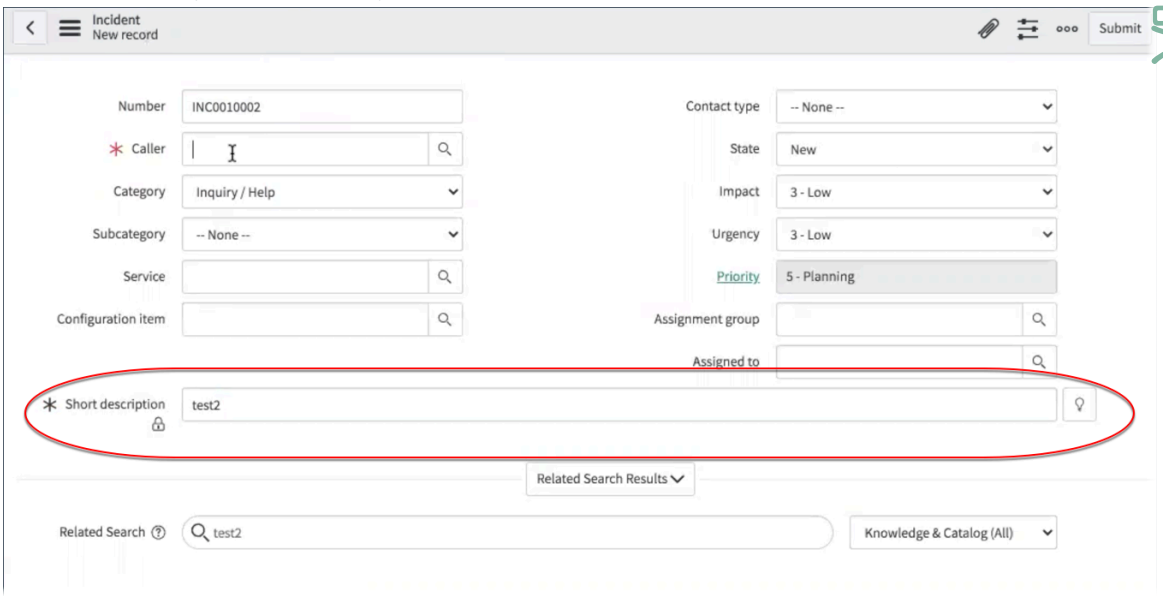
フィールド	説明
タイプ	暗号化ポリシーのアクセス指定のタイプ。ロールを使用して、割り当てられたロールを持つユーザーのみに暗号化フィールドへのアクセス権を付与します。
ターゲットロール	暗号化フィールドにアクセスできるロール。この例では、**Admin** を選択します。
有効	モジュールアクセスポリシーをアクティブ化するオプション。
結果	追跡オプションを使用すると、選択したロールのフィールドにアクセスできます (選択したロールに対してそのフィールドへのアクセスを制限するには、[却下] または [厳格な却下] を選択します)。

モジュールアクセスポリシーの例



9. [**Submit** (送信)] を選択します。
10. sn_kmf.admin ロールを持つユーザーとして、次の場所に移動します インシデント > 新規.

表示される暗号化フィールドの例



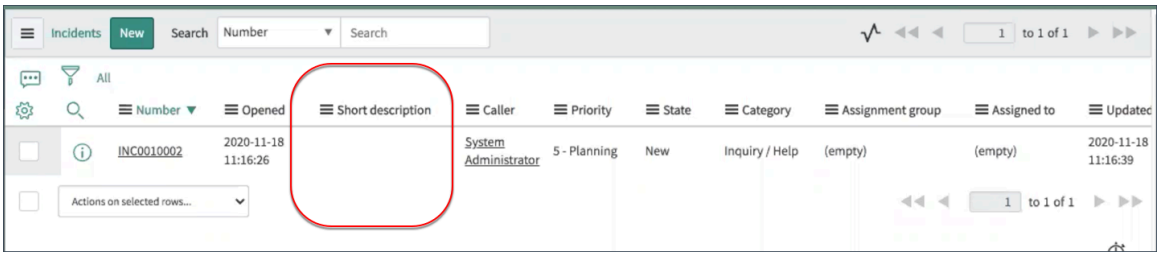
これで、モジュールアクセスポリシー構成に基づいて [簡単な説明] フィールドを表示できます。

注: モジュールアクセスポリシーを [追跡] に設定することで、sn_kmf.admin ロールに暗号化フィールド (簡単な説明) へのアクセス権が付与されました。フィールド名の下にあるロックアイコン ( ) は、フィールドが暗号化されていることを示しています。

これで、エンドユーザーとしてインシデントモジュールにアクセスして、暗号化フィールドの設定をテストできます。

11. 構成されたフィールド暗号化データの表示が制限されるユーザーとしてログインします。

暗号化されたフィールドレベルデータ



インシデント番号にアクセスしても、[簡単な説明] のデータは表示されません。

結果
フィールド暗号化エンタープライズ を使用して特定のフィールドへのアクセスを制御する対称キーを正常に使用しました。

添付ファイルの暗号化のチュートリアル

このチュートリアルでは、キー管理フレームワーク (KMF) がある フィールド暗号化エンタープライズ を使用してインスタンスの添付ファイルを暗号化する方法を示します。また、独自のキーを使用する方法も示します。

始める前に

注: この手順は、フィールド暗号化エンタープライズ 機能にのみ適用されます。フィールド暗号化エンタープライズ の取得についての詳細は「フィールド暗号化 のアクティブ化」を参照してください。

必要なロール：kmf 暗号化マネージャー

このタスクについて
このチュートリアルでは、顧客指定の暗号化キーを作成してアップロードしたインスタンスから開始します。キーを使用することもできますが、この例では顧客指定のキーを使用します。

機密性の高い添付ファイルをインスタンスにアップロードし、特定のユーザーからのアクセスを制限します。暗号化フィールド構成を使用して、機密データにアクセスできる権限を持つ担当者を指定します。

アクセス権を付与されたユーザーのみに表示するか、または、データの表示を制限されていないすべてのユーザーに表示するために、添付ファイルを暗号化する方法を示します。この例では、特定のロールが インシデントモジュールの添付ファイルにアクセスできないように制限しています。

注: フィールド暗号化エンタープライズ で複数のモジュールを使用できますが、添付ファイルの暗号化では単一のモジュールを使用する必要があります。

手順

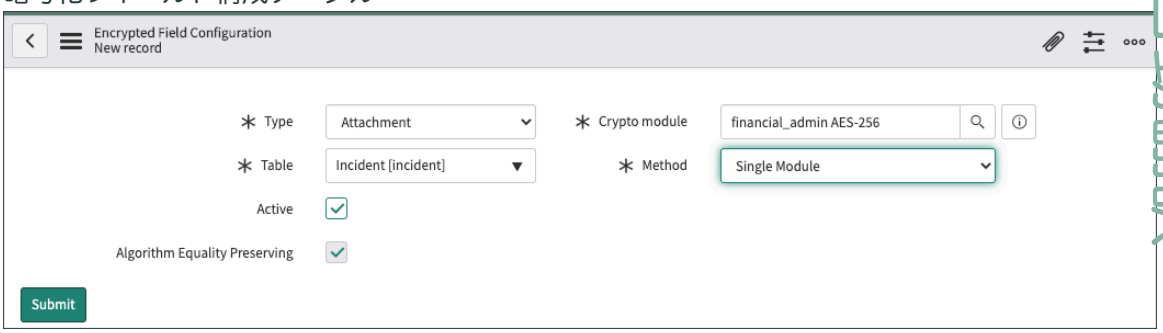
1. フィールド暗号化エンタープライズ が有効になっていることを確認します。
2. 暗号化モジュールを作成します。
   詳細については、「の暗号化モジュールを作成 フィールド暗号化」を参照してください。
3. 移動先 システムセキュリティ > 暗号化フィールドの設定.
4. **[New]** をクリックします。

5. フォームに次のように入力します。

暗号化フィールドの構成フィールド

フィールド	説明
タイプ	選択したテーブルから添付ファイルを暗号化するために個人キーを使用するには、添付ファイルを選択します。この例の場合、インシデントを選択します。
テーブル	機密情報にアクセスするテーブルを選択します。この例では、インシデント [インシデント] を選択します。
有効	フィールド構成を使用できるようにするために、アクティブをマークします。
アルゴリズム等価性保存	フィールド暗号化エンタープライズを選択すると、このフィールドは選択したテーブルに基づいて表示されます。
暗号化されたモジュール	個人キーで使用するために作成したモジュールを選択します。
メソッド	単一モジュールオプションは、1 つのモジュールにポリシーを適用するために使用されます。複数モジュールは、複数のモジュールにまたがってポリシーを適用するために使用されます。

暗号化フィールド構成テーブル



6. [送信] をクリックします。

モジュールアクセスポリシーを確立して、暗号化モジュールへのアクセスをアサインします。詳細については、「モジュールアクセスポリシーを作成する」を参照してください。

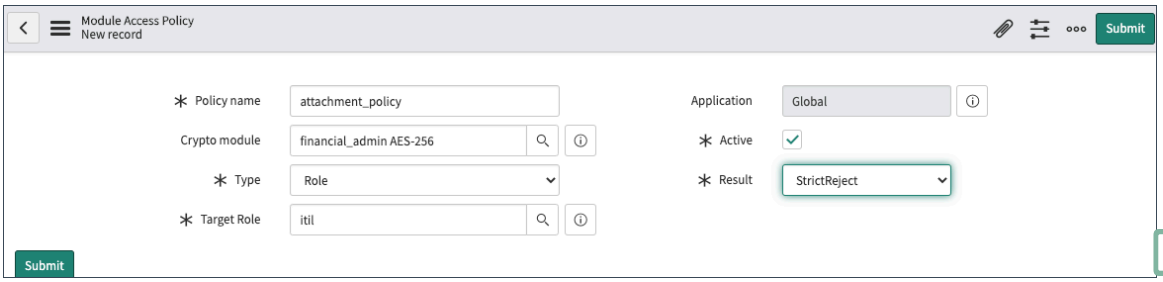
7. 移動先 キー管理 > モジュールアクセスポリシー > すべて.
8. **[New]** をクリックします。
9. フォームに次のように入力します。

モジュールアクセスポリシーフィールド

フィールド	説明
ポリシー名	ポリシーの名前 (「添付ファイルポリシー」など) を入力します。
暗号化モ	キーを暗号化するために作成した暗号化モジュールを選択します。

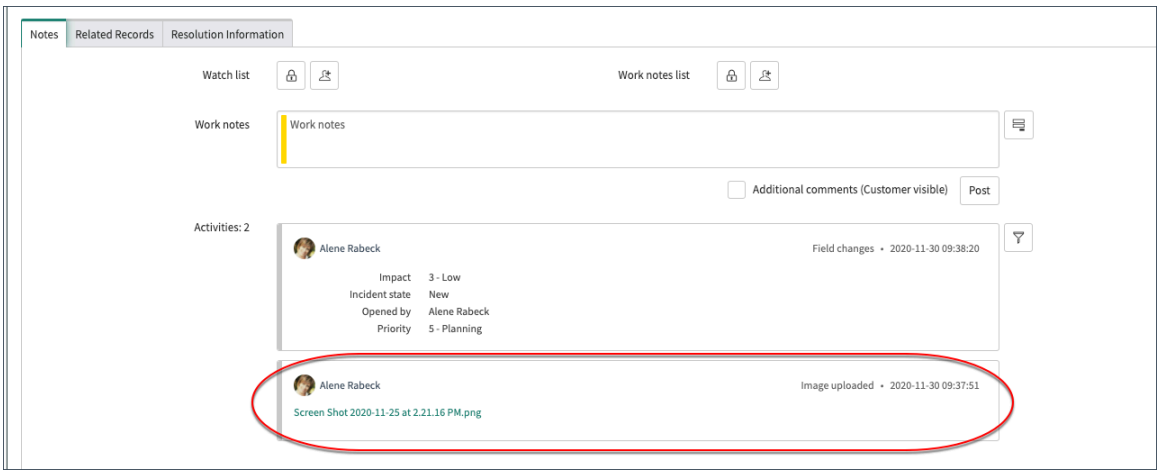
フィールド	説明
ジュール	
タイプ	割り当てられたロールを持つユーザーからの暗号化フィールドへのアクセスを制限するには、ロールを選択します。
ターゲットロール	暗号化フィールドにアクセスできないロールを選択します。この例では、**itil** を選択します。
有効	モジュールアクセスポリシーを使用できるようにするには、このチェックボックスをオンにします。
結果	選択したロールから添付ファイルへのアクセスを制御するには、[厳格な却下] を選択します(選択したロールにアクセス権を付与するには、[追跡] を選択します)。

モジュールアクセスポリシーフォーム



10. [送信] をクリックします。
11. admin またはインシデントを作成したユーザーとして、[インシデント] に移動し、メモ関連リストのアクティビティに添付ファイルを追加します。

ロールごとに利用可能な添付ファイル



12. 暗号化された添付ファイルへのアクセスが制限されているユーザーとしてログインします。
13. インシデントを開き、アクティビティ：セクションまでスクロールします。
制限付きロールのユーザーは、添付ファイルを開くためのリンクにアクセスできません。
14. これで、顧客指定のキーを使用して、 フィールド暗号化エンタープライズで特定の添付ファイルへのアクセスを制御できました。

フィールド暗号化エンタープライズ

フィールド暗号化エンタープライズ では キー管理フレームワーク (KMF) を使用して、インスタンスでのフィールドと添付ファイルの暗号化方法と復号化方法をカスタマイズおよび管理できます。フィールド暗号化エンタープライズを使用するには、サブスクリプションが必要です。

重要: このトピックでは、 フィールド暗号化 のエンタープライズバージョンについて説明します。フィールド暗号化 の標準バージョンの詳細、または 2 つのバージョンの違いについては、「フィールド暗号化 の探索」を参照してください。

フィールド暗号化エンタープライズ は フィールド暗号化 を前提としており、 キー管理フレームワーク とそのキー管理機能の完全なサポートを使用します。 フィールド暗号化エンタープライズ は、アプリケーションレベルのフィールド暗号化のためのキー保護とキーライフサイクル管理を提供します。すべてのキーは、最終的に FIPS (連邦情報処理標準) 140-2-L3 ハードウェアセキュリティモジュール (HSM) に基づくキーラッピング階層で保護されます。

フィールド暗号化エンタープライズ を使用すると、サポートされているフィールドを NIST 800-57 プラクティスに従って暗号化および復号化する方法を管理できます。また、適切なキーの保護と管理のための統合を含む、最新バージョンのフィールドレベルの暗号化を使用します。

具体的には、 フィールド暗号化エンタープライズ は KMF 暗号化モジュールを使用して、サーバー側の暗号化をより詳細に制御できます。 KMF 、キー階層とエンベロープ暗号化を使用して、適切なデータ暗号化キーの保護を検証します。インスタンスは、設定した暗号化モジュールでデータを暗号化します。各モジュールのアクセスポリシーを作成してから、暗号化仕様とアクセスポリシーを設定し、キーライフサイクル管理コントロールを制御できます。

フィールド暗号化エンタープライズ は、以下に基づくモジュールアクセスポリシーをサポートしています。

- スコープ
- ロール
- スクリプト
- リソース交換
- システムユーザー

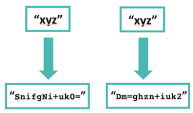
詳細については、「モジュールアクセスポリシーを作成する」を参照してください。

注: サポートされている フィールド暗号化 の機能と、 フィールド暗号化エンタープライズ エンタイトルメントをアップグレードしてサブスクライブする方法の詳細については、「 暗号化とキー管理のサブスクリプションバンドル」を参照してください。

自動翻訳

暗号化の用語

用語	説明
キー管理 	キー管理のサポート フィールド暗号化エンタープライズ の基礎は Key Management Framework (KMF) です。 次の機能が得られます。 • キーのライフサイクル管理 • キーロテーション詳細については、「キーのローテーション」を参照してください。 • FIPS 140-2-L3 ハードウェアセキュリティモジュール (HSM) によるキーの保護と生成 • ロールと職務の分離 • 本番インスタンスと非本番インスタンスなどのインスタンス間でデータ暗号化キーを安全に転送します。 • キーラッピングを使用した顧客指定のキー (CSK) • 非決定的暗号化 • 一括暗号化/復号化 • キーのアクセス/使用の監査 詳細については、「キー管理フレームワークリファレンス」を参照してください。
顧客指定のキー 	顧客指定のキーのサポート フィールド暗号化エンタープライズ の最大のメリットの 1 つは、暗号化に独自のキーを使用できることです。アドミニストレーターは、ServiceNow 指定のキーまたは独自の顧客指定のキー (CSK) を ServiceNow AI Platform の暗号化に使用できます。 キーのライフサイクルを管理し、キーの取り消し、ローテーション、および無効化のタイミングを決定することもできます。顧客指定のキーを有効にして暗号化モジュールを作成した後、トークンと短期公開キーをダウンロードします。トークンと公開鍵を使用してキーをラップし、インスタンスにアップロードします。顧客指定のキーを使用するには、「キータイプを選択するためにフィールド暗号化を設定する」および「での顧客指定のキーの使用 フィールド暗号化エンタープライズ」を参照してください。

用語	説明
フィールド暗号化 	フィールド暗号化と添付ファイルの暗号化の両方をサポート フィールド暗号化と添付ファイルの暗号化はどちらも、暗号化フィールド構成を介して暗号化モジュールとアクセスポリシーを使用します。暗号化フィールド構成フォームを使用して、列の暗号化タイプまたは添付ファイルの暗号化を選択します。詳細とサポートされているフィールドタイプについては、「暗号化フィールド構成を設定する」を参照してください。
非決定的暗号化 	非決定的暗号化のサポート フィールド暗号化エンタープライズ は、セキュリティ強化のために非決定的暗号化をサポートしています。システムが同じデータを複数回暗号化する場合、暗号化テキストは毎回異なります。非決定的暗号化は、暗号ブロックチェーン (CBC) を使用した Advanced Encryption Standard (AES) 暗号化で使用できます。 この機能は、暗号化仕様のアルゴリズム定義ステージの等価性保存オプションで有効にできます。暗号化モジュールの暗号化仕様を作成し、キーを暗号化し生成するためのアルゴリズムを定義します。 暗号化操作に使用するメカニズムを定義し、非決定的暗号化を有効にする方法の詳細については、「暗号化モジュールを作成する」を参照してください。
リソース交換 	リソース交換 フィールド暗号化エンタープライズ は KMF 暗号化 API を使用してインスタンス間を保護し、機密性、完全性、認証、および否認防止を提供します。リソース交換 は、安全な方法でインスタンス間でリソースを交換する機能を提供する KMF 機能です。詳細については、「キー管理フレームワークリソース交換」を参照してください。

注: フィールド暗号化エンタープライズをアクティブ化しないことを選択した場合でも、フィールド暗号化を使用できます。詳細については、「フィールド暗号化 の探索」を参照してください。

フィールド暗号化エンタープライズ はオンプレミスの顧客をサポートしています。ドメインセパレーションはサポートされていません。

追加の暗号化フィールドのサポート

フィールド暗号化 の標準バージョンでは、暗号化された列は 5 つに制限されています。フィールド暗号化エンタープライズ は、無制限の数の暗号化列をサポートしています。

サポートされるフィールド情報

次のフィールドタイプを暗号化できます。

- 添付ファイル
- 日付
- 日付/時刻

- メール
- HTML
- ジャーナル
- ジャーナル入力
- ジャーナルリスト
- 電話
- 文字列テキスト
- 翻訳 (変換) 済みフィールド
- 翻訳された HTML
- 翻訳されたテキスト
- URL

添付ファイルの暗号化

デフォルトでの添付ファイルの暗号化

フィールド暗号化 を使用しているお客様は、アクティブな暗号化フィールド構成 (EFC) タイプが *Attachment* のテーブルの添付ファイルをデフォルトで暗号化しています。

EFC 構成で定義されているこのデフォルトの暗号化は、アドミニストレーターがこれらのテーブルのアップロード時に添付ファイルを暗号化することを手動で宣言する必要がないことを意味します。

アドミニストレーターは、ユーザーが暗号化されていないファイルを添付することを禁止できます。

詳細については、「ユーザーが暗号化されていないファイルを添付できないようにする」を参照してください。

デフォルトの暗号化のオプトアウト

EFC 構成に基づいてデフォルトで添付ファイルを暗号化したくない場合は、ServiceNow サポートに連絡してこのオプションをオプトアウトできます。

この機能をオプトアウトするには、ServiceNow サポートでサポートケースを作成し、ケースレコードのコメントに次のステートメントを含めます。

「I [顧客名], understand that I am asking ServiceNow to turn off a recommended security best practice for attachments, and that [customer company] assumes any additional risk related to their configuration and use of unencrypted attachments in the ServiceNow application.」

API サポート

フィールド暗号化エンタープライズ は、暗号化フィールドに暗号化されたデータを挿入できるように、setDisplayValue() および setValue() API を更新します。また、getDisplayValue() と getValue() がクリアテキスト値を返すようにもなります。

次のスクリプトは、インシデントの簡単な説明が暗号化されている場合の API の変更を示しています。

```
var gr = new GlideRecord('incident'); //creates a new incident
gr.setValue('short_description','test123'); //sets the value to test123
```

```
var sys_ID = gr.insert(); //inserts the record in the Incident table.
gs.info(gr.getValue('short_description')); //displays the unencrypted value
```

getValue() を使用して暗号化テキストを取得すると、スクリプトで暗号テキストを返されなくなります。スクリプトは、ユーザーが暗号化モジュールにアクセスできるという想定で、プレーンテキストを返します。暗号化モジュールにアクセスできないユーザーについては、getValue() は暗号テキストを返します。

キー管理を使用したクラウド暗号化

ServiceNow® クラウド暗号化 は、拡張キー管理とともに、ブロック暗号化を使用したデータベースの暗号化ストレージを提供します。クラウド暗号化は ServiceNow Platform Encryption サブスクリプションバンドルで利用可能です。

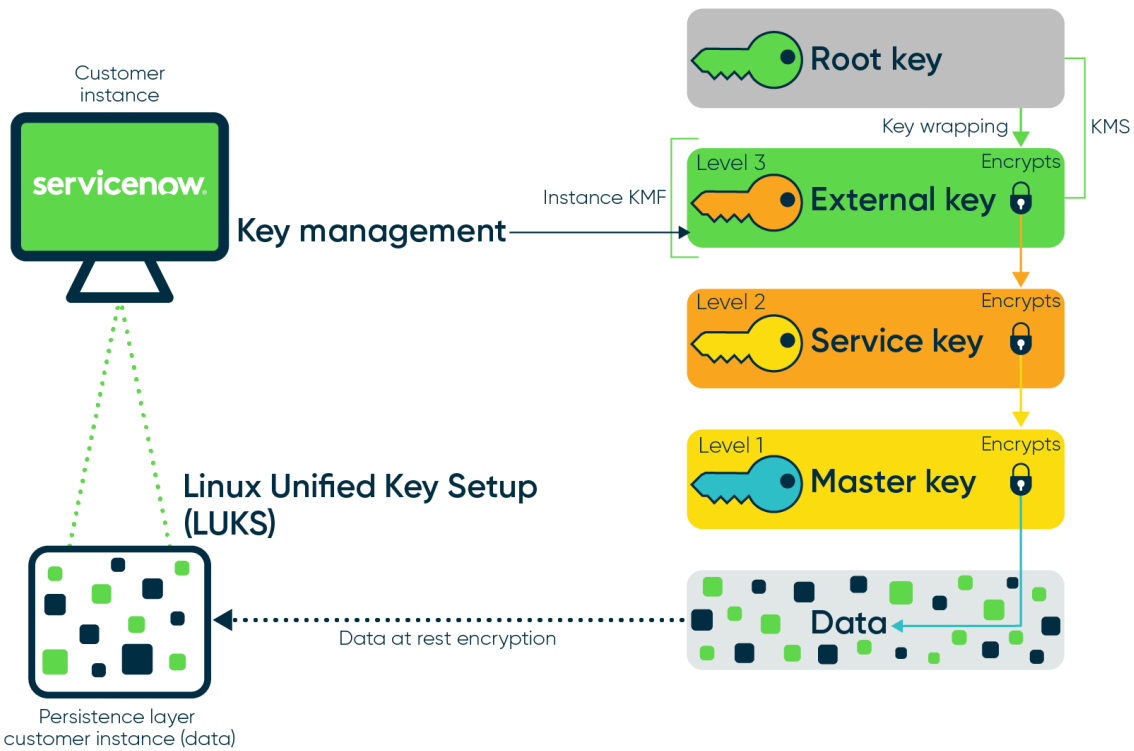
クラウド暗号化は以下の機能を提供します。

- 職務の分離
- ServiceNow 管理キーのローテーション
- 顧客管理キーオプション

> **注:** 組織が独自の暗号化ツールまたはライブラリ、エンタープライズキー管理システム、またはハードウェアセキュリティモジュール (HSM) によって生成されたキーマテリアルを使用する必要がある場合は、このオプションを検討してください。詳細については、「キー管理操作」を参照してください。

次の図は、クラウド暗号化 の仕組みを示しています。

クラウド暗号化の概要



クラウド暗号化 キー管理モジュールは、次のサブモジュールで構成されています。

- キー管理操作：
  - キーのリストにアクセスします。
  - キーのローテーション操作を実行します。
  - 顧客管理キーを取り消します。
- キー管理トランザクション：

  使用されたキーに対して発生したすべてのトランザクションを参照します。

  暗号化に独自の顧客管理キーを使用します。

状況によっては、顧客管理キーの使用時にキーの取り消し要求を選択できます。これを行うには、オプションの Cloud Encryption Withdraw and Resupply アドオン SKU のライセンスを取得してから、キーの取り消し機能のアクティブ化をカスタマーサービス & サポートのチームメンバーに依頼する必要があります。

[クォーラムコントロールポリシー設定] オプションは、取り消し機能が有効の場合にアクティブ化されます。無効の場合、モジュールがメニューに表示されません。この機能は、顧客管理キーを使用する場合にのみアクティブ化できます。このポリシーでは、取り消し機能がアクティブ化されていれば、クォーラムに関する設定を構成できます。この機能の詳細については、「クォーラムコントロールポリシー」を参照してください。

クラウド暗号化は、MariaDB データベースと RapterDB データベースの本番インスタンスと非本番インスタンスをサポートしています。クラウド暗号化は、ServiceNow Commercial Cloud、Government Customer Cloud (GCC) pod 101、および ServiceNow Protected Platform – Australia (SPP-AU) でサポートされています。

クラウド暗号化のライセンスと有効化

クラウド暗号化 のライセンスの詳細については、「暗号化とキー管理のサブスクリプションバンドル」を参照してください。

新しいインスタンスを使用する、ライセンス取得済みのお客様の場合、新しいインスタンスのプロビジョニングにクラウド暗号化が含まれます。

既存のインスタンスを使用する、ライセンス取得済みのお客様の場合、クラウド暗号化へのインスタンスの移行を要求するには、KB1117369 の手順に従ってください。インスタンスでクラウド暗号化を有効にするために サービスカタログ アイテムを要求するには、顧客 admin またはパートナー admin ロールを持っている必要があります。この機能を有効にするには、1 時間のメンテナンス期間が必要です。

クラウド暗号化 UI

クラウド暗号化が有効になっているときに、security_admin ユーザーが sn_kmf.admin ロールを持っていると、このユーザーにクラウド暗号化ユーザーインターフェイス (UI) が表示されます。

クラウド暗号化 UI にアクセスするには、ナビゲーションバーで「クラウド暗号化キー管理」を検索します。[キー管理操作] セクションに移動して、アクティブなキーの詳細や、インスタンスのクラウド暗号化が有効になっているかどうかなど、暗号化キーに関する情報を確認します。

キー管理操作

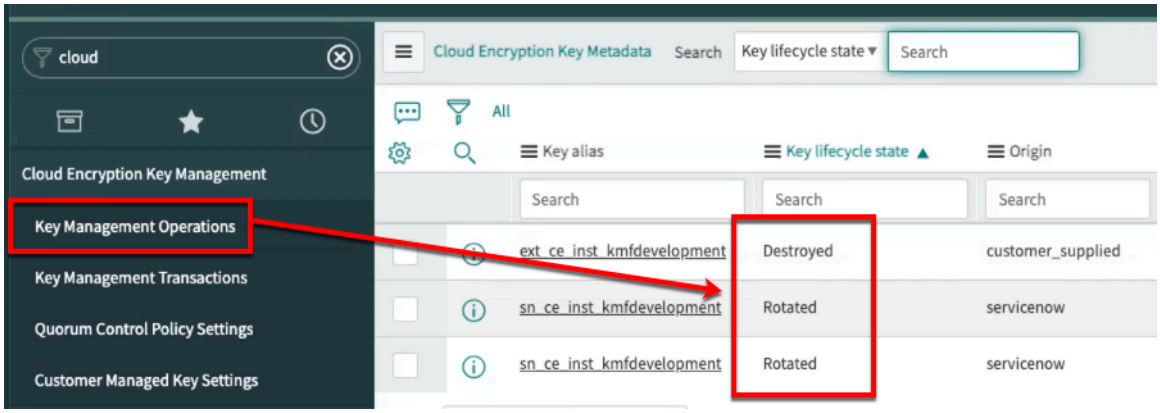
キー管理操作サブモジュールは、ServiceNow クラウド暗号化 で使用されるすべての暗号化キーを表示して管理するためのアクセスを提供します。

クラウド暗号化の開始

キーのライフサイクル状況

システムには常に 1 つのアクティブなキーしか存在しません。キーを選択すると、ローテーションされたキーまたは取り消されたキーや対応するタイムスタンプなど、選択したキーのアクティビティにアクセスできます。

キーのライフサイクル状況は、実行されたキー管理操作に従って更新されます。



詳細は「ServiceNow 管理キーのローテーション」または「顧客管理キーのローテーション」を参照してください。

注: キーロテーションプロセスは完了までに 20 分かかる場合があります。

ServiceNow 管理キーのローテーション

アクティブなクラウド暗号化 ServiceNow管理キーをローテーションします。

始める前に
必要なロール：sn_kmf.admin または sn_kmf.cryptographic_manager

重要: 顧客管理キーを使用している場合は、「顧客管理キーのローテーション」を参照してください。

手順

1. 移動先 すべて > クラウド暗号化キー管理 > キー管理操作.
   クラウド暗号化 のキーメタデータのリストがロードされます。インスタンスで使用されているすべてのキーがリストされます。一度に 1 つのキーのみをアクティブにすることができます。

   クラウド暗号化モジュールに初めてアクセスすると、最初のキーローテーションが実行された後にキーエントリが使用可能になります。ServiceNow 管理キーがシステムのデフォルトです。
2. テーブルからアクティブなキーを選択します。
   [キー定義] テーブルに、ServiceNow で生成されたキーに関する一般的な情報が表示されます。
3. [キーのローテーション] ボタンを選択します。
   キーのローテーションを続行するか操作をキャンセルするかを選択できる通知が表示されます。
4. **[OK]** を選択してキーをローテーションします。
   [キー定義] ページの上部に確認メッセージが表示されます。
5. [キー管理操作] 画面に戻り、クラウド暗号化キーメタデータテーブルを更新します。
   現在のアクティブなキーと、現在のアクティブなキーの代わりにローテーションするために生成されているキーのエントリが表示されます。利用可能なさまざまな状況については、「キー管理フレームワークの主要なライフサイクル状況」を参照してください。

アクティブなキーはキーバージョン `0` でリストされ、生成されたキーのバージョンは `1` です。

6. 元のキーのエントリを開くとキー管理トランザクションが表示されます。
詳細については、「キー管理トランザクション」を参照してください。
前にアクティブだったキーのバージョン `0` のキーのライフサイクル状況が [ローテーション済み] にアップデートされ、新しいキーのバージョン `1` が [有効] になります。

顧客管理キーの準備

次の手順に従って、インスタンスにアップロードする顧客管理キーを準備します。

始める前に

必要なロール：sn_kmf.admin または sn_kmf.cryptographic_manager

このタスクについて

顧客管理キーの場合は、任意の暗号化ライブラリまたは HSM を使用してキーを生成できます。このキーは AES 256 ビットのキーであり、RSAES_OAEP_SHA_256 暗号化スキーマを使用したクラウド暗号化ラッピング証明書でラップされている必要があります。

注:

OpenSSL 暗号化ツールを使用してキーを生成する場合、OpenSSL バージョンはバージョン 1.1.1x 以降である必要があります。

Windows を使用して顧客管理キーを作成してラップする場合は、Git Bash などの Bash シェルサポートアプリケーションを使用して、ラップされたキーを生成する必要があります。

手順

1. OpenSSL を使用して、AES-256 ビット対称キーとして使用するランダム値を生成します。

   たとえば、openSSL を使用すると、`openssl rand 32` コマンドでこのキーを生成できます。

   互換性のために、対称キーには次の属性が必要です。

属性	値
キータイプ	Advanced Encryption Standard (AES) アルゴリズムベースの対称キー。
鍵サイズ	256 ビット (32 バイト)
キーラッピング要件	○ RSA 暗号化アルゴリズム ○ 最適な非対称暗号化パディング (OAEP) ○ SHA-256 ハッシュ関数 (RSAES_OAEP_SHA_256) ○ Base64 アルゴリズムを使用したエンコード

2. キーをファイルに保存します。
たとえば、openSSL コマンド `openssl rand 32 > plaintext_key.bin` は 32 バイトのキーを生成し、`plaintext_key.bin` という名前のファイルに保存します。

重要: 今後の参照のためにこのファイルを安全に保存してください。このキーはアップロード用の公開鍵でラップされます。

3. インスタンスからダウンロードしたラッピング証明書ファイルから公開キーを抽出します。

```
openssl x509 -pubkey -noout -in wrapping_cert.pem > public_key.pem
```

注: ラッピング証明書をダウンロードするには、「」を参照してください。

4. RSAES_OAEP_SHA_256 アルゴリズムを使用して、ラッピング証明書と一緒にダウンロードした公開鍵で生成されたキーをラップします。

```
cat plaintext_key.bin | openssl pkeyutl -encrypt -inkey public_key.pem -pubin -pkeyopt
 rsa_padding_mode:oaep -pkeyopt rsa_oaep_md:sha256 | openssl base64 -A -out
 wrapped_key.txt
```

このコマンドで指定されたファイルには、CMK プロセスの SN に提供できるラップされた顧客管理キーが含まれています。

ServiceNow と顧客管理キーの切り替え

ServiceNow クラウド暗号化で使用する、顧客管理キーと ServiceNow 管理キーを切り替えます。

デフォルトでは、インスタンスは ServiceNow 管理キーを使用するように構成されていて、ServiceNow による暗号化キーの生成がアクティブになっています。しかし、アドミンであれば顧客管理キーを使用する選択が可能です。ServiceNow 管理キーに戻すことも選択できます。

顧客管理キーのローテーション

クラウド暗号化 の顧客管理キーをラップした後、顧客管理キーをインスタンスにローテーションします。

始める前に

必要なロール：sn_kmf.admin または sn_kmf.cryptographic_manager

手順

1. 移動先 すべて > クラウド暗号化キー管理 > キー管理操作.
[Cloud Encryption のキーメタデータ **(Cloud Encryption Key Metadata)]** リストがロードされます。インスタンスで使用されているすべてのキーがリストされます。
2. **[Cloud Encryption** のキーメタデータ **(Cloud Encryption Key Metadata)]** リストで、アクティブなキーのレコードを開きます。
キーが複数ある場合は、[キーのライフサイクル状況] が [有効] であるキーを選択します。インスタンスでは、有効なキーは 1 つのみです。
3. キー定義レコードで、[キーのローテーション] ボタンを選択します。
4. [顧客管理キーをアップロード] ウィンドウで、リストされている手順を実行します。

   a. [ラッピング証明書をダウンロード] を選択します。
   public_certificate....zip がローカルマシンにダウンロードされ、顧客管理キーをラップするために使用されます。

   警告: 顧客管理キーのローテーションや顧客管理キーへの切り替えのたびにラッピング証明書をダウンロードすることで、証明書に関連する潜在的な問題を回避します。

   b. [参照] を選択して顧客管理キーをアップロードし、ラップされた暗号化キーを検索して選択します。
   別のファイルを選択するには、ファイルを選択して [削除] を選択します。

c. [添付ファイル] ウィンドウを閉じます。

d. キーをアップロードするには、**[OK]** を選択します。
キーが適切な形式である場合は確認メッセージが表示され、それ以外の場合はエラーメッセージが表示されます。キーファイルはキー定義レコードに添付されています。

[キー管理トランザクション] テーブルに、証明書のダウンロードとキーのアップロードの手順が表示されます。要求ステップの詳細については、「キー管理トランザクション」を参照してください。

5. 移動先 すべて > クラウド暗号化キー管理 > キー管理操作 をクリックしてキーのリストを表示します。
キーのリストに、顧客管理キーの新しいレコードが表示されます。この新しいキーの [作成元] の値は **[customer_supplied]** で、ステータスは [有効] です。前のキーのステータスは [ローテーション済み] になります。

顧客管理キーに切り替える

ServiceNow クラウド暗号化 の顧客管理キーを使用します。

始める前に

必要なロール：sn_kmf.admin または sn_kmf.cryptographic_manager

顧客管理キーに切り替えるには、この手順の一環として、ラップされた顧客管理キーをアップロードできる状態にしておく必要があります。このキーをアップロードする準備の詳細については、「顧客管理キーの準備」を参照してください。キーのアップロード後、このプロセスによって新しいキーのキーローテーションが開始されます。

手順

1. 移動先 すべて > クラウド暗号化キー管理 > キー管理操作.
2. [クラウド暗号化のキーメタデータ **(Cloud Encryption Key Metadata)]** リストで、アクティブなキーのレコードを開きます。
キーが複数ある場合は、[キーのライフサイクル状況] が [有効] であるキーを選択します。インスタンスでは、有効なキーは 1 つのみです。
3. フォームの [関連リンク] セクションで、[顧客管理キーに切り替え] リンクを選択します。
4. [顧客管理キーに切り替え] ダイアログボックスで、[管理キーをアップロード] ボタンをクリックします。
5. [顧客管理キーをアップロード] ダイアログボックスで、リストされている手順を実行します。

a. [ラッピング証明書をダウンロード] を選択します。

⚠ 警告: 顧客管理キーのローテーションや顧客管理キーへの切り替えのたびにラッピング証明書をダウンロードすることで、証明書に関連する潜在的な問題を回避します。

b. [参照]を選択し、プロンプトに従ってデバイスからキーを選択してアップロードします。

c. [顧客管理キーに切り替え] を選択します。

顧客管理キーに切り替える要求がインスタンスによって生成されます。現在のフォームで、元々アクティブであったキーの [キーライフサイクル状況] が [ローテーション済み]に変わっていることがわかります。

6. 移動先 すべて > クラウド暗号化キー管理 > キー管理操作 をクリックしてキーのリストを表示します。
   キーのリストに、顧客管理キーの新しいレコードが表示されます。この新しいキーの [作成元] の値は **[customer_supplied]** で、ステータスは [有効] です。

結果
インスタンスが ServiceNow クラウド暗号化 の顧客管理キーを使用するようになりました。

重要: 暗号化キーのコピーは、常に安全な場所でキー管理操作ができるようにしてください。このキーがないと、インスタンスにアクセスできなくなる恐れがあります。

ServiceNow 管理キーに切り替える

顧客管理キーから ServiceNow クラウド暗号化 管理キーに戻します。

始める前に
必要なロール：sn_kmf.admin または sn_kmf.cryptographic_manager

手順

1. 移動先 すべて > クラウド暗号化キー管理 > キー管理操作.
2. [クラウド暗号化のキーメタデータ **(Cloud Encryption Key Metadata)]** リストで、アクティブなキーのレコードを開きます。
   キーが複数ある場合は、[キーのライフサイクル状況] が [有効] であるキーを選択します。インスタンスでは、有効なキーは 1 つのみです。
3. フォームの [関連リンク] セクションで、**[ServiceNow** 管理キーに切り替え] リンクを選択します。
4. **[ServiceNow** 管理キーに切り替え] ダイアログボックスで、**[ServiceNow** 管理キーに切り替え] ボタンを選択します。
   ServiceNow 管理キーに切り替える要求がインスタンスによって生成されます。現在のフォームで、元々アクティブであったキーの [キーライフサイクル状況] が [ローテーション済み]に変わっていることがわかります。
5. 移動先 すべて > クラウド暗号化キー管理 > キー管理操作 をクリックしてキーのリストを表示します。
   キーのリストに、ServiceNow 管理キーの新しいレコードが表示されます。この新しいキーの [作成元] の値は **[ServiceNow]** で、ステータスは [有効] です。

キーローテーションをスケジュール

ServiceNow 管理キーの自動ローテーションのスケジュールを設定します。このプロセスでは、暗号化キーが自動で廃止され、古いキーが新しく生成された暗号化キーに置き換えられます。顧客管理キーを使用している場合は、このスケジュールにより、カスタムキーを手動でローテーションするように促すリマインダーを提供できます。

始める前に
必要なロール：sn_kmf.admin

手順

1. 移動先 すべて > クラウド暗号化キー管理 > スケジュールされたキーローテーション設定.
2. [スケジュールされたキーローテーションを有効化] チェックボックスをオンにします。
3. ビジネスニーズに基づいて、残りのフィールドに入力します。

スケジュールされたキーローテーション設定

フィールド	説明
キーローテーション間の月数 (最大 60 か月)	キーローテーション間の月数。この値はデフォルトで 12 で、最大 60 か月にすることができます。
キーローテーションを実行する曜日	キーローテーションを実行する曜日。
キーローテーションを実行する時刻	キーローテーションを実行する時刻。
次回のキーローテーションの日時	次回キーローテーション実行予定の日時。この値は直接編集できず、選択に基づいて自動的に計算されます。
リマインダーを送信するキーローテーションまでの日数 (最大 15 日)	インスタンスが通知を送信するキーローテーションの実行日までの日数。
メール通知は、次に示す承認されたセキュリティアドミニストレーターのリストに送信されます	キーローテーションの通知を受信するユーザーのリスト。デフォルトでは、 システムアドミニストレーターがこのリストに含まれています。

4. [送信] を選択します。
   [送信] を選択すると、フォームの上部に通知が表示されます。この通知により、キーローテーションと通知のスケジュールを確認できます。

> ⚠ 警告:
>
> スケジュールされた各キーローテーションには一意の署名があるため、ジョブの完全性が保証され、不正な変更が検出されます。スケジュール済みジョブの署名は、インスタンスごとに一意です。キーローテーションのスケジュール済みジョブをソースインスタンス A からターゲットインスタンス B にクローンすると、インスタンス B のスケジュール済みジョブは署名検証に失敗します。これが発生した場合は、[スケジュールされたキーローテーションを有効化] チェックボックスをオフにしてから再度オンにすることで、署名を再作成できます。この問題の詳細については、「KB1247113」を参照してください。

顧客管理キーを取り消す

顧客管理キーの取り消し機能が有効になると、[キー管理操作] ページで取り消し操作を使用できるようになります。キーの取り消しとクォーラムの承認操作も管理できます。

始める前に

必要なロール：sn_kmf.admin または sn_kmf.cryptographic_manager

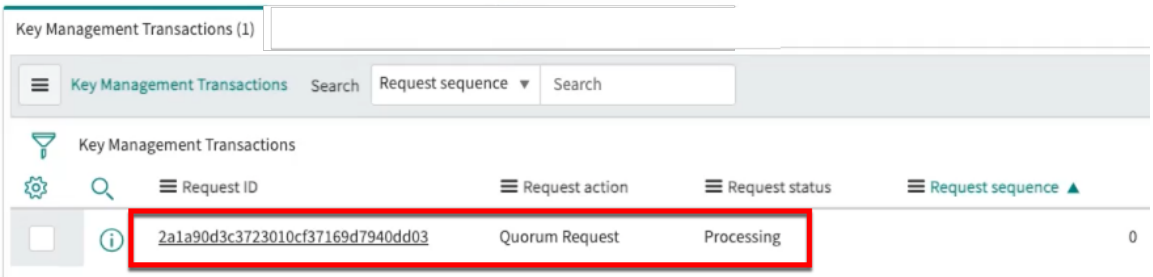
このセクションは、クラウド暗号化 へのオプションのアドオンである Cloud Encryption Withdraw and Resupply がライセンスされている場合にのみ適用されます。

手順

1. 移動先 すべて > クラウド暗号化キー管理 > > キー管理操作.
2. テーブルからアクティブな顧客管理キーを選択します。
   [キー定義] テーブルに、顧客キーに関する一般的な情報が表示されます。キーの取り消し機能が利用可能になります。
3. [キーを取り消す] を選択して、取り消しプロセスをトリガーします。

> ⚠ **警告:**
>
> キーの取り消しの警告メッセージが表示されます。キーを取り消すと、インスタンスのシャットダウンがトリガーされ、取り消されたキーを使用して復元操作が実行されるまでそのままになります。
>
> > ✖ **危険:** 取り消された同じキーでのみ復元操作を実行できます。別のキーにローテーションする場合は、取り消されたキーを復元した後に行う必要があります。
>
> 取り消された顧客管理キーが、ServiceNow がバックアップを保持する期間内に復元されない場合 (詳細については、バックアップおよび復元 SOP [標準運用手順] を参照)、インスタンスデータベースのバックアップにアクセスできなくなります。この方法で失われたバックアップデータは復旧できません。

4. **[OK]** を選択してキーを取り消します。
   キーの取り消し機能に疑問がある場合は、[キャンセル] を選択します。
   [キー定義] 画面に戻り、確認メッセージが表示されます。
5. [キー定義] ページを更新して、保留中の取り消し要求を表示します。



クォーラムコントロールポリシーが有効になっている場合、キーの取り消しを完了するには、承認ワークフローを正常に完了する必要があります。詳細については、「クォーラムコントロールを管理する」を参照してください。

顧客管理キーの再供給

キーの取り消し操作が完了したら、顧客管理キーをインスタンスに再供給する必要があります。

始める前に

必要なロール：sn_kmf.admin または sn_kmf.cryptographic_manager

> ℹ **注:** このセクションは、Cloud Encryption Withdraw and Resupply がライセンスされている場合にのみ適用されます。

手順

1. Now Supportに移動し、以下に移動します サービスカタログ > カタログ > インスタンス管理 > インスタンスの復元:管理キーの供給.
2. **[Request** (要求)] をクリックします。
3. [インスタンスの復元 - 管理キーの再供給 **(Instance Restore - Resupply Managed Key)]** ウィンドウで、[インスタンスを選択 **(Select Instance)]** ドロップダウンでインスタンスを選択します。
4. ラッピング証明書のテキストをクリックして、ラッピング証明書をダウンロードします。

> ⚠ **警告:** 顧客管理キーをローテーションまたはアップロードするたびに、新しいラッピング証明書をダウンロードする必要があります。

5. アップロードするキーを準備します。
   このプロセスの詳細については、「顧客管理キーの準備」を参照してください。
6. [ステップ **4]** セクションで、[参照してアップロード **(Browse and upload)]** をクリックして、ラップされたキーをローカルデバイスからアップロードします。
   キーがアップロードされると、[以下のファイルは正常にアップロードされました **(Uploaded below file successfully)]** の下にキーが表示されます。キーを再アップロードする必要がある場合は、[ファイルを削除] をクリックし、前の手順で説明したようにキーを再度アップロードします。
7. [キーのローテーション] をクリックして、再供給を完了します。

クォーラムコントロールポリシー

クォーラムコントロールポリシーでは、選択された承認者の合計数が顧客管理キーの取り消しのクォーラムに到達するために必要な承認の最小数を指定します。

⚠ **警告:** キーの取り消し機能をアクティブ化するには、法令による補遺に署名する必要があります。[クォーラムコントロールポリシー設定 (Quorum Control Policy Settings)] オプションは、取り消し機能が有効の場合に利用可能になります。無効の場合、モジュールがアプリメニューに表示されません。キーの取り消し後、暗号化キーが再供給されて再びアクティブになるまで、インスタンスは使用できなくなります。

状況によっては、キーの取り消しを作成できます。まず、カスタマーサービス & サポート からキーの取り消し機能を要求する必要があります。

クォーラムコントロールポリシーでは、選択された承認者の合計数が顧客管理キーの取り消しのクォーラムに到達するために必要な承認の最小数を指定します。たとえば、承認者が合計で 5 人いるのに、クォーラムに到達するために必要な承認が 4 回のみだとします。承認が 4 回得られると、取り消し要求が処理され、キーが取り消されます。

クォーラムの承認が有効なグループで取り消し操作が実行されるたびに、クォーラム承認のワークフローがトリガーされます。承認を出せるすべてのユーザーにメール通知が送信されます。タスクは承認者のアカウントでも生成され、承認者が ServiceNow アカウントにログインするとダッシュボードに表示されます。

このユーザーは、インスタンス、メール、または [キー管理操作] ページから承認を出せます。キーの取り消し操作はクォーラムに到達するまでブロックされます。

承認者の最小数に達すると、クォーラムに到達してキーの取り消しがトリガーされます。取り消しが実行され、要求を承認したユーザーの名前を含めてログに記録されます。

セットアップの詳細については、「クォーラムコントロールポリシー設定の構成」を参照してください。

クォーラムコントロールポリシー設定の構成

クォーラムコントロールポリシー設定を構成するには、次の手順を実行します。

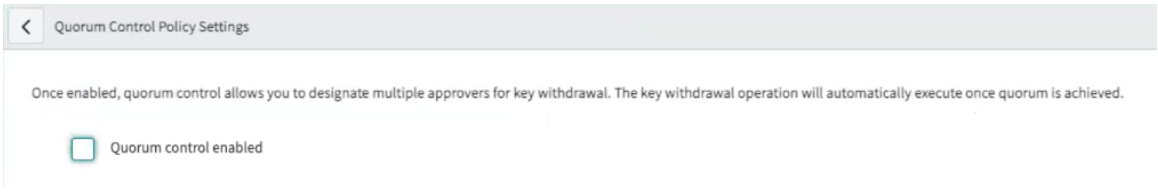
始める前に
必要なロール：sn_kmf.admin

このタスクについて

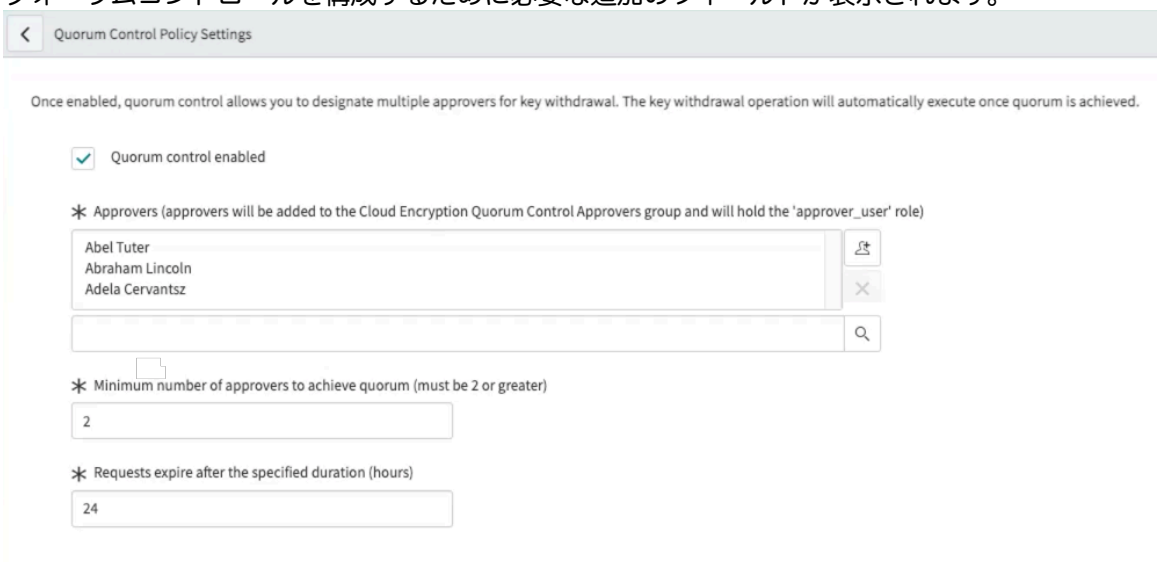
⚠ **警告:** キーの取り消し機能をアクティブ化するには、法令による補遺に署名する必要があります。[クォーラムコントロールポリシー設定 (Quorum Control Policy Settings)] オプションは、取り消し機能が有効の場合に利用可能になります。無効の場合、モジュールがアプリメニューに表示されません。キーの取り消し後、暗号化キーが再度アクティブになるまでインスタンスは使用できなくなります。

手順

1. カスタマーサービス & サポート からキーの取り消し機能を要求します。
2. 移動先 クラウド暗号化キー管理 > クォーラムコントロールポリシー設定.
3. [クォーラムコントロールを有効にする **(Quorum control enabled)]** チェック ボックスをオンにします。



クォーラムコントロールを構成するために必要な追加のフィールドが表示されます。



4. フォームのフィールドをすべて入力します。

フィールド	説明
承認者	ユーザーのリストからクォーラムのメンバーを指定します。ロックアイコン を選択して、ユーザーディレクトリを開きます。選択できる承認者の数に制限はありません。
クォーラムを達成するための承認者の最小数 (Minimum number of approvers to achieve quorum)	クォーラムを達成するために必要な承認者の最小数を指定します。たとえば、9 人の承認者が選択されている場合、最小数として 5 人をクォーラムに設定できます。システムで 5 回の承認を受け取ると、クォーラムに達して取り消し操作が開始されます。 注: 必要な承認者の最小数は 2 人です。
指定した期間 (時間) 後に要求が期限切れになる (Requests expire after the specified duration (hours))	最小数の承認を取得するための最大割り当て時間を、時間単位の数値で設定します。期間が終了すると、クォーラム要求も期限切れになります。取り消し要求を続行するには、新しいクォーラム要求が必要です。

5. [送信] をクリックします。
   確認メッセージが表示されます。

次のタスク
取り消しのアクションは キー管理操作 で実行可能です。

クォーラムコントロールを管理する

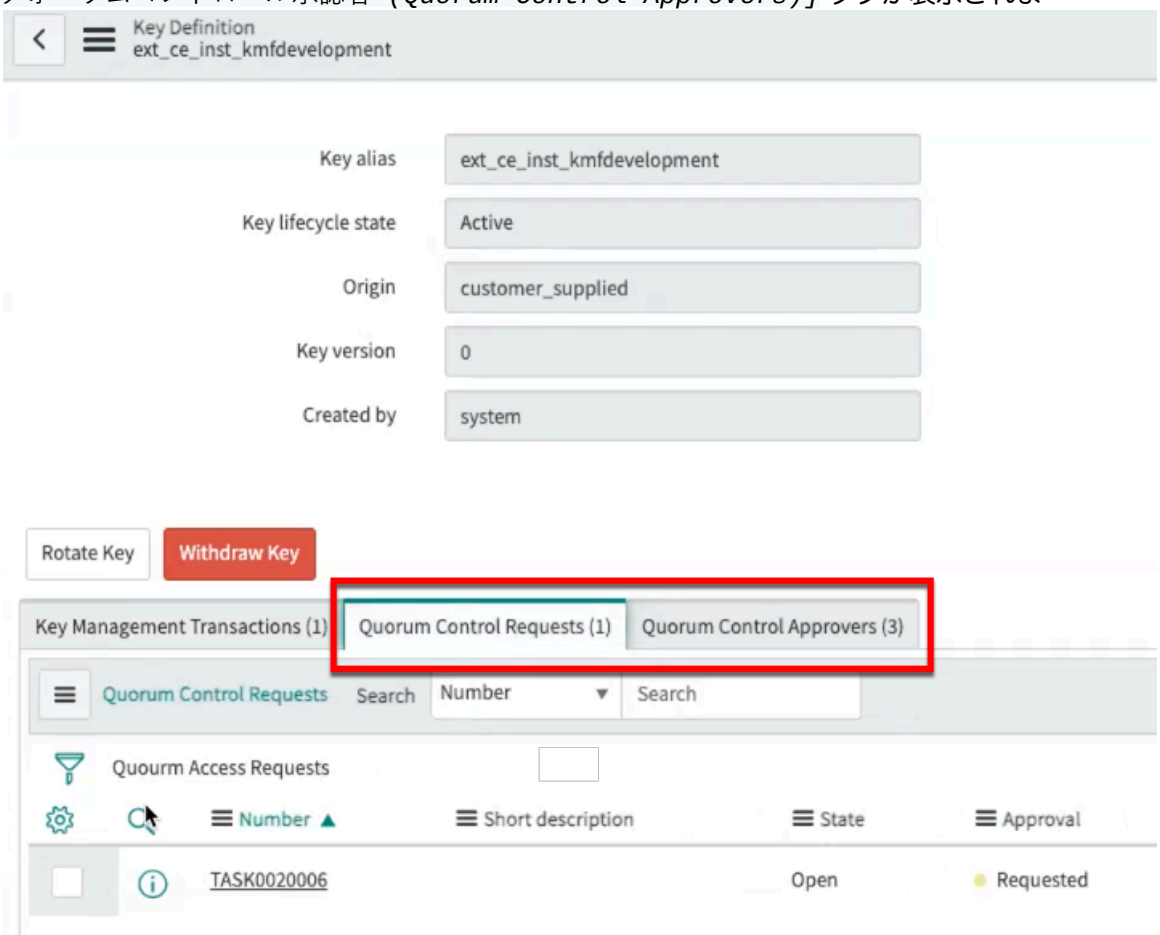
取り消し操作のワークフローがトリガーされた後に、[キー管理操作] ページからクォーラムアクションを管理できます。キーの取り消し操作はクォーラムに到達するまでブロックされます。

始める前に
必要なロール：sn_kmf.admin または sn_kmf.cryptographic_manager

クォーラムが承認または却下されると、クォーラムが達成または却下されたことを通知するメールがキー取り消しの要求者に届きます。

手順

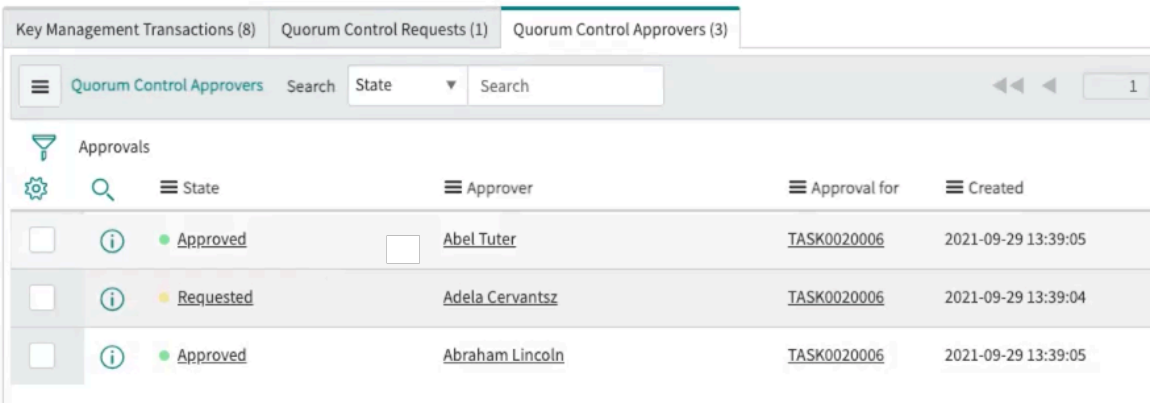
1. キー管理操作 で見つかった顧客管理キーを取り消す手順を実行します。
2. アクティブ化した [クォーラムコントロールの要求 *(Quorum Control Requests)*] および [クォーラムコントロール承認者 *(Quorum Control Approvers)*] タブが表示されます。



3. ユーザー クォーラムコントロール要求 作成された実際の要求を表示するタブ。

- ステータス：
  - オープン：キーの取り消しアクションによってクォーラムの到達が処理待ちになっています。
  - 完了してクローズ：クォーラムに到達しているため、この特定のクォーラム要求に対してこれ以上のアクションを実行できません。
- 承認：
  - 要求済み：承認メールが送信され、クォーラムに到達するためのワークフローがトリガーされました。
  - 承認済み：キーが取り消され、インスタンスがシャットダウンされます。
  - 拒否：クォーラム要求はキャンセルされ、この要求に対してこれ以上のアクションは実行されません。キーを取り消すには、新しい取り消し要求が必要となります。

4. [クォーラムコントロール承認者 (Quorum Control Approvers)] タブを開いて、承認者のリストと承認要求のステータスを表示します。



ステータス：

- 要求済み：承認者がまだ承認要求に対してアクションを実行していません。
- 承認済み：要求がメールまたは承認ページから承認されました。

5. [キー管理トランザクション] タブを選択して、キーの取り消し要求ステップの進捗を表示します。

- ステップ 0 - クォーラム要求：実際のクォーラム要求。キーの取り消しの手順をトリガーするには、クォーラム要求を完了する必要があります。
- ステップ 1 - キーの取り消し：キーの取り消しのステップ。これはステップ 2 ～ 7 で構成されています。
- ステップ 2 - Request_preparation：トリガーする要求と、ラッピングおよびローテーションを作成します。
- ステップ 3 - request_integrity_check：要求が正当で安全であることを検証します。
- ステップ 4 - request_validation：進行中の要求があり、一度に 1 つのローテーション要求のみが処理できることを検証します。
- ステップ 5 - hsm_key_delete：KeySecure を呼び出してアクティブなキーを削除します。
- ステップ 6 - key_metadata_withdraw：アクティブなキーのメタデータのライフサイクル状況を「破棄 (destroyed)」に変換します。
- ステップ 7 - post_withdraw：インスタンスをシャットダウンする呼び出しを行います。

クォーラムコントロール要求の承認または拒否

キー管理トランザクションからのクォーラムコントロール要求を承認または拒否します。

このタスクについて
クォーラム要求が作成されると、メンバーが必要とする承認の数が最小限になります。取り消し操作のワークフローがトリガーされた後は、いくつかの手法を用いてクォーラムアクションを管理できます。ユーザーは、[キー管理操作] ページの [インスタンス] にある [自分の承認 (My Approvals)] から、または要求メールから直接承認を出すことが可能です。キーの取り消し操作はクォーラムに到達するまでブロックされます。

この手順では、[キー管理操作] ページからクォーラム要求を承認または拒否する方法について説明します。

始める前に
必要なロール：sn_kmf.admin または sn_kmf.cryptographic_manager

手順

1. 移動先 すべて > クラウド暗号化キー管理 > キー管理トランザクション > クォーラムコントロール承認者.
2. テーブルからユーザー名を選択します。
3. 要求を承認または拒否します。

クォーラム要求の承認または拒否

クォーラム要求が作成されると、メンバーが必要とする承認の数が最小限になります。取り消し操作のワークフローがトリガーされた後は、いくつかの手法を用いてクォーラムアクションを管理できます。ユーザーは、[キー管理操作] ページの [インスタンス] にある [自分の承認 **(My Approvals)]** から、または要求メールから直接承認を出すことが可能です。キーの取り消し操作はクォーラムに到達するまでブロックされます。

始める前に
必要なロール：sn_kmf.admin または sn_kmf.cryptographic_manager

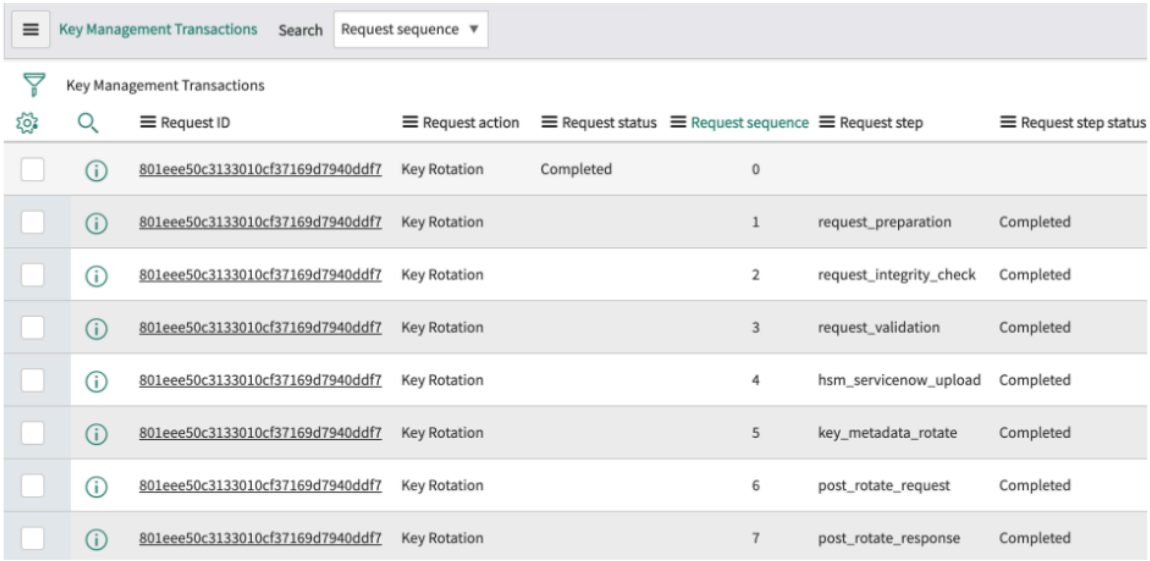
キー管理トランザクション

キー管理トランザクションサブモジュールには、ServiceNow インスタンス内のキーに対して発生したすべてのトランザクションが表示されます。

- キートランザクションは以下により定義されます。
  - 複数の要求ステップで構成されます。
  - 1 つの要求 *ID* がすべての要求ステップで共有されます。
  - トランザクションの最初のステップ (要求シーケンス 0) では、トランザクション全体の現在のステータスが提供されます。

    次の画像で示されているように、最初のステップ 0 の全体的な要求ステータスは [完了] です。
- 個々の要求ステップで、トランザクションに関して以下を識別できます。
  - トランザクションの各ステップの順序は、ステップのシーケンス番号で識別できます。
  - 各トランザクションのステータスは、要求ステップのステータスから確認できます。
  - 最初のステップ以降のステップで失敗すると、トランザクション全体のステータスが [失敗] になります。すべてのステップが完了すると、トランザクションのステータスも [完了] になります。

次の画面は、ServiceNow キーのローテーションで表示される情報のタイプ例です。



次の表は、[キー管理トランザクション] ページで利用可能なフィールド情報を示しています。

キー管理トランザクション

フィールド	説明
要求 ID	実行中のアクションに対してシステムで生成された一意の ID (1 つの要求 ID) が、すべての要求ステップで共有されます。
要求アクション	実行中のキー操作のアクションが表示されます。
要求ステータス	• 処理中：要求が入力されましたが、まだ完了していません。 • 完了：要求が正常に完了しました。 • 失敗：問題が発生し、プロセスが完了していません。 **注:** カスタマーサービス & サポート に連絡し、エラーが発生した要求番号を提供してください。
キーのエイリアス	英数字のエントリ
キーのライフサイクル状況	定義については、「キー管理フレームワークの主要なライフサイクル状況」を参照してください。
作成元	• ServiceNow キー • 顧客管理キー
キーバージョン	キーがローテーションされると、バージョン番号がインクリメントされます。
要求シーケンス	システム内で要求が処理されている順序が表示されます。
要求ステップ	キーローテーション中にステップが処理されているかどうかが表示されます。ステップの数と内容は、実行されたキー操作のタイプによって異なります。

キー管理トランザクション (続く)

フィールド	説明
	1. request_preparation：トリガーする要求と、ラッピングおよびローテーションを作成します。 2. request_integrity_check：要求が正当で安全であることを検証します。 3. request_validation：進行中の要求があり、一度に 1 つのローテーション要求のみが処理できることを検証します。 4. attachment_process：添付ファイルからラップされたキーマテリアルを抽出します(顧客管理キーをローテーションする場合の追加ステップ)。 5. hsm_<キーのタイプ> _upload：ラップされたキーマテリアルを HSM、KeySecure にアップロードします。 6. key_metadata_rotate：新しいキーメタデータを生成します。 7. post_rotate_request：キーローテーションを実行する要求を送信します。 8. post_rotate_response：顧客インスタンスからの要求に基づいてキーローテーションを実行するための応答。 **注:** 要求ステップが完了しない場合にステータスの進行を分析するために、カスタマーサービス & サポートに要求ステップを提供します。
要求ステップのステータス	• 完了：ローテーションに成功しました。 • 失敗：ローテーションに失敗しました。 **注:** 要求ステップが完了しない場合にステータスの進行を分析するために、カスタマーサービス & サポートに要求ステップを提供します。

クラウド暗号化のログ記録

クラウド暗号化 のログ記録オプションについて説明します。

クラウド暗号化のログ記録のテーブル

以下のテーブルを使用して、インスタンスでのクラウド暗号化のトランザクションに関連するログ記録情報を検索します。

テーブル	説明
クラウド暗号化のメタデータ (Cloud Encryption Metadata) [dare_key_metadata]	クラウド暗号化のメタデータ (Cloud Encryption Metadata) は、ライフサイクル管理の重要なメタデータをキャプチャします。このテーブルでは、キーのライフサイクル、ステータス、およびバージョン情報を確認できます。このテーブルはキーを操作するたびに更新されます。
キー管理トランザクション [dare_key_request]	キー管理トランザクションは、キー管理トランザクションの情報をキャプチャします。このテーブルでは、トランザクションの各ステップのログ記録を確認できます。このテーブルでは

テーブル	説明
	トランザクションのエラー情報が [エラーメッセージ] フィールドに記録されます。
システム監査 [sys_audit]	[システム監査] テーブルでは、インスタンスで行われたすべての監査済みレコードに対する挿入と更新をキャプチャします。このテーブルでは、インスタンス上のレコードに対する変更、変更が行われた日時、および変更を開始したユーザーアカウントを確認できます。

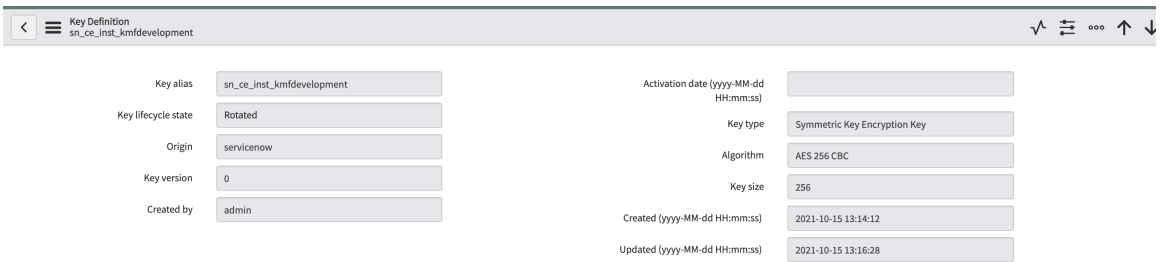
キーローテーション操作の監視

Cloud Encryption のキーメタデータ [dare_key_metadata] テーブルを使用して、キーのライフサイクルに関する情報を検索します。このテーブルでは、キーの作成元、アクティブ化日、ステータス、バージョンなどの情報を確認できます。

キー管理トランザクション [dare_key_request] テーブルを使用して、キー操作のトランザクションを監視します。このテーブルでは、進捗状況やステータス、また要求がプロセスのどのステップであるかなど、キーに関連するすべての要求を検索できます。完了した要求は、[完了] ステータスでこのテーブルに保持されます。

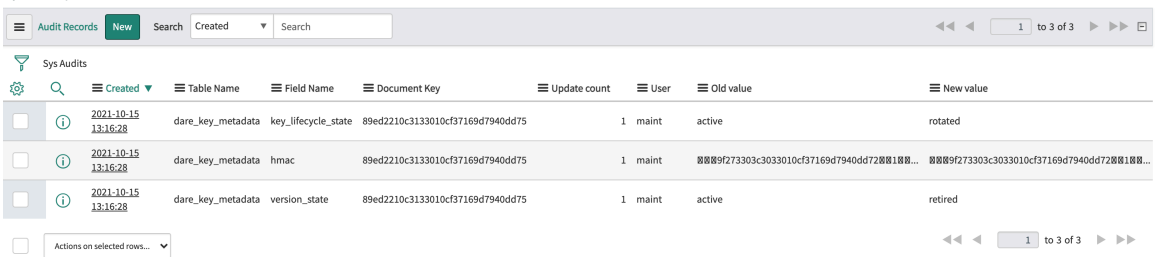
この例は、キーローテーションの操作を示しています。この操作中、古いキーのライフサイクル状況がアクティブからローテーション済みに更新され、バージョンのステータスがアクティブから廃止に更新されます。

ローテーションされたキーのキー定義



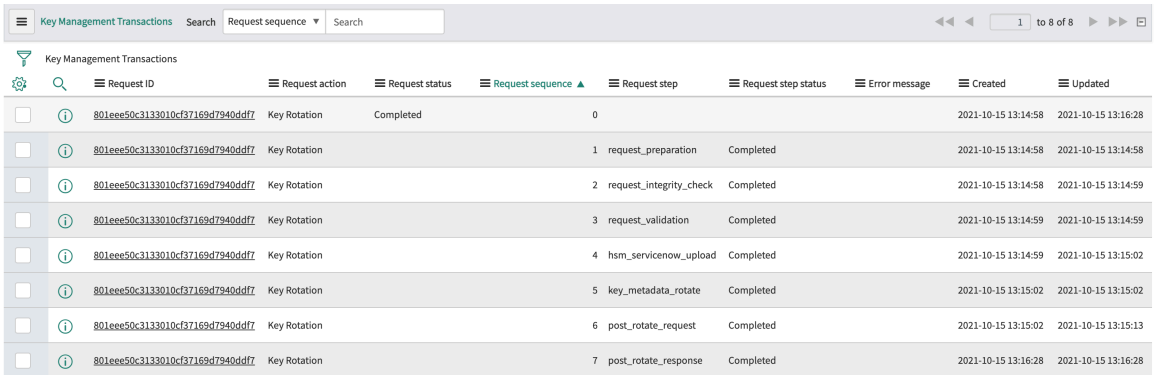
システム監査 [sys_audit] テーブルで、アドミニストレーターはクラウド暗号化のキーメタデータ [dare_key_metadata] テーブルのレコードに加えられた変更を確認できます。アドミニストレーターはどのレコードがいつ更新されたかを確認できます。ログエントリには、変更されたフィールドおよび、古い値と新しい値も記録されます。

取り消されたキーの監査ログ



アドミニストレーターは、クラウド暗号化のキーメタデータ [dare_key_metadata] テーブルにあるレコードを確認できます。以下の監査レコードでは、要求のステータスが処理中から完了に変更されています。

取り消されたキーの監査ログ

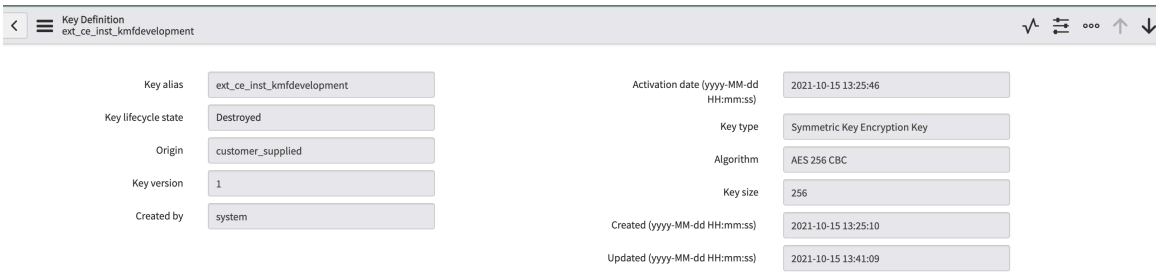


キーの取り消し操作のログ記録

キーの取り消しに関するログ記録情報は、監査 [sys_audit] テーブルに格納されます。このログ記録情報には、キーの取り消しを開始したユーザーと取り消しが行われた日時に関する情報が含まれています。

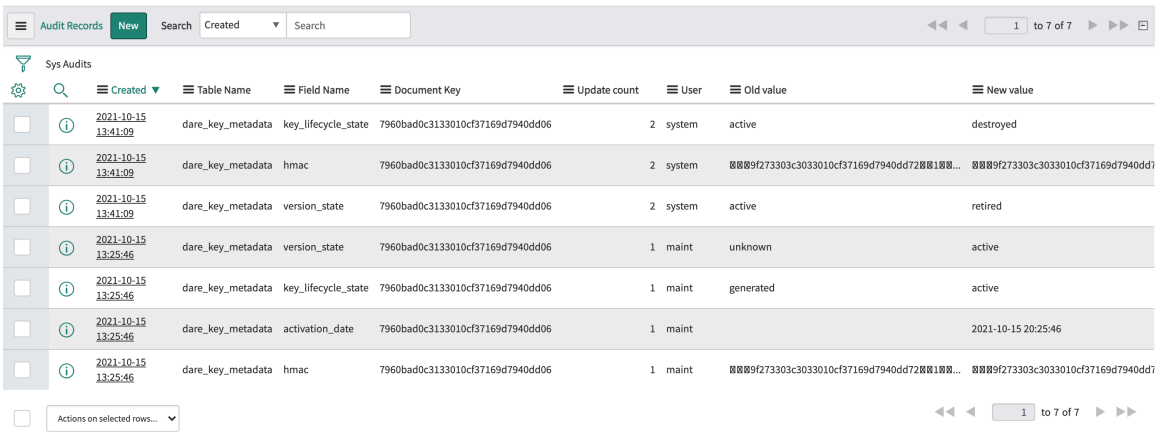
この例は、キーの取り消しの操作を示しています。この操作中に、キーのライフサイクルのステータスが [生成済み] から [アクティブ (active)] や [破棄 (destroyed)] に更新されます。鍵のバージョンが [不明] から [アクティブ] や [廃止] に更新されます。

取り消されたキーのキー定義



システム監査 [sys_audit] テーブルで、アドミニストレーターはクラウド暗号化のキーメタデータ [dare_key_metadata] テーブルの変更内容を確認できます。

取り消されたキーの監査ログ



改ざん検出

改ざん検出を使用して、クォーラムコントロールの設定に対する不正な変更を検出することでセキュリティを向上させます。

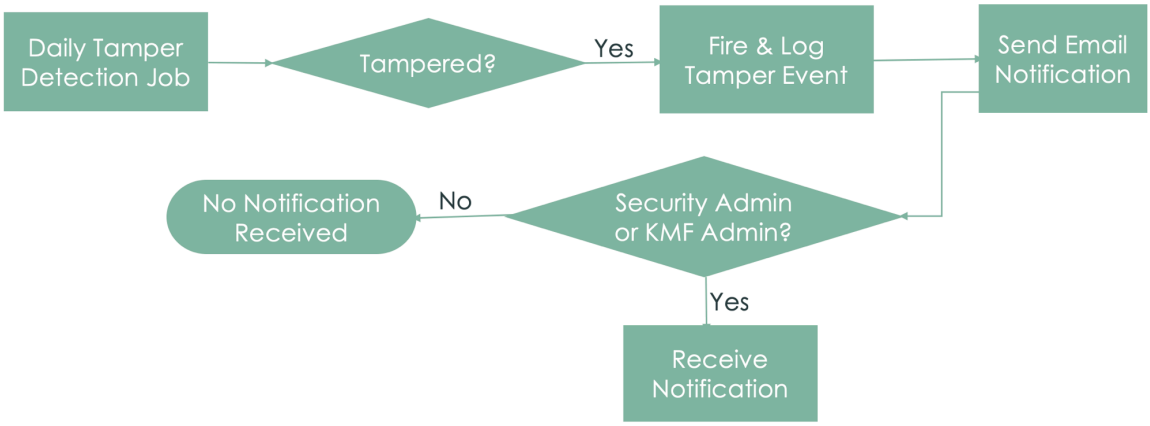
改ざん検出のプロセス

改ざん検出を有効にすると、不正な変更 (改ざん) がないかチェックすることで、クォーラムコントロール設定が検証されます。改ざん検出では、Hash-based Message Authentication Code (HMAC) を使用します。

1. 設定が変更または作成されると、インスタンスが HMAC を作成します。HMAC は設定 (dare_property) レコードの値に基づいて作成されます。
2. インスタンスでこれらの設定が使用されるたび、改ざん検出が HMAC を使用して設定を検証します。
3. 正常に検証された設定はプラットフォームで使用できます。検証されない場合は使用できません。

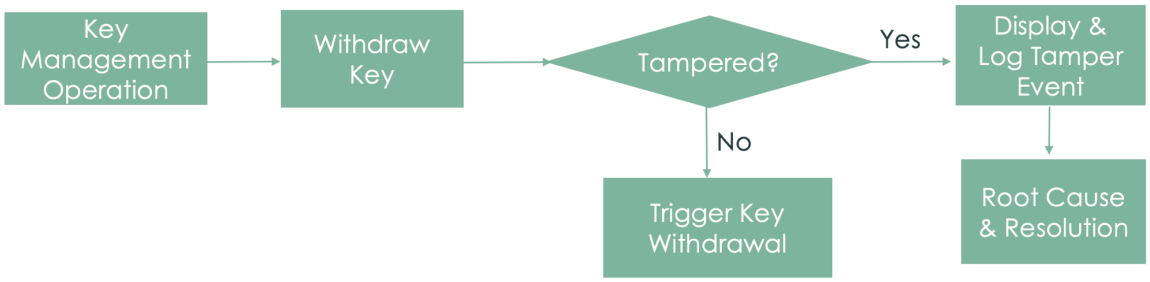
改ざん検出はインスタンスで毎日実行される

改ざん検出は、日次スケジュール済みジョブを使用して改ざんの設定をチェックし、検証が失敗した場合はノードログとセキュリティログに報告します。改ざん検出は、検証エラーに関する通知をセキュリティアドミニストレーターと KMF アドミニストレーターに送信します。



キーの取り消しの実行前に改ざん検出を実行する

改ざん検出では、キーの取り消しを要求するときにもプロパティを検証します。設定が検証に合格しないと、キーの取り消しは実行されません。この場合、キーの取り消しが競合しないうちに検証の問題を解決する必要があります。



改ざんの識別

検証に失敗すると、改ざん検出によってログが更新されます。

改ざん検出でクォーラムコントロール設定の検証に失敗すると、その失敗がノードログとセキュリティログに表示されます。ログのエントリーには、検証に失敗した設定 (dare_property) レコードの sys_id が含まれています。

```
2022-06-28 13:45:46 (582) Default-thread-5
 B6FAC1F6C3D01110CF37169D7940DD6E txid=231c4d72c310 SEVERE
```

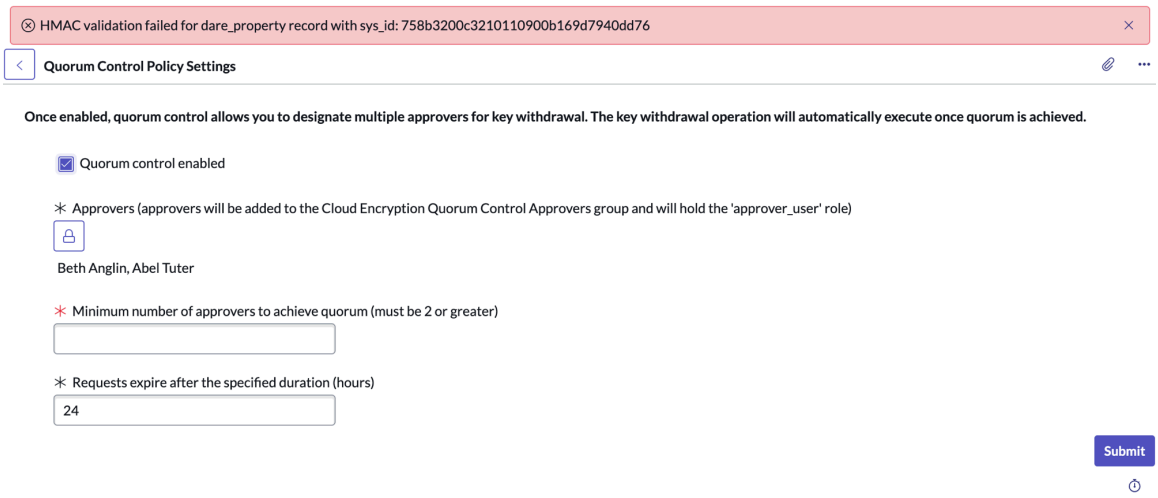
```
HMAC_VALIDATION_FAILED:The dare_property record with sys_id:
776e3200c3210110900b169d7940dd76 failed HMAC validation

2022-06-28 13:47:35 (264) Default-thread-8
B6FAC1F6C3D01110CF37169D7940DD6E txid=8e8cc972c310 SEVERE
HMAC_VALIDATION_FAILED:The dare_property record with sys_id:
758b3200c3210110900b169d7940dd76 failed HMAC validation
```

検証に失敗すると、この例のような情報がログ記録に表示されます。成功した検証はログに表示されません。

改ざん検出によりクォーラムコントロール設定ページに警告メッセージが表示される

クォーラムコントロール設定の検証に失敗した場合、インスタンスで [クォーラムコントロールポリシー設定 (Quorum Control Policy Settings)] ページを表示すると警告が表示されます。警告には、検証に失敗した設定 (dare_property) レコードの sys_id が含まれています。



改ざん検出は *Security Admin* および *KMF Admin* ロールのユーザーに通知を送信する

改ざん検出でクォーラムコントロール設定の検証に失敗した場合、セキュリティアドミニストレーターと KMF アドミニストレーターにこの例のような通知が届きます。

Subject: IMPORTANT: Cloud Encryption Properties Tampering Detected

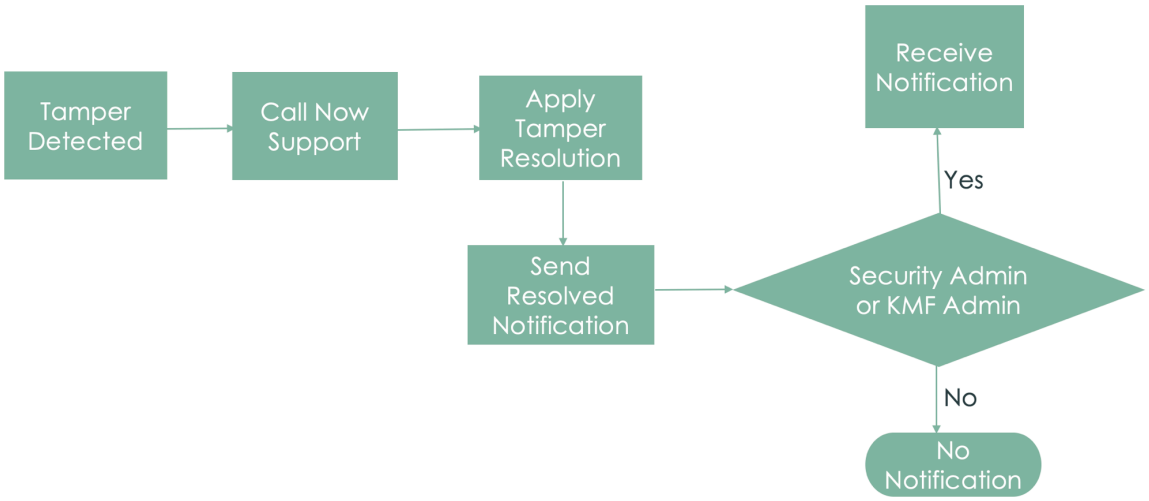
The following list of Cloud Encryption properties appear to have been tampered.
Contact ServiceNow support for additional information and how to correct this issue.

Tampered properties:
com.glide.dare_key_mgmt.quorum.approvers(903a3212c3210110cf37169d7940dda8)

Ref:MSG0000109_q5pzYMS7hcgHXAoUhipc

ServiceNow サポートによる改ざんの問題の解決

重要: 改ざん検出による検証の失敗は、ServiceNow サポートの支援がなければ解決できません。



改ざん検出でクォーラムコントロールの設定の検証に失敗した場合は、ServiceNow サポートに連絡して問題を解決してください。サポートエージェントが検証の失敗を解決すると、問題が解決されたことを示す通知がセキュリティアドミニストレーターと KMF アドミニストレーターに届きます。

Subject: IMPORTANT: Cloud Encryption Properties Reset

At your request, ServiceNow support has reset the following list of Cloud Encryption properties to their default values.
If you did not request this action, contact ServiceNow support immediately.

Reset properties:
com.glide.dare_key_mgmt.quorum.approvers(903a3212c3210110cf37169d7940dda8)

Ref:MSG0000116_0HAz8U1r59OmppqV8ysg

フルディスク暗号化

データベースサーバーは顧客データを格納する唯一のコンポーネントであるため、フルディスク暗号化 (FDE) は、データベースサーバー内のストレージシステム全体にのみ適用されます。FDE は、ストレージデバイスの物理的な紛失または盗難からのみ保護します。暗号化されたディスクサーバーの電源がオンで、データが提供されている場合、暗号化で追加の保護は提供されません。

フルディスク暗号化

フルディスク暗号化が、厳しい規制のある組織と関係がある場合があり、顧客の ServiceNow 展開に多大なコストがかかる可能性があります。ストレージデバイスの紛失や盗難を軽減するために ServiceNow によって実施されている対策も、選択の要因になる場合があります。

ServiceNow AI Platform の観点から、すべてのデータが復号化されます。

商用環境では、FIPS 140 検証済みハードウェアまたは検証中のストレージデバイスと、追加料金が必要な ServiceNow 専用ハードウェアオプションを使用できます。FDE はハードウェア自体に適用されるため、割り当てられたすべてのインスタンスに格納されているすべてのデータが暗号化されます。

FDE と専用ハードウェアオプションの選択の詳細については、ServiceNow の担当者にお問い合わせください。

エッジ暗号化

ServiceNow® エッジ暗号化™ は、社内の機密データを暗号化 (送信中に暗号化) してから、インターネット経由で ServiceNow インスタンスに送信し、暗号化はそのまま維持されます。

重要:

エッジ暗号化は、2024 年 11 月の時点で販売終了ステータスにあり、Yokohama リリースの時点で更新終了フェーズにあります。

ServiceNow Platform Encryption は、今後推奨されるソリューションです。詳細については、「 エッジ暗号化の更新の終了 」を参照してください。

探索

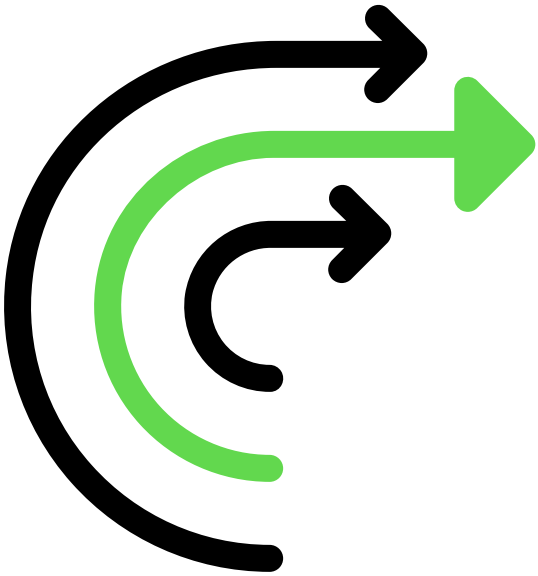
エッジ暗号化の主な機能とビジネス価値について説明します。

計画

エッジ暗号化の計画方法を把握します。

自動翻訳

インストール



エッジ暗号化のインストール

エッジ暗号化のアップグレード

アップグレード

エッジ暗号化のアップグレード方法を把握します。

構成

エッジ暗号化の設定方法を把握します。

自動翻訳

エッジ暗号化 の詳細

エッジ暗号化 は、ネットワークに常駐するネットワーク暗号化システムで、データセンターと ServiceNow クラウド間を移動する機密データを暗号化および復号化します。

エッジ暗号化 の概要

「クライアント側」暗号化とも呼ばれる Edge は、インフラストラクチャ上で維持されているプロキシを通過するために、すべての双方向ユーザートラフィックを必要とします。キーはインフラストラクチャ上のプロキシ内に保存されるため、キー管理を完全に制御できます。ServiceNow AI Platform は、暗号テキストを復号化してキーにアクセスすることはできません。

エッジ暗号化 機能は、データのエンドツーエンドの暗号化およびキー管理を制御する機能を提供する追加コストオプションです。エッジ暗号化 は、ServiceNow によって提供され、自身のネットワーク内にインストールされたプロキシアプリケーションを使用します。このプロキシアプリケーションは、指定されたデータパターンをトークン化するか、文字列フィールド、日付フィールド、日付/時刻フィールド、および添付ファイルデータを暗号化してから、環境からインスタンスに送信します。プロキシアプリケーションは、自分のネットワーク内のみに保存されているキーを使用して、同じデータを自分のネットワーク内でのみ復号化します。

関連する暗号化キーおよび構成は、ネットワーク内の Edge プロキシにのみ存在し、ServiceNow には表示されません。データは環境を離れた瞬間から暗号化され、取得時にのみ復号化されます。ServiceNow システムまたは担当者であってもプレーンテキストでデータにアクセスすることはできません。

エッジ暗号化 のユーザー

ネットワークのプロキシサーバーを介してインスタンスにログインしたユーザーだけが、暗号化されたデータをクリアテキストで表示できます。同様に、ネットワークのプロキシサーバーを介してインスタンスにログインした security_admin ユーザーだけが、エッジ暗号化 を設定および管理できます。

プロキシサーバーはネットワークに常駐しているため、ユーザーが暗号化キーを所有し、管理します。暗号化キーがインスタンスに送信されることは決してありません。そのため、ServiceNow に機密データがクリアテキストで表示されることは決してありません。

ユーザーは Edge プロキシの設定およびルールの管理に加えて、Edge プロキシを有効にしてサポートするために、環境内でサーバーを運用するための通常の要件 (ホスティング、ルーティング、バックアップ、DNS 構成などを含む) を担当します。

暗号化とトークン化

エッジ暗号化 は、機密情報を保護する手段として暗号化 (暗号化設定を使用) とトークン化 (暗号化パターンを使用) の両方をサポートしています。

暗号化設定

暗号化設定を使用して個々のフィールドを暗号化することができます。エッジ暗号化は、AES 128 ビットと AES 256 ビットの暗号化キーをサポートしています。エッジ暗号化 は、標準、等価性保存、および順序保存の暗号化タイプをサポートしています。

添付ファイルに加えて、以下のフィールドタイプを暗号化できます。

- 日付
- メール
- 日付/時刻
- HTML

- IP アドレス
- ジャーナル
- ジャーナル入力
- 複数行テキスト
- 1 行テキスト
- 文字列
- URL

注:

暗号化対象としてマークされたジャーナル フィールドがアクティビティストリームに追加された場合、フィールドへのすべてのユーザー入力がアクティビティストリームで暗号化されます。

サポートされているフィールド タイプ内のマルチバイト文字を暗号化できます。

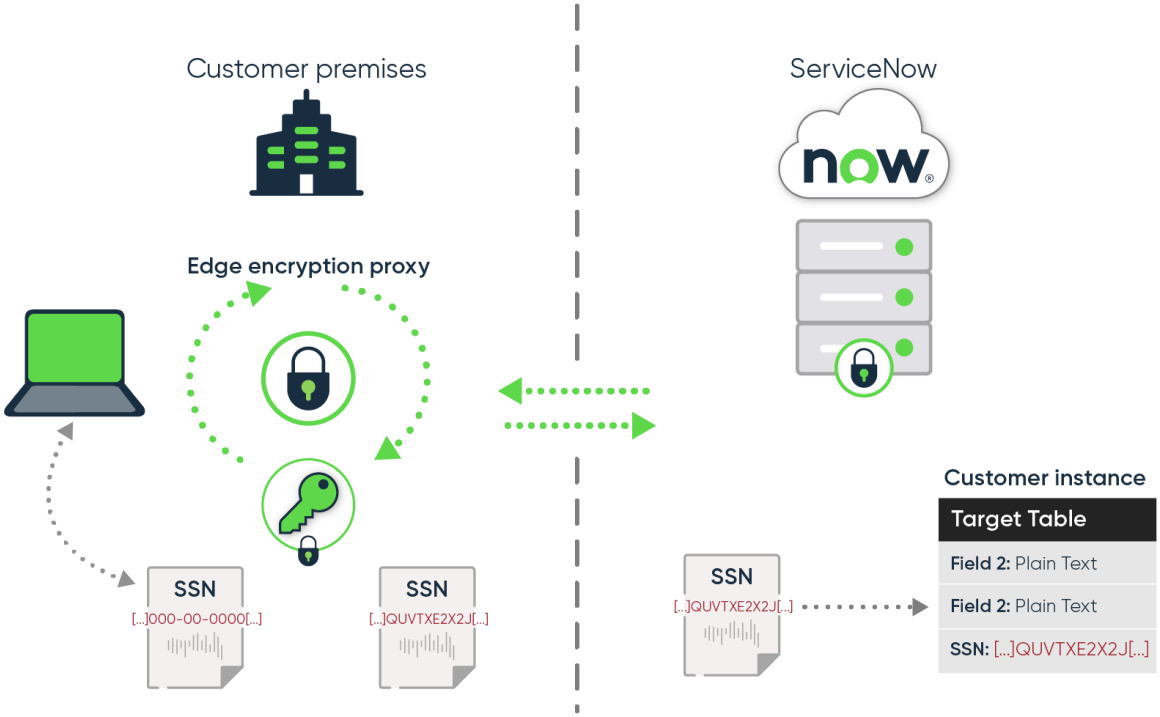
次のサービスカタログ変数タイプを暗号化することもできます。

- 文字列タイプ
  - 1 行テキスト
  - 複数行テキスト
  - 幅広 1 行テキスト
- 日付
- 日付/時刻
- URL
- メール
- HTML
- IP アドレス

暗号化パターン

暗号化パターンを使用して、社会保障番号やクレジット カード番号などの規則的なパターンに一致する文字列をトークン化することができます。暗号化の主要な手段としては暗号化設定をお勧めしますが、暗号化フィールド以外の場所にある機密情報を保護する場合は暗号化パターンを補足手段として使用してください。

注: エッジ暗号化 プロキシサーバーでネットワークにある MySQL データベースが必要になるのは、順序保存の暗号化または暗号化パターンを使用している場合のみです。クリアー テキスト値は、ネットワークのプロキシ データベースに格納されます。このため、プロキシ データベースを保護し、定期的にバックアップすることが重要です。推奨事項については、「エッジ暗号化のコンポーネント」を参照してください。



ServiceNow AI Platform 上の エッジ暗号化

エッジ暗号化 は、ブラウザと ServiceNow インスタンス間のゲートウェイとして機能します。ブラウザからのトラフィックは、途中のゲートウェイを経由して ServiceNow インスタンスに達します。さらにゲートウェイは、暗号化対象としてマークされた送信データを暗号化するように設定されます。受信トラフィックはゲートウェイ経由で復号化され、エンド ユーザーのブラウザにはクリアーテキストが表示されます。セキュリティ制御の観点から見た場合、この実装には、暗号化とキー管理が ServiceNow の外部で処理されるという利点があります。

長所と短所

フィールド暗号化エンタープライズ や フィールド暗号化 と同様に、エッジ暗号化 では、追加のセキュリティの結果として、インスタンス内にいくつかの機能上の制限が課せられます。ただし、ローカル Edge プロキシは、列レベルの暗号化と比較した場合、ソートに関連するいくつかの追加機能も提供します。

長所：

- エッジ暗号化 は、情報の閲覧者を完全管理し、データ侵害を防止します。
- 情報はプロキシサーバーに残り、暗号化されないままネットワークから離れることはありません。
- 情報は ServiceNow インスタンスに到達する前に、転送中に暗号化されます。
- あなたが、すべての暗号化キーを保持して管理します。ServiceNow 担当者であっても、他者があなたの鍵にアクセスすることはできません。
- 暗号化アルゴリズムの強度は、AES-128 または AES-256 から選択できます。
- エッジ暗号化 には、文字列テキスト、日付フィールドと日付/時刻フィールド、添付ファイル、URL、およびジャーナルを暗号化する機能が含まれています。
- エッジ暗号化 では、データベースおよびインスタンス内に保存されているデータの標準、等価性保存、および順序保存の暗号化が可能です。

自動翻訳

- 暗号化ルールを使用すると、暗号化する対象とその暗号化された情報をインスタンス内のどこに配置するかをプロキシサーバーに明確に指示するカスタムスクリプトを作成できます。これらのスクリプトは、データ構造が ServiceNow インスタンスと正確に一致しない場合に便利です。
- 暗号化パターンを使用すると、パスワードなどの情報をトークン化できます。

短所：

- エッジ暗号化 には、Edge プロキシクラスターを介した追加のネットワークホップと追加の処理が必要であり、トラフィックに遅延が生じる可能性があります。追加された エッジ暗号化 アプリケーションの処理遅延は、ネットワークホップに比べてごくわずかです。
- 独自の暗号化キーを管理するのは、複雑で時間がかかる場合があります。
- 最大 2 つのキーを維持できますが、列/データの異なるサブセットや異なるロールなどに対して異なるキーを柔軟に定義することはできません。
- エッジ暗号化 には、サーバーまたはプラットフォームがデータを復号化して、復号化されたデータを操作できないという副作用があります。その結果、エッジ暗号化で列を暗号化するときに、ServiceNow AI Platform の機能とデータ処理が制限される可能性があります。

使用前の注意事項

暗号化とトークン化はデータの性質を変える処理であるため、エッジ暗号化 は他のインスタンス プロセスに影響を与える可能性があります。エッジ暗号化 の使用にあたっては、インスタンスに及ぼす影響を慎重に考慮してください。

プロキシサーバーはネットワークにインストールされて維持されるため、エッジ暗号化 ではネットワークの管理が必要になります。円滑な実装ができるように、ネットワーク要件を確認してください。

次のトピックを確認し、インスタンスにおけるエッジ暗号化の影響を理解してください。

- エッジ暗号化の計画立案
- エッジ暗号化 のシステム要件
- エッジ暗号化環境のサイジング
- 順序保存とトークン化のデータベースのサイズを計算する
- エッジ暗号化 の制限事項
- エッジ暗号化のキー管理

エッジ暗号化のコンポーネント

エッジ暗号化 は、 ネットワーク内のサーバー上で実行される エッジ暗号化 プロキシサーバーと、エッジ暗号化 プラグインで構成されます。プラグインは ServiceNow インスタンスにインストールする必要があります。順序保存の暗号化タイプまたは暗号化パターンを使用する場合は、プロキシ データベースもネットワークにインストールする必要があります。

プロキシアプリケーション

エッジ暗号化 プロキシサーバーを通過する際、 エッジ暗号化 プラグインを使用して、どのフィールド、パターン、および添付ファイルを暗号化するかを指定できます。また、暗号化ルールを管理して、特定の要求を暗号化したり、一括暗号化ジョブをスケジュールしたりすることもできます。

プロキシサーバー

エッジ暗号化 プロキシサーバーは暗号化ルールを使用して、HTTP 要求の中で何らかのデータがある場合にどれを暗号化する必要があるかを識別し、そのデータを暗号化してから要求をインスタンス

に転送します。復号化では、 エッジ暗号化 プロキシサーバーは、暗号化されたデータが HTTP 応答にないかを調べ、そのデータを復号化してから応答をクライアントに送り返します。この処理を行うためには、すべての HTTP 要求および応答が エッジ暗号化 プロキシサーバーを通過する必要があります。これらの HTTP 要求には、ブラウザから発生した要求や、SOAP 要求、REST 要求などがこれに含まれます。

プロキシデータベース

順序保存の暗号化または暗号化パターンを使用している場合、プロキシサーバーはネットワークにある MySQL データベースを利用します。ネットワーク内のすべてのプロキシサーバーで、同じデータベースを使用する必要があります。

プロキシ データベースには下記のテーブルが含まれています。

プロキシ データベースのテーブル

名前	説明
db_id	一意のデータベース ID
edge_token_map	暗号化パターン データ
token_map	順序保存の暗号化データ

プロキシ データベースのバックアップ

暗号化パターンではトークン化が利用されるため、クリアテキスト値はプロキシデータベースに格納されます。データベースが失われた場合、クリアテキスト値は復元できません。そのため定期的なバックアップが重要になります。データの消失を防ぐため、ServiceNow の推奨事項に従ってプロキシデータベースをバックアップしてください。

- データベースを 24 時間ごとにバックアップします。
- MySQL データベースのバイナリログファイルを少なくとも 2 日間保持します。バックアップの復元が完了したら、バイナリログを使用して、最新のバックアップ以降に失われたデータを再生成します。

エッジ暗号化 クライアント

エッジ暗号化 では 3 つのクライアントを使用して、プロキシがインスタンスを実行中であることをインスタンスに通知したり、プロキシとインスタンスの間の要求を同期したり、潜在的な暗号化後にインスタンスに対するすべてのユーザー要求を転送したりします。

クライアント	説明
ハートビート/キープアライブクライアント	このプロキシが稼働していることをインスタンスに通知できるよう、5 秒間隔で ServiceNow インスタンスに要求を送信することを担当します。要求は、Edge プロキシテーブルの last_response_on フィールドを駆動し、その結果、プロキシの状態を駆動します。システムに要求の送信に関する問題がある場合、または要求や要求処理が遅延する場合、その他のクライアント (ユーザートラフィックに対するものを含む) が稼働している場合でも、インスタンスはプロキシを応答なしとしてマークすることがあります。

クライアント	説明
	このクライアントは、インスタンス上のプロキシのオンラインステータスを制御します。 *edgeencryption.proxy.keepalive.interval* プロパティは、このクライアントのポーリングレートを制御します。デフォルトは 5 (秒) です。
ポーリング/同期クライアント	エッジ暗号化設定 (たとえば、暗号化するテーブル、列、添付ファイルや、キー、ジョブ、ルール、トークン化パターン) に従って同期するよう、プロキシがインスタンスに送信するさまざまな要求を担当します。 *edgeencryption.config.poll.interval* プロパティは、このクライアントのポーリングレートを制御します。 ⚠ **警告:** この設定を変更しないでください。プロキシのポーリング間隔のデフォルト設定を変更すると、インスタンスのエッジ暗号化設定を更新するときに同期が遅延する可能性があります。
デフォルト/ユーザートラフィッククライアント	その他すべてについて、このクライアントはすべてのエンドユーザー要求を処理し、暗号化がある場合はそれを施した後、ServiceNow インスタンスに転送します。このクライアントは、インスタンスからの応答も処理し、復号化がある場合はそれを施した後、エンドユーザーに転送します。

エッジ暗号化のキー管理

エッジ暗号化 で使用する暗号化キーについては、お客様の責任でその提供と管理を行っていただく必要があります。

このトピックでは、エッジ暗号化 製品のキーについて説明します。フィールド暗号化で使用できるキー管理フレームワークに関する情報をお探しの場合は、「キー管理フレームワーク」を参照してください。

エッジ暗号化 で使用する暗号化タイプをサポートする目的で暗号化キーを取得および作成する際は、次の点を考慮してください。

- AES 128 ビットまたは AES 256 ビットのどちらを使用するか。デフォルトの AES 128 ビット暗号化キーは、使用しない場合でも定義する必要があります。
- ファイルシステム、Java KeyStore、または Enterprise Key Management (EKM) のどれを使用するか。
- 暗号化キーをいつローテートするか。
- どのような場合に一括暗号化ジョブを使用して新しいキーでデータを再暗号化するか。

プロキシ設定ファイルやキーストアからキーを削除する場合は、そのキーを使用しているインスタンス上のすべてのデータを事前に復号化しておくことが重要です。それには、新しい暗号化キーを追加して、一括キー ローテーション ジョブをスケジュールします。

キーストア

エッジ暗号化 は、次のタイプのキー ストレージをサポートしています。

ファイルストア

キーは、エッジ暗号化 プロキシからアクセス可能なファイル システム内のファイルに格納されます。ファイルに保存された暗号化キーは暗号化されていないため、これらのファイルの保護についてはユーザーの責任になります。

Java KeyStore

キーは、Java の JCEKS KeyStore に格納されます。Java KeyStore はパスワードで保護されているため、ファイルストア内のファイルにキーを格納する方法よりも安全です。1 つの Java KeyStore に複数のキーを格納することができます。キーはキー別名により識別されるため、複数のキーを簡単に管理できます。

Enterprise Key Management (EKM)

キーは、SafeNet KeySecure または Unbound Technology のキー管理システムを使用して格納および取得されます。

エッジ暗号化 プロキシの keystore ディレクトリーには、keystore.jceks という名前の Java JCEKS KeyStore ファイルが付属しています。このキーストア ファイルには ServiceNow 公開キーが含まれています。この公開キーは、ServiceNow の署名が付いた暗号化ルールの検証に使用されます。

注: ベース システムの Java JCEKS KeyStore 以外のキーストアを使用する場合は、ServiceNow 公開キーをキーストアにインポートする必要があります。公開キーの別名は *servicenow* です。

暗号化キーだけでなく、Java JCEKS KeyStore は RSA キー ペアやデジタル証明書の格納にも使用します。RSA キー ペアは、インスタンスに格納される暗号化設定と暗号化ルールにデジタル署名する場合に使用します。デジタル証明書は、エッジ暗号化 プロキシがブラウザーや他のクライアントとのセキュア接続を確立するために使用します。

エッジ暗号化の **SafeNet** キーのバージョニング

SafeNet キーバージョニングを使用してキーの変更を簡素化します。SafeNet キーバージョニングでは、キーごとに新しいエイリアスを作成するのではなく、同じエイリアスを維持してバージョンを増やします。

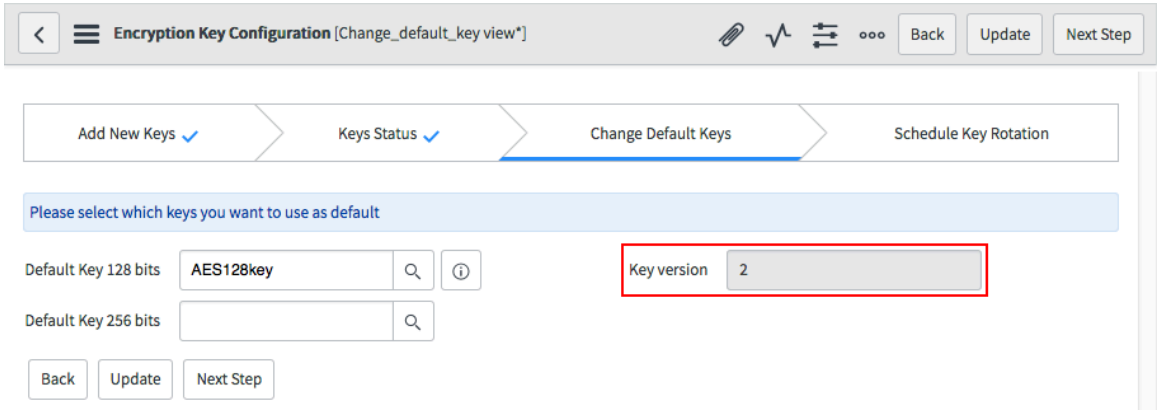
Edge プロキシサーバーで SafeNet キー バージョニングを設定する場合は、SafeNet でキー バージョニングをセットアップする必要があります。

注: London リリースより前にインストールされた Edge プロキシは、SafeNet キーをサポートしていますが、SafeNet キーバージョニングはサポートしていません。Kingston 以前のプロキシで誤ってバージョニングされたキーを使用した場合、London リリース以降へのアップグレード時に London 以降のプロキシによってこの問題が検出され、データ消失の可能性を防ぐためにプロキシが起動しません。

最初に、一括キー ローテーション ジョブまたは単一のキー ローテーション ジョブをスケジューリングして、古い SafeNet でバージョニングされたキーをバージョニングされていないキーに置き換え、必要に応じて新しい SafeNet でバージョニングされたキーを作成する必要があります。新しくバージョニングされたこのキーは London 以降のプロキシで安全に使用でき、プロキシを再起動することができます。

暗号化キーの設定

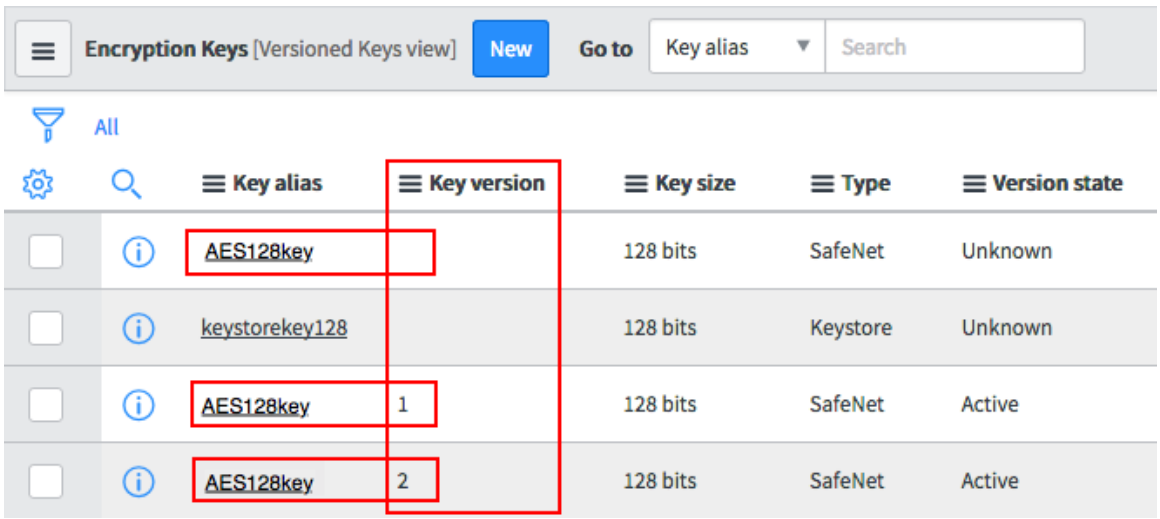
SafeNet でバージョニングされたキーを使用する場合、[暗号化キーの設定] フォームの [デフォルトキーの変更] セクションには、デフォルトの 128 ビットおよび 256 ビット キーの [キーバージョン] に対応した新しいフィールドが含まれます。[キーバージョン]フィールドはグレー表示され、編集することはできません。



手順については、「インスタンスで暗号化キーを設定する」を参照してください。

バージョニングされたキー

SafeNet でバージョニングされたキーを使用している場合は、 エッジ暗号化の設定 > 暗号化キーの設定 > すべてのキーでは、バージョニングされたキー にはキーバージョンが含まれます。



[暗号化キーの設定] フォームの [デフォルトキーの変更] セクションで入力した最初のエントリについては、バージョン番号は表示されません。プロキシサーバーが SafeNet からキーを要求すると、別名用の新しい行が追加され、[キーバージョン] が追加されます。

上記の例では、**[AES128key]** が 3 回表示されています。

- 1 つ目の表示は最初のエントリで、[キーバージョン] が示されていません。
- 2 つ目の表示は、SafeNet から返されたキーの最初のバージョンで、[キーバージョン] 列に **1** と表示されています。
- 3 つ目の表示は、SafeNet から返されたキーの 2 つ目のバージョンで、[キーバージョン] 列に **2** と表示されています。
- 他のバージョンのキーが SafeNet から返されるたびに、現在使用中の [キーバージョン] を記録するための新しい行が追加されます。

暗号化の設定とパターン

エッジ暗号化 では、フィールドを暗号化して文字列をトークン化することができます。

暗号化設定

暗号化設定を使用して個々のフィールドを暗号化することができます。エッジ暗号化 は、AES 128 ビットの暗号化キーをサポートしています。Java Cryptography Extension (JCE) の Unlimited Strength Jurisdiction Policy (無制限強度管轄ポリシー) ファイルがインストールされている場合、エッジ暗号化 は各暗号化タイプについて AES 256 ビットの暗号化キーをサポートします。エッジ暗号化 は次のタイプの暗号化設定をサポートしています。

標準暗号化

フィールド暗号化の値は、フィールド値が同じままになる場合でも、フィールドが暗号化されるたびに異なる値になります。標準暗号化は、最も堅牢な暗号化形式です。標準暗号化を使用しているフィールドを、ソート、グループ化、またはフィルターすることはできません。

等価性保存暗号化

フィールド暗号化の値は、フィールド値が同じままになる場合は同じです。フィールドの等価比較とグループ化操作をサポートします。

> **注:** すでにデータが格納されているフィールドに対して等価性保存暗号化を選択した場合は、フィールドでグループ化アクションを実行しても、暗号化フィールドと非暗号化フィールドが混在していると、同じ値を持つフィールドがグループ化されないことがあります。

順序保存暗号化

トークンと暗号化を使用して、プロキシ データベースのデータを保護します。等価比較、グループ化操作、およびデータのソート機能をサポートします。順序保存暗号化タイプは、MySQL データベースが エッジ暗号化 プロキシサーバー用に設定されている場合にのみサポートされます。

> **注:** 順序保存暗号化を使用している場合、プロキシ データベースがダウンしたときに、順序保存暗号化を使用してフィールドに更新を加えることができます。ただし、これらのフィールドに基づいてデータをソートしようとすると、ソート順が正しくなりません。また、グループも想定どおりに機能しません。プロキシ データベースが再び稼動したら、順序トークンの修復ジョブをスケジュールして、欠落しているトークンを修復してください。

暗号化タイプ

次の暗号化タイプは、セキュリティの質が低下する順序で記載されています。

暗号化タイプ	説明
標準 AES 256	フィールドをフィルター、ソート、または比較することはできません。
標準 AES 128	フィールドをフィルター、ソート、または比較することはできません。
等価性保存 AES 256	等価比較を使用してフィールドをフィルターすることができます。
等価性保存 AES 128	等価比較を使用してフィールドをフィルターすることができます。
順序保存 AES 256	フィールドをソートでき、等価比較によるフィルターを使用できます。ネットワークで MySQL データベースを使用する必要があります。

暗号化タイプ
次の暗号化タイプは、セキュリティの質が低下する順序で記載されています。

(続く)

暗号化タイプ	説明
順序保存 AES 128	フィールドをソートでき、等価比較によるフィルターを使用できます。ネットワークで MySQL データベースを使用する必要があります。

暗号化パターン

暗号化パターンを使用して文字列内の機密データを保護することができます。暗号化パターンが保存されて有効化されると、エッジ暗号化 プロキシサーバーによって、要求のパターンと一致する文字列が識別されます。見つかったクリアー テキスト文字列はプロキシ データベースに格納され、インスタンス上でトークンに置き換えられます。暗号化パターンを使用して、社会保障番号やクレジットカード番号などの規則的なパターンに一致する文字列をトークン化することができます。暗号化の主要な手段としては暗号化設定をお勧めしますが、暗号化フィールド以外の場所にある機密情報を検索して保護する場合は暗号化パターンを補足手段として使用してください。

注: エッジ暗号化 プロキシサーバーでネットワークにある MySQL データベースが必要になるのは、順序保存の暗号化または暗号化パターンを使用している場合のみです。クリアー テキスト値は、ネットワークのプロキシ データベースに格納されます。このため、プロキシ データベースを保護し、定期的にバックアップすることが重要です。推奨事項については、「エッジ暗号化のコンポーネント」を参照してください。

関連トピック

- 暗号化設定を使用したフィールド暗号化
- 暗号化パターンを使用して文字列をトークン化する

エッジ暗号化と一緒にインストールされるもの

エッジ暗号化 では、暗号化関連のデータを格納するためのテーブル、デフォルトの動作を設定するためのシステムプロパティ、および、エッジ暗号化 を管理するための edge_encryption ロールがインストールされます。

エッジ暗号化とともにインストールされるテーブル

エッジ暗号化 は、次のテーブルを追加します。

エッジ暗号化の設定 [sys_encryption_configuration]
: 添付ファイルが暗号化される際の対象となる暗号化フィールドおよびテーブルが含まれています。

エッジ暗号化のルール [sys_encryption_rule]
: 各ルールのレコードが含まれています。ルールは、名前、使用時の条件、スクリプト、および順序フィールドを持っています。

エッジ暗号化の無効な挿入ログ [sys_edge_encryption_invalid_insert_log]
: 暗号化されていないデータを暗号化フィールドに保存しようとしたときに作成されるログ メッセージが含まれています。

エッジ暗号化のプロキシ [sys_encryption_proxy]
: 暗号化プロキシ アプリケーションに関する情報が含まれています。

Edge Proxy の暗号化タイプ [sys_proxy_encryption_type]
: 暗号化フォームでの暗号化タイプの有効化と無効化に使用されます。

暗号化ジョブの実行 [sys_encryption_job_execution]

一括暗号化ジョブをサポートします。

暗号化ジョブ実行チャンク [sys_encryption_job_execution_chunk]

一括暗号化ジョブをサポートします。

スケジュールされた暗号化ジョブ [sysauto_encryption_job]

暗号化、復号化、キー ローテーション、順序トークンの修復、およびデータベースの回復のための、スケジュール済みジョブを列挙します。

暗号化キーの設定 [sys_encryption_key_configuration]

デフォルト暗号化キーを列挙します。

暗号化キー [sys_encryption_key]

利用可能なキーとキー属性を列挙します。

プロキシ暗号化キー [sys_encryption_proxy_key]

プロキシ暗号化キーを列挙します。

エッジ暗号化でインストールされるプロパティ

エッジ暗号化 は、次のプロパティを追加します。

注: システムのプロパティ [sys_properties] テーブルを開くには、ナビゲーションフィルターに「sys_properties.list」と入力します。

glide.edge.pattern.disallowed.chars

パターンで許可されていない文字のリスト。

- タイプ：値をカンマで区切ったリストの文字列
- 場所：システムプロパティ [sys_properties] テーブル

glide.edge.pattern.min.size

許容される最小パターン サイズ。パターンを小さくするほど一致が増えるため、オーバーヘッドが増加します。

- タイプ：数値
- デフォルト値：5
- 場所：システムプロパティ [sys_properties] テーブル

sn_edge_encryption.logging.destination

メッセージが記録される場所。

- タイプ：文字列
- デフォルト値：ファイル
- 場所：システムプロパティ [sys_properties] テーブル

sn_edge_encryption.logging.verbosity

使用するログ レベル。

- タイプ：文字列
- デフォルト値: info
- 場所：システムプロパティ [sys_properties] テーブル

sn_edge_encryption.encryption.proxy.buildtag

インスタンスに登録されるプロキシ バージョン。

- タイプ：文字列
- 場所：システムプロパティ [sys_properties] テーブル

sn_edge_encryption.cleartext.allowed

true の場合、クリアー テキストは暗号化フィールドに保存できます。これは、ユーザーが エッジ暗号化 プロキシを通過せずにインスタンスにアクセスしようとした場合に発生します。false の場合、クリアー テキストは暗号化フィールドに保存できなくなります。

- タイプ：ブーリアン
- デフォルト値：false
- 場所：システムプロパティ [sys_properties] テーブル

エッジ暗号化の計画立案

エッジ暗号化 の実装を成功させるためには計画と準備が必要です。

計画段階では次の質問に答えてください。

- 暗号化するフィールドはどれか。
- どの暗号化タイプを使用するか。
- エッジ暗号化 プロキシはいくつ必要か。推奨事項や考慮事項については、「エッジ暗号化環境のサイジング」を参照してください。
- 順序保存の暗号化タイプまたは暗号化パターンを使用する場合、MySQL データベースをどこに配置するか。
- どのキー管理システムを使用するか。

エッジ暗号化 を実装するために必要なタスクは、システムアドミニストレーター、ネットワークアドミニストレーター、およびセキュリティ チームのメンバーで異なります。

- システムアドミニストレーターには security-admin ロールが必要です。システムアドミニストレーターは次のことを行う必要があります。
  - エッジ暗号化 プロキシアプリケーションをダウンロードします。
  - プロキシがインスタンスへの接続に使用する エッジ暗号化 ユーザー アカウントをセットアップします。ユーザーに edge_encryption ロールをアサインする必要があります。
  - 暗号化キーの設定を行い、デフォルトキーを設定します。
  - インスタンス上で エッジ暗号化 の設定を行います。
  - 暗号化ジョブをスケジュールします 。
  - エッジ暗号化 を監視します。
  - 暗号化ルールを作成および編集します。
- ネットワークアドミニストレーターは次のことを行う必要があります。
  - エッジ暗号化 プロキシアプリケーションをインストールします。
  - 順序保存の暗号化および暗号化パターンに使用されるプロキシサーバーおよびプロキシ データベースのネットワーク アドレスを把握します。
  - 順序保存の暗号化および暗号化パターンに使用するプロキシ データベースをインストールします。

- プロキシ アプリケーションを開始および停止します。
- 暗号化キー管理を実行します。
- ユーザーを暗号化プロキシ アプリケーションにマッピングする方法を決定します。これは DNS 設定またはルーティング ルールを使用して行うことができ、ネットワークごとに固有になります。
- 複数のプロキシサーバーを管理します。
- ロードバランサーのプールなどの設定を行います。

- セキュリティアドミニストレーターは、各フィールドにアサインする暗号化タイプを決定する必要があります。

エッジ暗号化 のシステム要件

エッジ暗号化 プロキシアプリケーションは、Microsoft Windows または Linux オペレーティングシステムを実行しているサーバーまたは仮想マシンで実行できます。パフォーマンスを最適化するため、設定がこれらの要件を満たすことを確認します。

Java の要件

エッジ暗号化 プロキシサーバーをインストールまたは実行するホストマシンでは、サポートされているバージョンの Java を保持する必要があります。現在サポートされているバージョンは 11.x バージョンシリーズの Java 11.0.6 以降です。

注: Java 8 は Utah リリースからサポートされなくなりました。Utah バージョンの エッジ暗号化 プロキシをインストールする前に、エッジ暗号化 プロキシを使用する環境を Java 11 にアップグレードしてください。

注: Java は無制限の強度キーを自動的に許可しません。AES 256 ビット暗号化の使用を特別に有効にする必要があります。

OpenJDK のサポート

ServiceNow AI Platform は OpenJDK バージョン 11 をサポートしています。

プロキシサーバーの最小構成

プロキシサーバーには以下の最小構成が必要です。

- プロキシサーバーあたり 4 GB の RAM (ほとんどの展開で 6 GB を推奨)

  注: プロキシサーバーホストには、プロキシサーバーより少なくとも 1 GB 多い RAM が必要です。プロキシサーバーホストには、オペレーティングシステムサービス用に追加で 1 GB が必要です。たとえば、プロキシサーバーが RAM を 4 GB 使用するよう設定する場合は、プロキシサーバーホストに少なくとも 5 GB の RAM を取り付ける必要があります。

  プロキシサーバーには少なくとも 4 GBのメモリが必要なため、32 ビット JRE と 32 ビット オペレーティング システムは London リリース以後はサポートされなくなりました。

- 3 GHz 以上の CPU (最適なパフォーマンスには 4 コア CPU を推奨)
- ロードバランサーの背後で稼働する複数のプロキシサーバー。必要なプロキシサーバーの数は、アプリケーションノードの数、同時ユーザー数、およびフェイルオーバーに必要なサーバー数によって異なります。詳細については、「エッジ暗号化環境のサイジング」を参照してください。
- サーバーの使用率やリソースの可用性に応じて他のサービスと同時に実行できる機能

プロキシサーバーをサポートするシステム

次のシステムがサポートされています。

サポートされているシステム	説明
Windows Server 2012、2012-R2、2016、2019 の各エディション	• 仮想マシンまたは物理ハードウェア • 64 ビット システム
Linux	• 仮想マシンまたは物理ハードウェア • 64 ビット システム 64ビットの Linux システムでは、32 ビットの GNU C ライブラリ (glibc) をインストールする必要があります。CentOS のインストールコマンドは `yum install glibc.i686` です。

プロキシサーバーのバージョン要件

エッジ暗号化 プロキシバージョンを ServiceNow インスタンスバージョンと同期させてください (たとえば Tokyo などの同じメジャーリリース)。アップグレードプロセス中のダウンタイムをなくすために、 エッジ暗号化 プロキシには下位互換性があります。ただし、ユーザーが新しい機能や重要なバグ修正にアクセスできるようにするために、できるだけ早くアップグレードすることが重要です。

プロキシサーバーの接続要件

エッジ暗号化 アプリケーションを実行するプロキシサーバーは、ネットワーク内のマシンと通信できる必要があります。プロキシサーバーに次のネットワーク権限があることを確認してください。

ネットワーク権限	説明
ファイアウォールアクセス	プロキシサーバーとクライアントデバイスの間にあるファイアウォールの設定を行い、接続を許可します。ネットワークで非武装地帯 (DMZ) を使用してセキュリティの追加層をローカルエリアネットワーク (LAN) に追加していて、ネットワークセキュリティプロトコルによりネットワーク内部から DMZ へのポートアクセスを制限している場合は、プロキシサーバーを DMZ 内のマシンに展開する必要があります。
ネットワークアクセス	プロキシサーバーからの接続が可能になるように各クライアントを設定します。ネットワークセキュリティにより、クライアントに接続できる新しいマシンの設定ができない場合は、接続権限を持つ既存のマシンにプロキシサーバーをインストールします。
インスタンスアクセス	プロキシサーバーがインスタンスにネットワークアクセスできることを確認します。プロキシサーバーネットワークが、TCP ポート 443 経

ネットワーク権限	説明
	由のトラフィックを許可するように設定されていることを確認します。
ネットワークアカウント	ローカルアドミニストレーターかドメインアドミニストレーターのいずれかを使用して、プロキシサーバーをインストールします。

順序保存およびトークン化データベースのシステム要件

順序保存の暗号化および暗号化パターンでは、エッジ暗号化 プロキシサーバー用に Oracle MySQL データベースを設定する必要があります。順序保存暗号化は、先にデータを復号化しなくても、比較演算を暗号化データに直接適用できます。暗号化パターンを使用すると、データベースに送信して格納する前に、文字列パターンをトークンで置換 (トークン化と呼ばれる) できます。MySQL データベースのサイズが理由で、順序保存およびトークン化データベースを実行するには専用のプロキシサーバーを使用します。

データベースシステムの最低要件には次のものが含まれます。

MySQL データベース	要件
バージョン	MySQL データベースバージョン 5.7 と 8.0 **注:** MySQL バージョン 5.5 および 5.6 はテストされなくなり、サポートが終了しました。
OS	64 ビット システム
CPU	2 GHz 以上の CPU (最適なパフォーマンスには 4 コア CPU を推奨)
RAM	16 GB
ディスク	ストレージエリアネットワーク (SAN) またはローカルストレージ (RAID 10 を推奨)
サイズ	潜在的なレコードの数にレコードサイズを掛けて決定します。「順序保存とトークン化のデータベースのサイズを計算する」を参照してください。
設定	高可用性クラスター。MySQL サーバーの設定方法が分からない場合は、設定情報について MySQL を参照してください。

エッジ暗号化環境のサイジング

環境に合わせてプロキシサーバーの台数を選択することは重要なタスクです。ユーザー数、冗長性の必要性、許容可能なレイテンシーを考慮してください。

冗長性

ハードウェア障害に備えて冗長のプロキシサーバーを維持してください。プロキシサーバーが到達不能になった場合でもすべてのユーザーに機能パスを提供できるように、プロキシサーバーをロードバランサーの背後に配置してください。少なくとも 2 台のプロキシサーバーが常に使用可能となるようにしてください。

サイズ

ここでのサイズとは、データの暗号化によって生じるレイテンシーの増大を避けるために必要なプロキシサーバーの台数のことです。使用状況によっては、プロキシサーバーを追加してレイテンシーを減らすことができます。たとえば、定期的な一括暗号化を実行する場合は、負荷を処理するためにプロキシサーバーを追加するか、ユーザー負荷が軽い状態のときに一括暗号化を実行します。また、プロキシサーバーを実行するハードウェアはパフォーマンスやレイテンシーに影響を与えます。CPU が高速で数が多く、RAM の容量が多いハードウェアでプロキシサーバーを実行する方が、低速で性能が限られたシステムと比べて、よい高いスループットが得られます。

下記のガイドラインでは、プロキシサーバーが最低限のハードウェア要件で実行されていることを前提としています。プロキシサーバーの台数を決定するには次のことを行ってください。

- インスタンス上の 2 つのアプリケーション ノードごとに 1 台ずつプロキシサーバーをセットアップすることを検討します。
- 冗長性を確保するため、ロードバランサーの背後に少なくとも 2 台のプロキシサーバーをセットアップします。
- 500 人の同時ユーザーごとに追加のプロキシサーバーを用意します。
- 必要な冗長性に応じて、フェイルオーバー用のプロキシサーバーを追加します。

たとえば、2,000 人のユーザーが存在するインスタンスの場合は、ロードバランサーの背後に少なくとも 5 台のプロキシサーバーを用意します。この計算には、500 人のユーザーごとに 1 台ずつのプロキシサーバーと、フェイルオーバー用の追加のプロキシサーバーが含まれています。ユーザー数がいつしきい値の 500 人に近づくかを事前に判断して、ロードバランサー プールにもう 1 台のプロキシサーバーを配置してください。

ロードバランサー

要求のバランスを保ち、サーバーの応答時間を向上させるために、プロキシサーバーをロードバランサー プールに分散させてください。「最小数の接続」による方法を使用するようにロードバランサーを設定します。この方法では、有効な接続が最も少ないプロキシサーバーに要求が接続され、1 台のプロキシの過負荷が防止されます。

CPU 使用率

データの暗号化とトークン化は CPU に高い負荷がかかる処理です。そのため、データの暗号化中に CPU のスパイクが生じるのは正常であり、予期されることです。CPU 使用率が数分間で一度に 80% を超えると、プロキシサーバーが過負荷状態になる可能性があります。こうした状態が生じると、CPU 使用率が高い間、レイテンシーが増加します。レイテンシーが持続する場合は、もう 1 台のプロキシサーバーを追加することでレイテンシーの減少が促される可能性があります。

メモリ

プロキシサーバーには最低 4 GB の RAM が必要です (6 GB を推奨)。プロキシサーバーの初期メモリ制限と上限メモリ制限を推奨値に設定してください。

順序保存とトークン化のデータベースのサイズを計算する

順序保存の暗号化または暗号化パターンを使用する場合は、潜在的なレコードの数にレコード サイズを掛けて MySQL データベースのサイズを決定します。

始める前に
必要なロール：admin

このタスクについて
順序保存およびトークン化データベースの実行には専用のマシンを使用してください。プロキシサーバーと同じハードウェア上でデータベースを実行しないでください。

手順

1. 順序保存の暗号化による暗号化フィールドを含む可能性があるレコードについて、その潜在的な数を決定します。
   a. 順序保存の暗号化を使用する暗号化設定の数に、それぞれの設定が適用されるレコードの数を掛けます。
   b. 拡張分を考慮する場合は、結果に 3 を掛けます。
2. 手順 1 の結果に 1,536 を掛けます。
   1,536 はレコードの平均サイズ (バイト単位) です。
3. 暗号化パターンを使用する場合は、トークン化されるレコードについて手順 1 ～ 2 を実行し、結果を合計に加算します。

結果
以上で計算した値が、順序保存およびトークン化データベースの推奨サイズ (バイト単位) になります。

エッジ暗号化 の制限事項

エッジ暗号化 はシステムの機能に影響を与えます。フィールド暗号化が及ぼす影響について十分に評価してください。

フィールド タイプの制限

次のフィールドタイプのみを暗号化できます。

- 日付
- メール
- 日付/時刻
- HTML
- IP アドレス
- ジャーナル
- ジャーナル入力
- 複数行テキスト
- 1 行テキスト
- 文字列
- URL

次のフィールドタイプは暗号化できません。

- 選択肢フィールド
- 仮想フィールド
- システムテーブルのフィールド (sys_user の特定のフィールドを除く)
- テーブル内のシステムフィールド

- 自動番号付けスキームに関連付けられている数値フィールド。
- 上記以外のその他のフィールドタイプ

追加の制限：

- ジャーナルフィールドが暗号化されている場合は [投稿] ボタンが無効になります。たとえジャーナルフィールドが複数あり、それらのフィールドのうち 1 つだけが暗号化されている場合でも同様です。
- [検索項目] およびヘッダーフィルターボックスでは、暗号化フィールドは使用できません。
- インデックスとして使用されるフィールドを暗号化するときは、順序保存および等価性保存の暗号化タイプのみを使用できます。標準の暗号化タイプを使用してインデックス付きのフィールドを暗号化することはできません。

詳細については、「フィールドタイプ」を参照してください。

フィルタリングと検索の制限

標準暗号化

標準暗号化フィールド設定の文字列、日付、日付/時刻、または URL フィールドをフィルターの左オペランドとして選択した場合、フィルターオプションを利用できません。

等価性保存暗号化

等価性保存暗号化フィールド設定の文字列、日付、日付/時刻、または URL フィールドをフィルターの左オペランドとして選択した場合、次の演算子を利用できます。

- 次の値に等しい
- ではない
- は、空
- は空でない

順序保存暗号化

順序保存暗号化フィールド設定の文字列フィールドをフィルターの左オペランドとして選択した場合、[次の値に等しい]、[次の値ではない]、[は空]、 [は空でない] の他に次の演算子を利用できます。

- **>**
- 次の値未満 **(<)**

順序保存暗号化フィールド設定の日付フィールドまたは日付/時刻フィールドをフィルターの左オペランドとして選択した場合、[次の値に等しい]、[次の値ではない]、[は空]、 [は空でない] の他に次の演算子を利用できます。

- トランザクションの後
- トランザクションの前
- トランザクションの日以降
- トランザクションの日以前

日付および日付/時刻ピッカー

日付フィールドの場合、日付ピッカーを使用して日付を指定します。

日付/時刻フィールドの場合、日時ピッカーを使用して日時を指定しま

す。

条件フィルターのリスト

リストでは [この値で絞り込み] および [この値を除外] オプションがサポートされています。完全一致のみが返されるか、フィルタリングで除外されます。

> **注:** 条件フィルターでの暗号化フィールドの追加は、UI ポリシーやビジネスルールなどのスクリプトでサポートされます。

設定の制限

暗号化設定の制限と動作は以下のとおりです。

- フィールドを エッジ暗号化 の設定テーブルに追加した後で、設定レコードを削除することはできません。フィールド暗号化が不要になった場合は、エッジ暗号化 設定テーブルでレコードを非アクティブ化し、暗号化ジョブをスケジュールしてデータを復号化します。
- 親テーブルのフィールドが暗号化対象としてマークされている場合、すべての継承テーブルのフィールドも暗号化されます。たとえば、タスクテーブルの [簡単な説明] フィールドが暗号化される場合、インシデントテーブルの [簡単な説明] フィールドの内容も暗号化されます。
- 親テーブルから継承されたフィールドが暗号化対象としてマークされている場合、親テーブルのフィールドは暗号化できません。たとえば、インシデントテーブルの [簡単な説明] が暗号化対象としてマークされている場合、タスクテーブルの [簡単な説明] は暗号化できません。この例では、問題テーブルの [簡単な説明] は暗号化できます。
- 暗号化設定が定義されているフィールドを任意の形式にエクスポートすると、プロキシサーバー経由でエクスポートした場合でも、その出力に暗号化された値が含まれます。
- 暗号化設定が定義されているフィールドにデータをインポートすることはできません。

- 継承された日付および日付/時刻フィールドは暗号化できません。親テーブルから継承された日付または日付/時刻フィールドは、[列] フィールドのドロップダウンリストにリストされず、これらのフィールド用に日付または日付/時刻の暗号化設定を作成できません。
- 文字列または URL フィールドは、親テーブルまたは子テーブルからのみ暗号化することができますが、両方からは暗号化できません。

インスタンスの制限

インスタンスで エッジ暗号化 の使用が及ぼす影響は以下のとおりです。

- バックエンドロジックは暗号化されたデータを処理できません。インスタンスに暗号化されたデータが含まれている場合、暗号化フィールドのデータの評価に依存しているビジネスルール、バックエンドスクリプト、またはバックエンド機能は正しく実行されません。

  注: 等価性保存または順序保存の暗号化により暗号化されたデータもまた、同一の暗号化された値と比較されるときは、等価性のチェックを受けます。

- メール処理はメールシステムからインスタンスに直接送られ、Edge プロキシを経由させることはできないため、メールを介して送受信されるデータは Edge プロキシで暗号化または復号化することはできません。
  - 受信メールのデータと添付ファイルは暗号化されません。
  - 送信メールのデータと添付ファイルは暗号化されたままになり、復号化することはできません。
- サーバーで実行されるスクリプトは、暗号化されたデータを変更できません。
- グローバル検索はサポートされていません。グローバル検索では、暗号化されたデータとクリアテキストデータの両方について検索が試みられるため、結果が期待どおりにならないことがあります。
- 暗号化されたデータを、フィールドが暗号化されていないレコードにコピーして貼り付けることはできません。
- 選択した暗号化のタイプによっては、暗号化フィールドのユーザーインターフェイス機能が低下します。たとえば、比較、グループ化、ソート、検索の機能が影響を受ける可能性があります。一般に、選択する暗号化の強度が増すほど、機能性は低下します。
- Java KeyStore、SafeNet、および Unbound Technology 以外の、サードパーティによるソフトウェアまたはハードウェアの暗号化キー管理はサポートされていません。
- 複数のプロキシサーバーから単一のインスタンスへの接続はサポートされていますが、暗号化プロキシクラスターの管理およびモニタリングは利用できません。プロキシはそれぞれ個別に管理する必要があります。
- ワークロードや暗号化フィールド数などのシステム設定は、暗号化フィールドのパフォーマンスに影響する可能性があります。
- エッジ暗号化 プロキシサーバーは 1 つのインスタンスにしか接続できません。
- インスタンスで Oracle データベースを使用していて、暗号化対象としてマークしている文字列フィールドが 2925 文字より多い場合、順序保存の暗号化を選択していてもフィールドをソートできません。
- インスタンスで Oracle データベースを使用している場合、文字セットとしてサポートされるのは Unicode AL32UTF8 のみです。
- 暗号化されたデータはレポートでは使用できません。
- エッジ暗号化 はデータアーカイブでは使用できません。
- エッジ暗号化プロキシは、バッチ REST 要求 API を使用する要求を暗号化できません。エッジ暗号化プロキシを使用している場合は、 glide.uxf.disable_rest_batching システムプロパティを true に設定して REST バッチ処理を無効にします。

エッジ暗号化のインストール

エッジ暗号化 プロキシは、手動でインストールするか、エッジ暗号化 インタラクティブインストーラーを使用してインストールできます。

Java の要件

エッジ暗号化 プロキシサーバーをインストールまたは実行するホストマシンでは、サポートされているバージョンの Java を保持する必要があります。現在サポートされているバージョンは、17.x バージョンシリーズの Java 17.0.3 以降です。

注: Java 11 は Yokohama リリースをもってサポートされなくなりました。Yokohama 以降のバージョンの エッジ暗号化 プロキシをインストールする前に、エッジ暗号化 プロキシを使用して環境を Java 17 にアップグレードします。

プロキシサーバーのインストール

エッジ暗号化 のインストールには次の手順が含まれます。

- インタラクティブインストーラーまたは手動インストーラーを使用して、ネットワークのサーバーに エッジ暗号化 プロキシアプリケーションをインストールします。
- 暗号化設定と暗号化ルールにデジタル署名するための RSA キー ペアを生成します。
- AES 256 暗号化の使用を予定している場合は、Java Cryptography Extension (JCE) をインストールします。
- セキュア SSL 接続を使用している場合は、サーバー証明書を取得して Java KeyStore にインポートします。
- キーストアと暗号化キーをセットアップします。
- 順序保存の暗号化タイプまたは暗号化パターンを使用する場合は、ネットワークのマシンに MySQL データベースをセットアップします。
- 目的のプロパティを設定します。プロパティは edgeencryption.properties 設定ファイルにあります。
- エッジ暗号化 がプロキシサーバーからの要求を処理できるよう、プロキシサーバーが信頼できるソースであることを指定します。

プロキシサーバーへのアクセス

インストールが完了したら、`<ホスト>:<ポスト>` の URL 形式を使用して各ユーザーのブラウザーから エッジ暗号化 プロキシを参照します。値は、edgeencryption.properties ファイルのホストプロパティとポートプロパティによって決まります。

次の値はそれらの例です。

プロパティ	サンプル値
edgeencryption.proxy.host	hostname.mycompany.com
edgeencryption.proxy.http.port	8081

クライアントはアドレス `http://hostname.mycompany.com:8081/` を使用してプロキシサーバーにアクセスします。

注: DNS 設定とルーティングルールを使用することもできます。ホストとポートの値はネットワークアドミニストレーターが決定します。

エッジ暗号化を要求する

エッジ暗号化 プラグイン (com.glide.edgeencryption) は個別のサブスクリプションとして使用できます。

始める前に

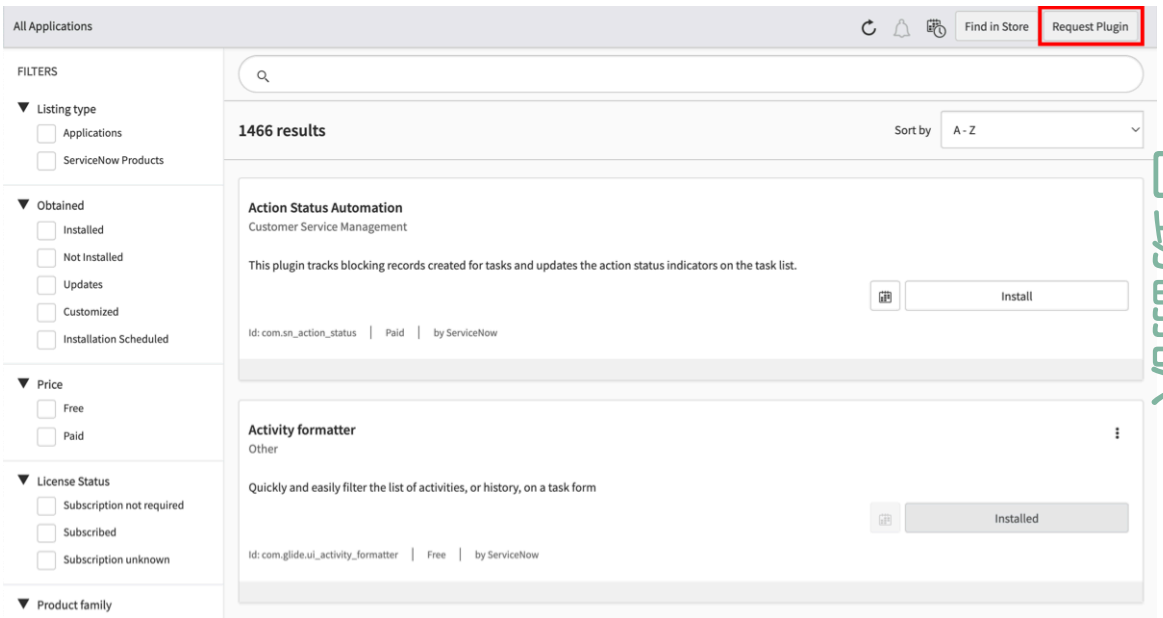
サブスクリプションを購入するには、ServiceNow アカウントマネージャーにお問い合わせください。アカウントマネージャーは通常数日以内に、組織の本番および準本番インスタンスでプラグインが有効化されるように手配することができます。

アカウントマネージャーがいない場合、購入後に有効化を延期することを決定するか、無料で準本番インスタンスで製品を評価する場合には次の手順を実行します。

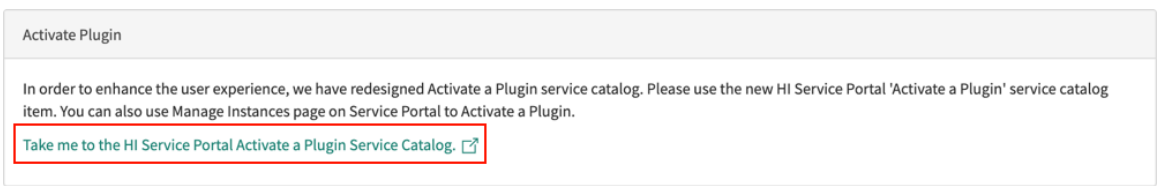
必要なロール：なし

手順

1. 移動先 すべて > システムアプリケーション > 利用可能なすべてのアプリケーション > すべて.
2. [すべてのアプリケーション] ページで [プラグインの要求] をクリックして、Now Support で [プラグインをアクティブ化] フォームを開きます。



3. Now Support で、Now Support サービスポータル サービスカタログ にアクセスするリンクを選択します。



4. インスタンスを選択します。
5. [アクション] > [プラグインのアクティブ化] を選択します。
6. [プラグインのアクティブ化] フォームで、次の情報を入力します。

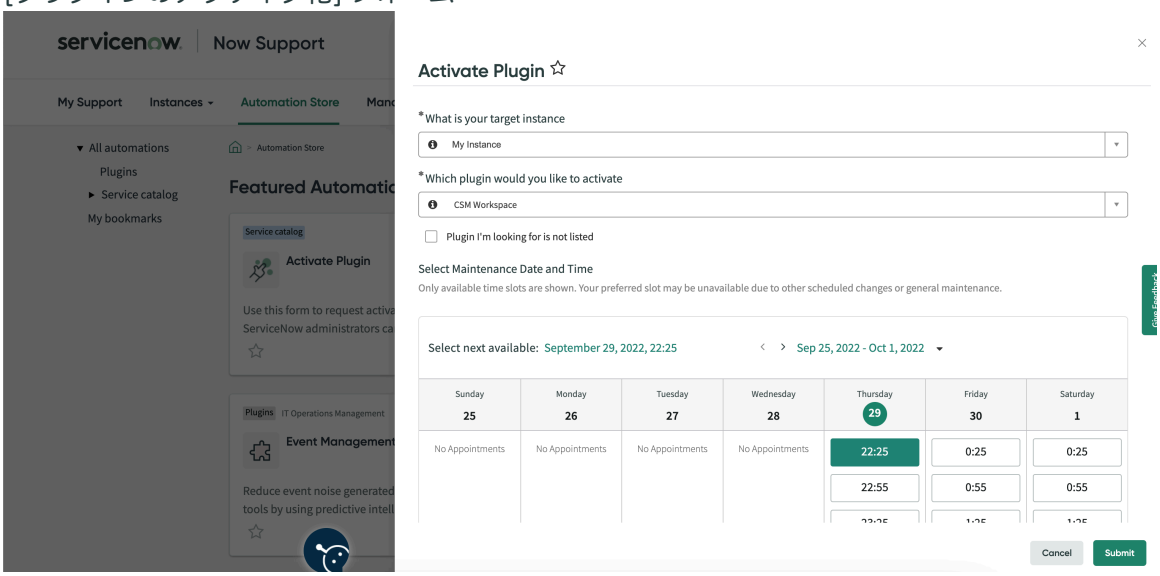
[プラグインのアクティブ化] フォーム

フィールド	説明
ターゲットインスタンスは何ですか	プラグインをアクティブ化するインスタンス。
どのプラグインをアクティブ化しますか	アクティブ化するプラグインの名前です。 **注:** 必要なプラグインが表示されない場合、または OEM またはオンプレミスのインスタンスでプラグインをアクティブ化している場合は、[探しているプラグインが表示されていません **(Plugin I'm looking for is not listed)]** チェックボックスをオンにして、プラグインの名前を入力します。
メンテナンスの日時を選択 (Select Maintenance Date and Time)	プラグインをアクティブ化する日時。 **注:** プラグインは、米国太平洋標準時で、毎営業日の朝と夕方の 2 回のバッチでアクティブ化されます。特定の時刻にプラグインをアクティブ化する必要がある場合は、[理由/コメント **(Reason/Comments)]** フィールドに要求を入力します。

Example

たとえば、 [自分のインスタンス (My Instance)] という名前のインスタンスで CSM Workspace プラグインをアクティブ化するには、次のフォームを参照してください。

[プラグインのアクティブ化] フォーム



7. **[Submit** (送信)] を選択します。
   プラグインの要求の詳細については、次を参照してください。 のサービスカタログ [KB0751715] 記事からのプラグインの要求 Now Support ナレッジベース.

エッジ暗号化ユーザー アカウントを設定する

エッジ暗号化 プロキシは、暗号化設定情報を取得および更新するために、ユーザーとしてインスタンスに接続します。この目的のためにユーザー アカウントを作成し、edge_encryption ロールをユーザーに与えます。

始める前に

ロールをアサインする前に エッジ暗号化 プラグインをインストールする必要があります。

必要なロール：admin

手順

1. ServiceNow インスタンスで、エッジ暗号化 プロキシ アプリケーションで使用されるユーザー アカウントを作成します。
2. edge_encryption ロールをユーザーにアサインします。

エッジ暗号化プロキシサーバーをダウンロードする

エッジ暗号化 プロキシサーバー アプリケーションをインスタンスからダウンロードし、エッジ暗号化 プロキシサーバーを実行する予定の各コンピューターにファイルをコピーします。

始める前に

この手順を始める前に、エッジ暗号化 プラグインをインスタンスにインストールして有効化する必要があります。

注: エッジ暗号化 プロキシは、Oracle JRE でのみ正式にサポートされています。

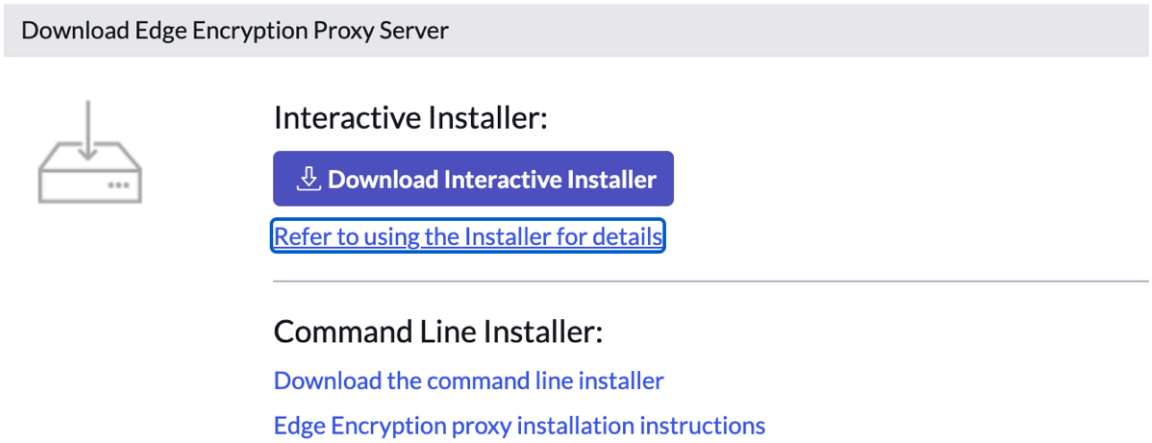
必要なロール：security_admin

このタスクについて

手順

1. 移動先 すべて > エッジ暗号化の設定 > インストールとダウンロード > ダウンロード.
2. 対話型インストーラーを使用するには、[インタラクティブインストーラのダウンロード] をクリックします。

   プロキシサーバーを手動でインストールする場合は、プロキシサーバーの OS バージョンを選択します。



> **注:** プロキシサーバーが稼働するには少なくとも 4 GBのメモリが必要なため、32 ビット JRE と 32 ビットオペレーティングシステムは Washington DC リリース以後はサポートされなくなりました。

3. エッジ暗号化 プロキシサーバーを実行する予定の各コンピューターにインストーラーをコピーします。

> **注:** エッジ暗号化 プロキシサーバーを手動でインストールする場合は、エッジ暗号化 プロキシサーバーを実行する予定の各コンピューターに ZIP ファイルをコピーします。

次のタスク

エッジ暗号化 インストーラーをダウンロードしたら、「インタラクティブ インストーラーを使用してエッジ暗号化プロキシサーバーをインストールする」の手順を実行します。手動でインストールする場合は、「コマンド ライン インストーラーを使用してエッジ暗号化プロキシサーバーをインストールする」のの手順を実行します。

インタラクティブ インストーラーを使用してエッジ暗号化プロキシサーバーをインストールする

インタラクティブ インストーラーを使用して、Windows または Linux コンピューターに エッジ暗号化 プロキシサーバーをインストールします。

始める前に

> **注:** SafeNet KeySecure キーストア ファイルは、エッジ暗号化 インストーラーではサポートされていません。SafeNet KeySecure キーストアを使用するには、「コマンド ライン インストーラーを使用してエッジ暗号化プロキシサーバーをインストールする」の手順を実行します。

この手順を始める前に、エッジ暗号化 プラグインをインスタンスにインストールして有効化する必要があります。エッジ暗号化 インストーラーを実行するマシンに Java バージョン 11.0.6 以降がインストールされていることを確認してください。

必要なロール：

- ServiceNow インスタンスでの security_admin
- Windows ホストでのローカルアドミニストレーターまたはドメインアドミニストレーター
- Linux ホスト上のファイルシステムへのフルアクセス権限があるサービスユーザー

このタスクについて
新しいプロキシサーバーをインストールした後、再度インストーラーを実行してテストを行い、インストールに関する問題を検出したり、現在の設定を変更したりすることができます。次の選択肢があります。

- [新規インストール]：プロキシサーバーを新規にインストールします。
- [インストールの確認]：以前のインストール環境で問題を検出して修正するためのテストを実行します。
- [既存への再インストール]：以前のインストール環境で問題を検出して修正するためのテストを実行し、既存の設定を表示または変更します。

注: Linux マシンの特権ポート (ポート 80 または 443) にプロキシサーバーをインストールする場合は、ファイル システムへのフル アクセス権を持つ root ユーザーとしてインストーラーを実行する必要があります。プロキシサーバーのインストール後にファイル システムのアクセス権を制限する場合、プロキシ インストーラーの SetUID 機能を使用できます。この機能を有効にするには、インストーラーを root または sudo root として起動します。インストーラーのプロンプトが表示されたら、特権のないユーザーのユーザー名とユーザー グループを入力します。指定したユーザーのファイル システム権限を使用して、プロキシサーバーがインストールされます。この手順をスキップして、root 権限でデフォルトのインストールを続けることができます。

手順

インストーラーを使用して、インスタンス用の複数のプロキシを複数のマシンにインストールし、次の条件が適用されることを確認します。

- すべてのプロキシに、同じ暗号キーと同じ RSA キー ペアを持たせる必要があります。これらは、暗号化設定と暗号化ルールへのデジタル署名に使用されます。
- 暗号化キーは、インスタンスで設定されたデフォルトのキーである必要があります。
- プロキシ データベースをインストールの一部としてセットアップする場合、すべてのプロキシで同じプロキシ データベースを使用する必要があります。

  等価性保存の暗号化、順序保存の暗号化、またはトークン化を行う場合、状況に応じてプロキシ データベースが必要になります。これらの機能をどれも使用しない場合は、プロキシ データベースは必要ありません。

次のタスク
NVDA を使用する場合は、「Java アプリケーションで 32 ビット NVDA を使用するように Windows 64 ビットホストを設定する」を参照してください。NVDA は、キーボード ユーザーを対象にアクセシビリティ対応 Java アプリケーションの読み上げのために開発された支援技術 (スクリーンリーダー) です。

エッジ暗号化 プロキシサーバーをインストールしたら、プロキシサーバーの初期メモリ制限と上限メモリ制限を設定します。

エッジ暗号化 プロキシサーバーをインストールする (インタラクティブ インストーラー)

Windows または Linux コンピューターに エッジ暗号化 プロキシをインストールします。

始める前に
必要なロール：admin

手順

1. エッジ暗号化プロキシサーバー インストーラーをダウンロードします。
2. エッジ暗号化 プロキシ インストーラーを開きます。

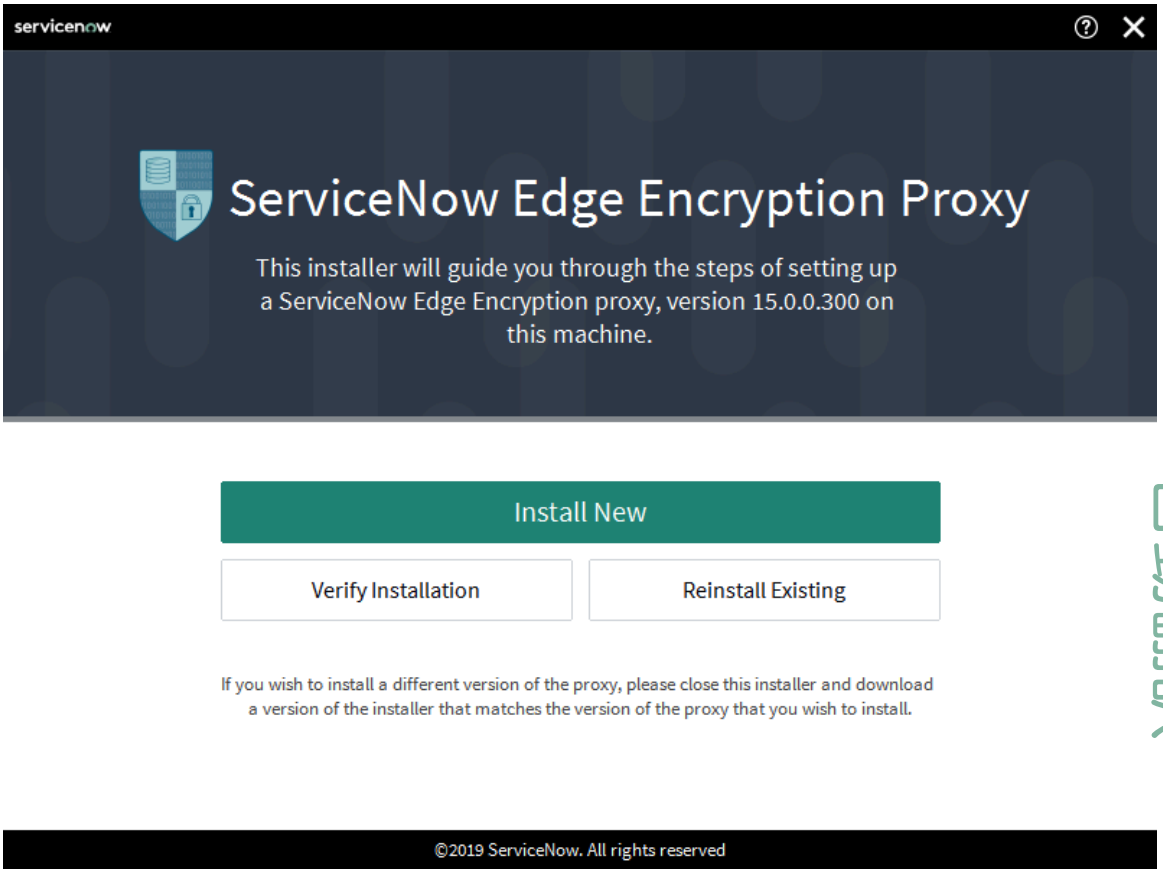
注: Windows マシンにインストールする場合は、インストーラーをアドミニストレーターとして実行する必要があります。

a. Windows マシンでインストーラーをアドミニストレーターとして実行するには、コマンド プロンプトを右クリックし、[アドミニストレーターとして実行]を選択します。

b. コマンド ラインから、ダウンロードした .jar ファイルがあるディレクトリーに移動します。

c. コマンド `java -jar <ファイル名>.jar` を実行します。

ServiceNow エッジ暗号化プロキシインストーラー



3. プロキシサーバーを新規にインストールするには、[新規インストール] を選択します。
すでにプロキシがインストールされている場合は、インストーラーで次の操作を実行できます。
   - [インストールの確認]：以前のインストール環境で問題を検出して修正するためのテストを実行します。
   - [既存への再インストール]：以前のインストール環境で問題を検出して修正するためのテストを実行し、既存の設定を表示または変更します。

4. インストールの場所とターゲットの **ServiceNow** インスタンスの設定を行います。

   a. [参照] をクリックしてインストールの場所を選択します。または、手動でインストール パスを入力します。

   b. ターゲットの ServiceNow インスタンスの URL を入力します。
   プロトコルとポート番号を含めます。

Example
https://example.servicenow.com:443

c. ターゲットの ServiceNow インスタンスで edge_encryption ロールを持つユーザーのユーザー名とパスワードを入力します。

5. [次へ **(Next)**] をクリックします。
6. 接続設定とプロキシ設定を行います。

設定	説明
プロキシ ホスト	プロキシサーバーをインストールするマシンの完全修飾ドメイン名。 注: **[FQDN** を検出] をクリックすると、マシンの完全修飾ドメイン名が検索され、[プロキシ ホスト] フィールドに自動入力されます。 このプロパティとポートによって、クライアントがプロキシサーバーへのアクセスに使用する URL が定義されます。
HTTP ポート	HTTP 通信のプロキシのポート。
HTTPS ポート	HTTPS 通信のプロキシのポート。
プロキシ名	プロキシとそのサービスの名前。プロキシ名は一意である必要があります
プロキシ ポーリング間隔	ポーリング間隔 (秒単位)。デフォルト設定では、プロキシが暗号化設定の変更を認識するまでに 5 秒かかります。値を大きくするほど、オンラインになったプロキシをインスタンスが検出するのに時間がかかります。 注: プロキシ ポーリング間隔のデフォルト設定を変更すると、プロキシがオンラインになったときに検出が遅れることがあります。
プロキシ キープアライブ ping 間隔	プロキシからインスタンスに発行される ping の間隔 (秒単位)。プロキシとインスタンス間の接続を確認するために、定期的に ping が発行されます。デフォルト値は 10 です。最小値は 5 です。

7. [インストール]をクリックします。
エッジ暗号化 プロキシサーバーがインストールされます。インストールには数分かかることがあります。

CyberArk のプロパティの保護を設定する

オプションで、CyberArk のプロパティの保護を設定して、エッジ暗号化 のパスワードを一元管理されたセキュアデジタルボールトに安全に格納します。

始める前に
必要なロール：admin

このタスクについて
CyberArk 接続パラメーターとプロキシサーバーの保護された認証情報を設定する前に、CyberArk AIM (アプリケーション ID 管理) を購入して設定する必要があります。AIM クライアントのインストールの一部として、JavaPasswordSDK.jar ファイルが AIM クライアントインストールディレクトリにインストールされます。CyberArk ボールトは独立した強化されたサーバーにインストールされ、AIM クライアントはそのサーバーに安全にアクセスできます。

注: CyberArk AIM クライアントは、Edge プロキシがインストールされているすべてのホスト コンピューターにインストールする必要があります。

Edge インストーラでは、JavaPasswordSDK.jar ファイルの場所を指定して、Edge プロキシへの CyberArk 接続を設定する必要があります。また、AIM クライアントのインストール時に定義した他の値も入力する必要があります。

CyberArk のパスワード保存の設定はオプションです。CyberArk のパスワード保存の設定をしない場合は、CyberArk の画面で [スキップ] をクリックします。

手順

1. エッジ暗号化 インストーラの CyberArk 接続ページで、CyberArk 接続パラメーターを入力します。

CyberArk 接続パラメーター

設定	説明
PasswordSDK.jar へのパス	CyberArk の設定中にホスト Windows マシンにインストールされた JavaPasswordSDK.jar ファイルへのパス。
アプリ ID	CyberArk の設定中に入力された [アプリ **ID]**。
安全な名前	CyberArk の設定中に入力された [安全な名前]。

2. [次へ] をクリックします。
3. インストーラの [CyberArk で保護された認証情報] ページで、CyberArk によって保護される認証情報を入力します。
   - すべての保護されたパスワードに単一の認証情報名を使用するには、[すべての認証情報に単一の認証情報名を適用する] チェックボックスをオンにし、認証情報名を入力して、[適用] をクリックします。
   - 次のフィールドの 1 つ以上に認証情報名を入力します。認証情報の名前は、CyberArk の構成中に SSH キーに入力されるユーザー名です。

CyberArk で保護された認証情報

設定	説明
エッジ暗号化ユーザー	エッジ暗号化ユーザーの CyberArk 認証情報名。
署名鍵キーストア	署名鍵キーストアの CyberArk 認証情報名。

設定	説明
HTTPS 証明書キーストア	HTTPS 認定キーストアの CyberArk 認証情報名。
暗号化キーストア	暗号化キーストアの CyberArk 認証情報名。
データベース	データベースキーストアの CyberArk 認証情報名。
SafeNet HTTPS 証明書キーストア	SafeNet HTTPS 認定キーストアの CyberArk 認証情報名。
SafeNet サーバー	SafeNet サーバーの CyberArk 認証情報名。
フォワードプロキシ	フォワードプロキシの CyberArk 認証情報名。

4. [次へ] をクリックします。

署名キーを設定する

エッジ暗号化 プロキシ インストーラーでプロキシサーバーをインストールした後、署名キーを設定します。

始める前に
必要なロール：admin

このタスクについて
プロキシサーバーによって設定やプロパティに変更が加えられると、署名キーによる署名が行われます。署名キーは、JCEKS KeyStore の非対称 RSA キー ペアである必要があります。

注: 複数のプロキシをインストールする場合は、各プロキシで同じ署名キーを使用する必要があります。

手順

1. エッジ暗号化 インストーラーの [署名キー] ページで、署名キーを格納するためのホスト マシン上のキーストアを選択します。
   - [新しい **Java KeyStore** を作成する]：新しいキーストアのディレクトリーの場所、名前、およびパスワードを入力します。
   - [既存のキーストアを使用する]：キーストア ファイルの場所とパスワードを入力します。
2. [次へ **(Next)**] をクリックします。
3. 署名キーを選択するか作成します。
   - [新しいキー]：このプロキシの署名キーを作成します。
   - [既存のキーを使用する]：選択したキーストアの RSA キーペアを使用します。
   - [既存のキーをインポートする]：別のキーストアから RSA キーペアをインポートします。キーストア ファイルを参照し、キーストアのパスワードを入力し、キー エイリアスを選択します。キーの新しいエイリアスを指定します。
4. [次へ **(Next)**] をクリックします。

HTTPS 証明書を設定する

クライアントからセキュア SSL 接続を使用して エッジ暗号化 プロキシサーバーに接続できるようにするため、HTTPS 証明書をプロキシサーバーにインポートします。

始める前に
必要なロール：admin

このタスクについて
エッジ暗号化 プロキシは、接続を試みるクライアントに対して、HTTPS 証明書を提供します。

手順

1. エッジ暗号化 インストーラーの [HTTPS 証明書] ページで、証明書を格納するためのキーストアを選択します。
   - [新しい **Java KeyStore** を作成する]：新しいキーストアのディレクトリーの場所、名前、およびパスワードを入力します。
   - [既存のキーストアを使用する]：キーストア ファイルの場所とパスワードを入力します。
2. [次へ **(Next)**] をクリックします。
3. 証明書を選択するかインポートします。
   キー エイリアスは、証明書に対して与えられるエイリアスです。
   - [既存の証明書を使用する]：選択したキーストアにある既存の証明書を使用します。
   - [ファイルまたはキーストアからインポートする]：別のキーストアまたは .cer ファイルから証明書をインポートします。キーストアまたは .cer ファイルを参照し、パスワードを入力し、エイリアスを選択します。証明書に対して新しいエイリアスを指定する必要があります。
4. [次へ **(Next)**] をクリックします。

AES 128 ビット暗号化キーを設定する

エッジ暗号化 プロキシインストーラーで HTTPS 証明書を設定したら、データを暗号化するように AES 128 ビット暗号化キーを設定します。

始める前に
必要なロール：admin

このタスクについて
暗号化キーは、/keys ディレクトリーにあるプレーン テキスト ファイルか、キーストア内部にある秘密キーです。AES 128 ビットおよび AES 256 ビットの暗号化キーに対してキーストアを使用する場合は、両方で同じキーストアを使用する必要があります。

Edge プロキシサーバーで SSL 証明書を更新する場合は、「SSL 証明書の更新 (Update SSL certificate)」を参照してください。

手順

1. 暗号化キーの場所を選択します。

オプション	説明
ファイルストア	ファイルを使用して単一の暗号化キーを格納します。/keys ディレクトリーにある既存のファイルを使用することも、新しいファイルを生成することもできます。新しいファイルを生成する場合は、エイリアスを入力し、[生成] をクリックします。暗号化キーを含むファイルが作成されます。

自動翻訳

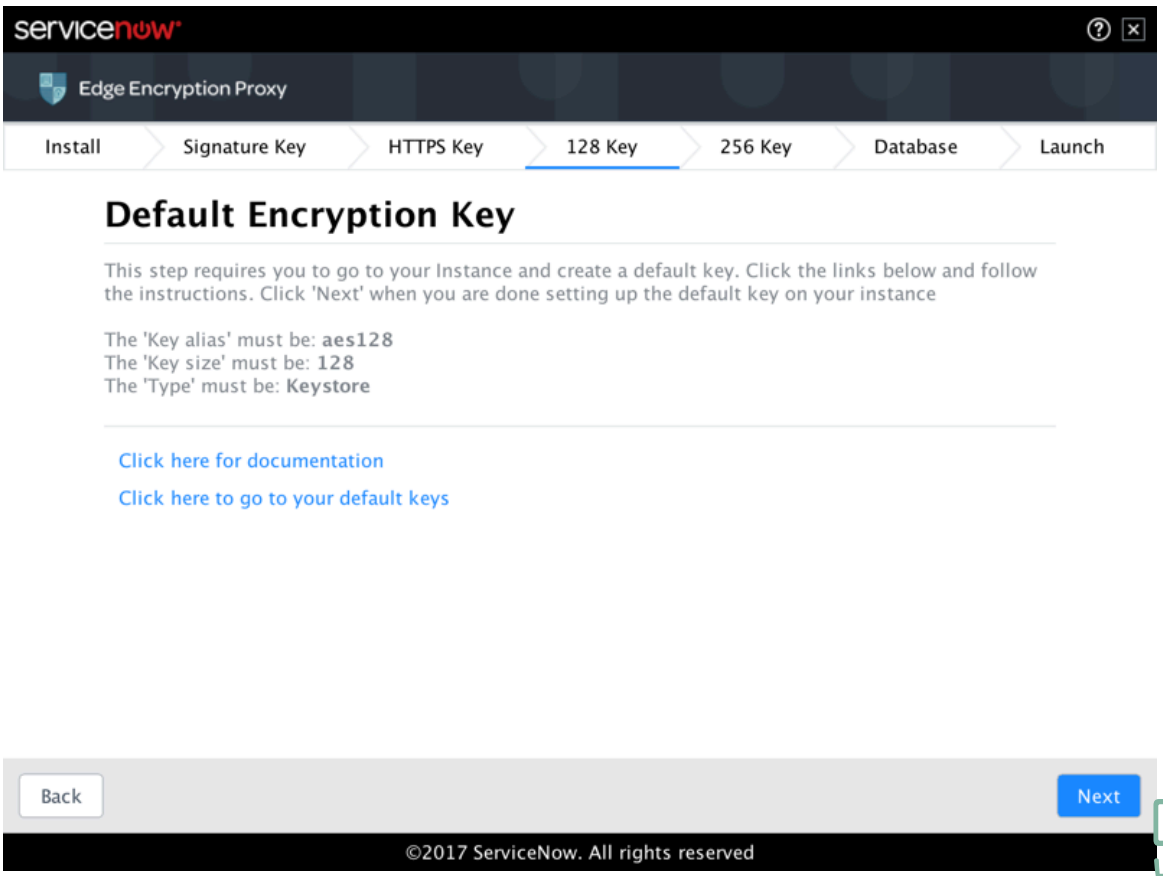
オプション	説明
	注: この選択肢は、格納場所と暗号化キーの両方を指定します。[ファイルストア] を選択した場合は、[次へ] をクリックして手順 5 に進みます。
新しい Java キーストアを作成	暗号化キーを格納するキーストアを作成します。
Java キーストアファイル	暗号化キーを既存の Java KeyStore ファイルに格納します。

2. [次へ **(Next)**] をクリックします。
3. 暗号化キーを選択または作成します。

オプション	説明
新しいキー	暗号化キーとエイリアスを作成します。 注: エイリアス名 (キー名、キーエイリアス) では、Java KeyStore の要件に従って、小文字と数字を使用する必要があります。keytool ユーティリティの詳細については、Java SE のドキュメント を参照してください。
既存のキーを使用	選択したキーストアで既存の暗号化キーを使用します。
既存のキーをインポート	別のキーストアから暗号化キーをインポートします。

4. [次へ **(Next)**] をクリックします。
5. インストーラーで定義した要件に従って、インスタンスのキーを設定します。
インスタンスのキーを設定するには、インスタンスに移動し、デフォルトキーを定義します。「インスタンスで暗号化キーを設定する」を参照してください。キーのエイリアス、サイズ、タイプがインストーラーで定義した要件と一致していることを確認します。

デフォルトキーの要件



6. インスタンスでキーを設定したら、インストーラーに戻り、[次へ] をクリックします。

AES 256 ビット暗号化キーを設定する

Edge プロキシインストーラーで AES 128 ビット暗号化キーを設定したら、データを暗号化するように AES 256 ビット暗号化キーを設定します (オプション)。

始める前に
必要なロール：admin

このタスクについて
暗号化キーは、/keys ディレクトリーにあるプレーン テキスト ファイルか、キーストア内部にある秘密キーです。AES 128 ビットおよび AES 256 ビットの暗号化キーに対してキーストアを使用する場合は、両方のキーで同じキーストアを使用する必要があります。AES 256 ビット暗号化キーを設定しない場合は、[スキップ] を選択して、プロキシサーバーのインストールを続行します。

Edge プロキシサーバーで SSL 証明書を更新する場合は、「SSL 証明書の更新 (Update SSL certificate)」を参照してください。

手順

1. 暗号化キーの場所を選択します。

オプション	説明
ファイルストア	ファイルを使用して単一の暗号化キーを格納します。/keys ディレクトリにある既存のファイルを使用することも、新しいファイルを生成することもできます。新しいファイルを生成する

オプション	説明
	場合は、エイリアスを入力し、[生成] を選択します。暗号化キーを含むファイルが作成されます。 注: この選択肢は、格納場所と暗号化キーの両方を指定します。[ファイルストア] を選択した場合は、[次へ] を選択して手順 5 に進みます。
新しい Java キーストアを作成 (Create New Java KeyStore)	暗号化キーを格納するキーストアを作成します。
Java キーストアファイル	暗号化キーを既存の Java KeyStore ファイルに格納します。

2. [次へ] を選択します。
3. 暗号化キーを選択または作成します。

オプション	説明
新しいキー	暗号化キーとエイリアスを作成します。 注: エイリアス名 (キー名、キーエイリアス) では、Java KeyStore の要件に従って、小文字と数字を使用する必要があります。keytool ユーティリティの詳細については、Java SE のドキュメント を参照してください。
既存のキーを使用	選択したキーストアで既存の暗号化キーを使用します。
既存のキーをインポート	別のキーストアから暗号化キーをインポートします。

4. [次へ] を選択します。
5. オプション: AES 256 ビット暗号化を使用する場合は、「AES 256 ビット暗号化キーを設定する」を参照してください。
6. AES 256 ビット暗号化を使用する場合は、AES 256 ビットのデフォルトの暗号化キーをインスタンスに設定する必要があります。

   これを行うには、インスタンスに移動し、デフォルトキーを定義します。「インスタンスで暗号化キーを設定する」を参照してください。キーのエイリアス、サイズ、タイプがインストーラーで定義した要件と一致していることを確認します。
7. インスタンスでキーを設定したら、インストーラーに戻り、[次へ] を選択します。

SSL 証明書の更新

Edge プロキシサーバーで SSL 証明書を更新する場合は、古い証明書を削除する必要があります。

始める前に
必要なロール：admin

このタスクについて
Edge プロキシサーバーで SSL 証明書を更新する場合は、古い証明書も削除する必要があります。削除しない場合、Edge プロキシサーバーが新しい証明書を使用するように設定されていても、古い証明書 (KeyStore ファイルのエイリアス形式) が引き続き使用されます。

手順

1. Edge プロキシサーバーで、Java KeyStore のエントリをリストします。

```
keytool -list -keystore keystore.jceks -storetype jceks -storepass MY_SUPER_PASSWORD
```

2. 古い SSL 証明書を削除します。

```
keytool -delete -alias MY_OLD_ALIAS -keystore keystore.jceks -storetype jceks -storepass
 MY_SUPER_PASSWORD
```

3. Java KeyStore に新しい SSL 証明書を追加します。

エッジ暗号化プロキシデータベースを設定する

順序保存の暗号化タイプまたは暗号化パターンを使用する場合は、オプションで エッジ暗号化 プロキシデータベースのプロパティを設定できます。

始める前に
必要なロール：admin

このタスクについて
順序保存の暗号化タイプまたは暗号化パターンを使用する場合は、ネットワーク内で MySQL データベースを実行する必要があります。このタスクではプロキシからデータベースに接続しますが、データベースのインストールまたは設定は行いません。

> **注:** 複数のプロキシサーバーを使用する場合は、すべてのプロキシサーバーで同じプロキシ データベースを使用する必要があります。インストーラーに入力する値は、すべてのプロキシサーバーについて同じにする必要があります。

手順

1. プロキシデータベースの場所であるデータベースの URL を確認または変更します。
2. [名前] フィールドに、プロキシ データベースの名前を入力します。
   デフォルト値は [*edgeencryption*] です。
3. プロキシ データベースにアクセスするためのユーザー名とパスワードを入力します。
4. [次へ **(Next)**] をクリックします。

エッジ暗号化プロキシサーバーを起動する

エッジ暗号化 プロキシのインストールと設定を終えたら、インストーラーからプロキシを起動できます。

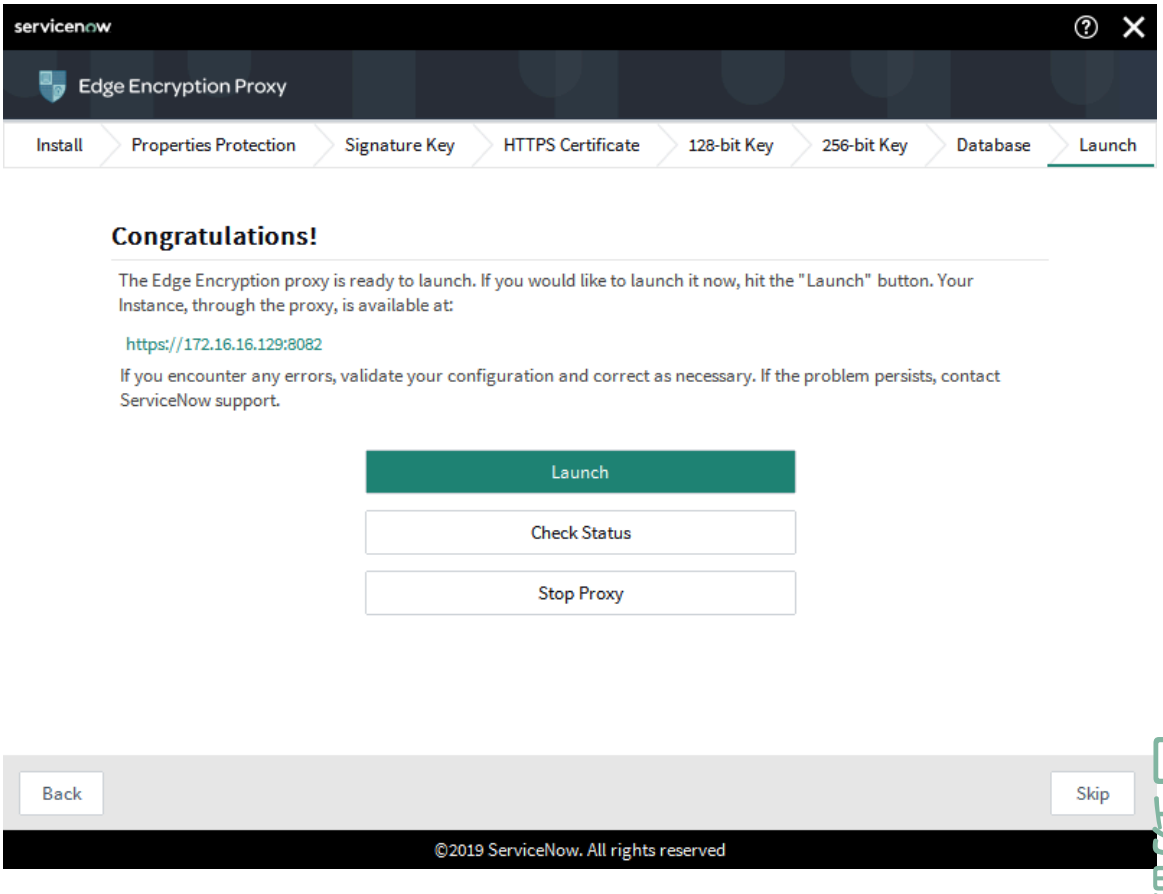
始める前に
必要なロール：admin

手順

1. インスタンスでキーを設定し、プロキシデータベースを設定したら、エッジ暗号化 プロキシインストーラーに戻り、[起動] をクリックします。
2. 問題が検出された場合、またはプロキシサーバーのステータスを確認する場合は、[ステータスのチェック] をクリックして、プロキシが実行されているかどうかを確認できます。

プロキシのステータスがメッセージに表示されます。

プロキシのステータス



次のタスク
エッジ暗号化 プロキシサーバーのインストールが正常に完了したら、プロキシサーバーの初期メモリ制限と上限メモリ制限を設定するの手順を実行します。

エッジ暗号化プロキシサーバーのインストールを検証してトラブルシューティングを行う

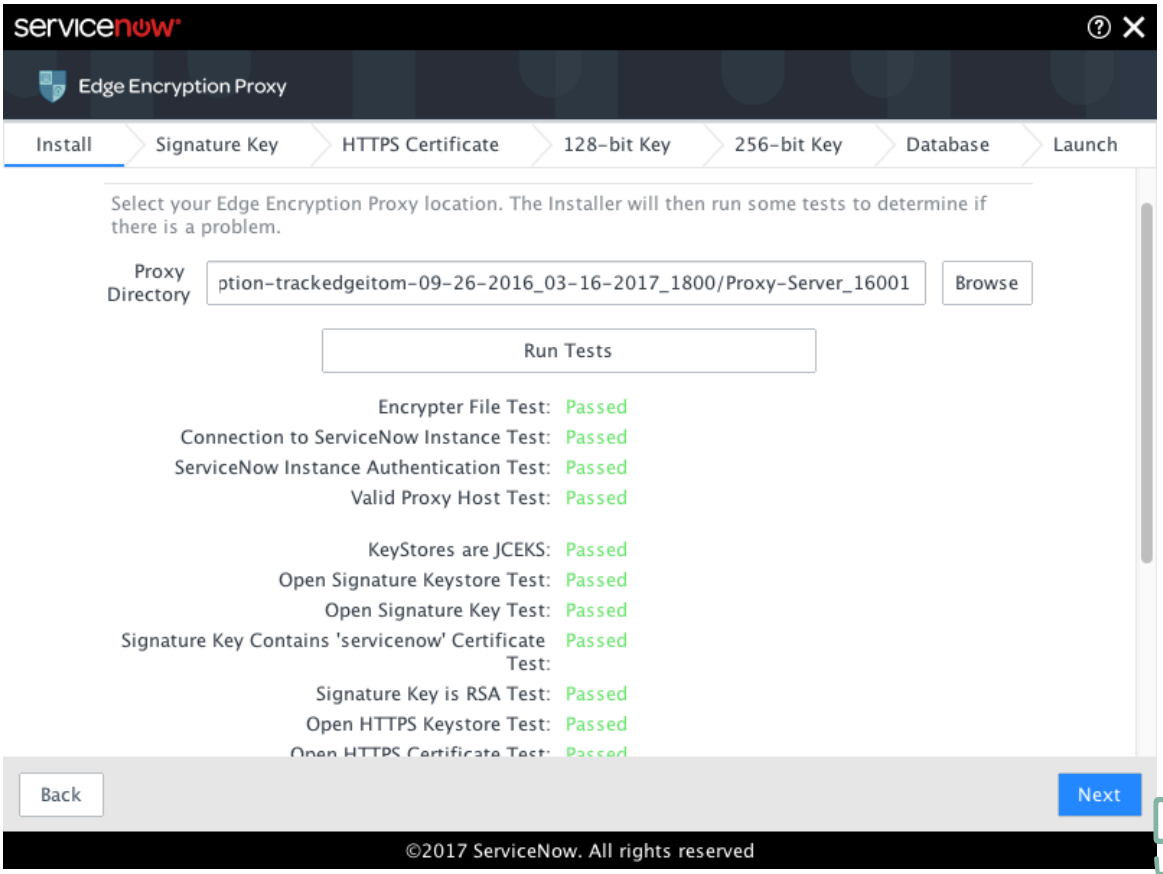
エッジ暗号化 プロキシをインストールした後、インストールを検証して、問題を特定したり、プロキシを開始および停止したりできます。

始める前に
必要なロール：admin

手順

1. エッジ暗号化 プロキシ インストーラーを開きます。
2. [インストールを検証] を選択します。
3. [プロキシのディレクトリー] をクリックし、プロキシのディレクトリーを選択します。
4. [テストを実行] をクリックします。
   テストの結果が表示されます。

エッジ暗号化プロキシのインストールのテスト



5. [次へ **(Next)**] をクリックします。
   問題が発生した場合は、インストーラーの画面を移動して設定を修正できます。問題が検出されない場合、インストーラーは [起動] ページに移動します。[起動] ページから、プロキシのステータスを確認したり、プロキシを停止したり、プロキシを起動したりできます。

コマンド ライン インストーラーを使用してエッジ暗号化プロキシサーバーをインストールする

ネットワークに複数の エッジ暗号化 プロキシサーバーを手動でインストールします。

始める前に

必要なロール：security_admin (ServiceNow インスタンス) およびローカルアドミニストレーター (ホスト マシン)。

順序保存の暗号化タイプまたは暗号化パターンを使用する場合は、ネットワークのマシンに MySQL データベースをセットアップします (まだ存在しない場合)。

> **注:** Unbound Technology 暗号化キーを エッジ暗号化 とともに使用する場合、Unbound クライアント マシンでコマンドライン インストーラを使用してプロキシサーバーをインストールします。エッジ暗号化 プロキシサーバーは、Unbound テクノロジ クライアントと同じマシンで実行する必要があります。

このタスクについて

最初に、単一の エッジ暗号化 プロキシサーバーをセットアップします。最初のプロキシサーバーが正常に実行されたら、1 つのインスタンスにさらにプロキシサーバーを追加して、最適な環境を確保します。必要な追加のプロキシサーバーの数の決定については、「エッジ暗号化環境のサイジング」を参照してください。

エッジ暗号化プロキシサーバーをインストールする (コマンド ライン インストーラー)

64 ビットの Windows または Linux コンピューターに エッジ暗号化 プロキシをインストールします。

始める前に
必要なロール：admin

インストーラーを実行するには、11.x バージョンシリーズの Java バージョン 11.0.6 以降が必要です。

このタスクについて
ターゲット マシンに適したコマンドを使用して、ネットワークのマシンに エッジ暗号化 プロキシサーバーをインストールします。エッジ暗号化 プロキシサーバーを Windows マシンにインストールする場合は、プロキシサーバーを Windows サービスとしてさらにインストールする必要があります。

エッジ暗号化プロキシサーバーをアップグレードすると、現在のインストール ディレクトリー配下の backup.dist-upgrade-<タイムスタンプ> ディレクトリーに、古いプロキシのバックアップが格納されます。このバックアップ ディレクトリーはアップグレード プロセス中に生成されるもので、古いプロキシの情報が格納されます。

コマンド ラインからアップグレードを実行すると、コマンドを実行したディレクトリーに dist-upgrade.log が生成されることがあります。dist-upgrade.log には、アップグレード プロセスのログが記録されています。

アップグレードに失敗した場合は、failed-backup.dist-upgrade-<timestamp> というディレクトリーが作成されます。また、元のプロキシ ディレクトリーの logs/wrapper.log にも、失敗に関する情報が格納される場合があります。

手順

1. インストール ディレクトリーを作成します。
2. エッジ暗号化 プロキシのアーカイブ ファイルをインストール ディレクトリーにコピーします。
3. ターミナルを開き、インストール ディレクトリーに移動します。

   注: Windows マシンにインストールする場合は、Windows コマンド プロンプトをアドミン権限で起動する必要があります。

4. ターゲットマシンに対して次のコマンドを実行し、構成に従って変数を変更します: `java -jar edgeencryption-<version>-all.jar -m install –n <proxy_name> --instancehost <host> -p <port> --protocol https –s <install_path>`

オプション	変数	説明
なし	version	現在の操作の実行に使用している エッジ暗号化 プロキシのバージョン番号。
-m	モード	ランタイムモード。オプションは、新しい Edge プロキシサーバーの場合は「インストール」、既存の Edge プロキシサーバーをアップグレードする場合は「dist-upgrade」です。

自動翻訳

オプション	変数	説明
-n	proxy_name	インストールされている エッジ暗号化 プロキシサーバーの名前。特定のプロキシ インスタンスを識別できるように、一意のproxy_nameを使用します。
--instancehost	host	ServiceNow インスタンスのホスト名 (たとえば、mycompany.service-now.com)。
-p	ポート	ServiceNowインスタンスがリッスンするポート。通常、セキュリティで保護された HTTPS 接続はポート **443** でリッスンし、HTTP 接続はポート **80** でリッスンします。
--protocol	protocol	バックエンド ServiceNow ホストに接続するときに、インストールされる Edge プロキシが使用するプロトコル。これは通常、ホストインスタンスがサポートするプロトコルに応じて 、HTTPS (セキュアな TLS 接続に優先) または **HTTP** (TLS を使用しない接続) になります。
-s	install_path	新しい Edge プロキシがインストールされているディレクトリまたはフォルダ (宛先ディレクトリ) へのパス。ディレクトリがまだ存在しない場合は、このコマンドで作成します。存在する場合は、既存のインストールを含んでいてはなりません。 このオプションをスキップすると、デフォルトのフォルダ名は、現在のディレクトリのproxy_nameとポート (EdgeProxy_443 など) から派生します。

注: ブラウザからコマンドをコピー アンド ペーストしないでください。コピー/ペースト操作によって予期しない文字がターゲット マシンに貼り付けられ、コマンドが正しく実行されないことがあります。マニュアルを参考にして手動でコマンドを入力するようにしてください。

ヘルプ画面を表示するには、 次のコマンドを –help オプションを指定して実行します: java -jar edgeencryption-<version>-all.jar --help

5. Windows マシンにインストールする場合は、エッジ暗号化 プロキシを Windows サービスとしてインストールします。

   a. オプション: 新しいプロキシで conf/wrapper.conf ファイルを開き、次のテーブルのプロパティを設定して、サービスの名前を変更します。

プロパティ	説明
wrapper.ntservice.name	エッジ暗号化 プロキシ サービスの一意の名前。
wrapper.ntservice.displayname	エッジ暗号化 プロキシ サービスの表示名。
wrapper.ntservice.description (オプション)	プロキシサーバーの説明。

   この手順を実行しない場合、エッジ暗号化 プロキシ サービスは エッジ暗号化という名前でインストールされます。

   b. ファイルを保存して閉じます。

   c. Windows のコマンドプロンプトを開き、cd ServerName_port/bin を実行します。

   d. edgeencryption.bat install を実行します。

結果
ProxyName_port ディレクトリーが現在のディレクトリーに作成されます。edgeencryption.properties ファイルが、コマンド ラインからのホスト、ポート、プロトコルの値に更新されます。

デジタル署名用の **RSA** キー ペアを作成して設定する

RSA キー ペアを作成します。プロキシサーバーはこれを使用してデジタル署名を作成し、暗号化のプロパティおよび設定への変更に対して署名を行うことができます。

始める前に
必要なロール：admin

デジタル署名を生成および検証するには、RSA キー ペアを生成して JCEKS Java KeyStore に格納する必要があり、このキー ペアを使用するように各プロキシを設定する必要があります。暗号化キー ペアの生成には keytool コマンドを使用します。

SElinux (CentOS) にインストールされたプロキシで keytool ユーティリティを使用するには、プロキシの java-installation ディレクトリーから共有ライブラリーをロードできるようにする必要があります。それには、次のコマンドを root として実行します。

```
chcon -R -t texrel_shlib_t proxy_install_dir/java/jre /lib
```

Java 1.8 バージョンの keytool ユーティリティを使用する必要があります。ユーティリティのコピーは <プロキシのインストール ディレクトリー>java/jre/bin/keytool にあります。

手順

1. プロキシのダウンロード ディレクトリーの KeyStore ディレクトリーに移動します。
2. デフォルトのパスワードを変更します。

   デフォルトパスワードは *changeme* です。

```
keytool -keystore keystore.jceks -storetype jceks -storepasswd -new <newpassword>
```

3. 暗号化キー ペアを作成します。

   注: keytool ユーティリティーでパスワードのプロンプトが表示されたときに、キーのパスワードを入力しないでください。

   次のコマンドを 1 行で入力します。

```
keytool -genkeypair -alias <key alias> -keyalg rsa -keystore keystore.jceks
-storetype jceks -storepass <keystore password> -keysize 2048
```

4. 暗号化プロキシのプロパティ ファイル (edgeencryption.properties) を更新します。

   a. <インストール ディレクトリー>/conf/ ディレクトリーに移動します。

   b. edgeencryption.properties ファイルを開きます。

   c. デジタル署名のプロパティを入力します。

      これらのプロパティはすべてのプロキシで同じにする必要があります。

5. edgeencryption.properties ファイルを保存して閉じます。

セキュア SSL 接続のための証明書をインポートして設定する

セキュア SSL 接続を使用するには、サーバー証明書をインポートして Java KeyStore に追加します。

始める前に

必要なロール：admin

Java KeyStore に追加する前に、サーバー証明書および一致する秘密鍵を取得する必要があります。

手順

1. openssl コマンドを使用して証明書署名要求 (CSR) を生成します。

```
openssl req -newkey rsa:2048 -keyout PRIVATEKEY.key -out MYCSR.csr
```

2. CSR (上記の例では MYCSR.csr) を認証局に送信して署名してもらいます。
3. openssl コマンドを使用して、インポートのための P12 キーストアを作成します。

```
openssl pkcs12 -export -in MYSIGNEDCERT.pem -inkey PRIVATEKEY.key -name shared >
 MY_SERVER.p12
```

4. 証明書と秘密鍵を jceks ファイルに格納します。

```
keytool -importkeystore -destkeystore keystore.jceks -deststoretype jceks -srckeystore
 MY_SERVER.p12 -srcstoretype pkcs12 -alias MYALIAS
```

   例で MYALIA として表示されているエイリアスは、任意の値にすることができます。このエイリアスは、<installation directory>/conf/ フォルダーにある edgeencryption.properties ファイルの edgeencryption.proxy.https.cert.alias プロパティで使用します。

5. Edge プロキシを停止して再起動します。

> **注:** 再起動中、プロキシサーバーは一時的にオフラインになります。その時間の長さは環境、およびプロキシ サービスの停止と再起動にかかる時間によって異なります。

キーストアと暗号化キーをセットアップする

エッジ暗号化 プロキシサーバーで使用されるキーストアと暗号化キーをセットアップします。

始める前に

必要なロール：security_admin

手順

1. 組織のニーズに応じて、使用する適切なタイプのキーストアを慎重に決定してください。

サポートされているキーストア	説明
ファイルストア	キーは、エッジ暗号化 プロキシサーバーからアクセスされるファイル システム内のファイルに格納されます。ファイルに格納された暗号化キーは暗号化されていないため、これらのファイルの保護についてはユーザーの責任になります。
Java KeyStore	Java KeyStore では次のようになります。 ○ キーは Java JCEKS KeyStore に格納されます。 ○ パスワードで保護されているため、ファイル システム内のファイルにキーを格納する方法よりも安全です。 ○ 複数のキーを格納できます。各キーをキー エイリアスで表すことで、複数のキーを簡単に管理できます。 エッジ暗号化 プロキシの keystore ディレクトリーには、keystore.jceks という名前の Java JCEKS KeyStore ファイルが付属しています。このキーストア ファイルには ServiceNow 公開キーが含まれています。この公開キーは、ServiceNow の署名が付いた暗号化ルールの検証に使用されます。
Enterprise Key Management (EKM)	**SafeNet KeySecure** キーは、SafeNet KeySecure のキー管理を使用して格納および取得されます。 Gemalto を使用してライセンスを確保し、ライブラリーをダウンロードし、ネットワーク内のホスト マシンに SafeNet KeySecure キーストアをインストールしてから、エッジ暗号化 プロキシサーバーでキーストアを設定する必要があります。 **Unbound Technology** base64 でエンコードされ、ラップされた暗号化キーは、テキストファイルとして エッジ暗号化 プロキシサーバーで保存されます。Unbound Technology の実装 (以前の Dyadic Security) は、ラッピングキーの制御を維持します。 エッジ暗号化 プロキシサーバーは、Unbound technology クライアントと同じマシンで実行する必要があります。

注: ベース システムの Java JCEKS KeyStore 以外のキーストアを使用する場合は、ServiceNow 公開キーをキーストアにインポートする必要があります。公開キーの別名は *servicenow* です。

2. ローカル ネットワークでキーストアと暗号化キーをセットアップします。

Java KeyStore キーストアをセットアップする

Java KeyStore キーストアを使用して暗号化キーを格納できます。

始める前に
必要なロール：admin

Java 1.8 バージョンの keytool ユーティリティを使用する必要があります。ユーティリティのコピーは <プロキシのインストール ディレクトリー>/java/jre/bin/keytool にあります。

このタスクについて
エッジ暗号化 プロキシの keystore ディレクトリーには、keystore.jceks という名前の Java JCEKS KeyStore ファイルが付属しています。このキーストア ファイルには ServiceNow 公開キーが含まれています。この公開キーは、ServiceNow の署名が付いた暗号化ルールの検証に使用されます。

手順

1. キーストアのプロパティをセットアップします。
   a. <インストール ディレクトリー>/conf/ ディレクトリーに移動します。
   b. edgeencryption.properties ファイルを開きます。
   c. Java KeyStore のプロパティを入力します。
2. edgeencryption.properties ファイルを保存して閉じます。

次のタスク
Java KeyStore をセットアップした後、「Java KeyStore keytool を使用して暗号化キーを作成する」の手順を実行します。

Java KeyStore keytool を使用して暗号化キーを作成する

暗号化プロキシの配布に付属している keytool を使用して、AES 128 ビットおよび AES 256 ビットの暗号化キーを作成できます。

始める前に
必要なロール：admin

Java 1.8 バージョンの keytool ユーティリティを使用する必要があります。ユーティリティのコピーは <プロキシのインストール ディレクトリー>/java/jre/bin/keytool にあります。

keytool ユーティリティの詳細については、Java SE のドキュメント を参照してください。

このタスクについて
注: Java KeyStore では、エイリアス名 (キー名、キー エイリアス) で小文字と数字を使用する必要があります。

手順

1. キーストア ディレクトリー <インストール ディレクトリー>/keystore/ に移動します。
2. 暗号化キーを作成するには、次のいずれかのコマンドを実行します。

注: キーストア ディレクトリー以外のディレクトリーからこれらのコマンドを実行する、つまり、前の手順をスキップする場合は、現在のディレクトリーからキーストア ディレクトリーへのパスを含めるように *-keystore* オプションを変更する必要があります。たとえば、現在 <インストールディレクトリー>\bin ディレクトリーにいる場合、オプションは -keystore ../keystore/keystore.jceks となります。

オプション	説明
AES 128	keytool -genseckey -alias 128bitkey -keyalg aes -keysize 128 -keystore keystore.jceks -storetype jceks
AES 256	keytool -genseckey -alias 256bitkey -keyalg aes -keysize 256 -keystore keystore.jceks -storetype jceks

デフォルトキーを割り当てるときに、インスタンスでエイリアスを追加します。

注: キーのパスワードはキーストアのパスワードと同じにする必要があります。

SafeNet KeySecure キーストアをセットアップする

SafeNet キーストアを使用している場合は、ライブラリ セットをプロキシ配布ディレクトリーにコピーします。

始める前に
必要なロール：admin

この手順を実行する前に、SafeNet キーストアをインストールしてセットアップする必要があります。ライブラリをダウンロードするために Thales を使用してライセンスを確保します。

注: IngianNAE バージョン 8.12 の場合は、commons-collections.jar ファイルもダウンロードする必要があります。

このタスクについて
注: Linux では、ファイルパスにスラッシュ (/) を使用します。

手順

1. <インストール ディレクトリー>/conf/ ディレクトリーに移動し、edgeencryption.properties ファイルを開きます。
2. SafeNet キーストアのプロパティを入力します。

注: ユーザー名/パスワード認証とクライアント証明書認証のどちらかを使用して SafeNet キーストアを設定することはできますが、両方を組み合わせることはできません。

Example

ユーザー名とパスワードの認証を使用する SafeNet キーストアの例

```
edgeencryption.nae.retries = 3
edgeencryption.nae.enabled = true
edgeencryption.nae.server = url
edgeencryption.nae.port = 9000
edgeencryption.nae.protocol = ssl
edgeencryption.nae.keystore.path = keystore/safenet_truststore
edgeencryption.nae.keystore.password = password
```

自動翻訳

```
edgeencryption.nae.user = safenet_user
edgeencryption.nae.password = safenet_password
```

Example

クライアント証明書の認証を使用する SafeNet キーストアの例。この認証方法では、プロパティファイルに SafeNet サーバーのユーザー名とパスワードを格納する必要がなくなります。

```
edgeencryption.nae.retries = 3
edgeencryption.nae.enabled = true
edgeencryption.nae.server = url
edgeencryption.nae.port = 9000
edgeencryption.nae.protocol = ssl
edgeencryption.nae.keystore.path = keystore/safenet_clientcert
edgeencryption.nae.keystore.password = password
edgeencryption.nae.client.certificate = cert_name
```

3. SafeNet キーストアにキーを追加または作成します。
   デフォルトキーを割り当てるときに、インスタンスでキー名 (エイリアス) を追加します。
4. edgeencryption.properties ファイルを保存して閉じます。

Kingston 以前から London 以降へのアップグレード

Edge でのキーストレージに SafeNet NAE サーバーを使用する場合は、プロキシを Kingston 以前から London 以降にアップグレードする前に、Gemalto SafeNet クライアントの ProtectApp JAR ファイルをコピーし、新しいプロパティを追加する必要があります。

始める前に
必要なロール：admin

このタスクについて
注: Linux では、ファイルパスにスラッシュ (/) を使用します。

手順

1. 次のファイルを <installation directory>/lib から <installation directory>/nae ディレクトリーにコピーします。
   - commons-collections<version>.jar
   - ingrianlog4j-api-<version>.jar
   - ingrianlog4j-core-<version>.jar
   - ingrianNAE-<version>.jar
2. プロキシの現在のバージョン (未アップグレード) で、<installation directory>/conf/edgeencryption.properties ファイルに次の 2 つのプロパティを追加して更新します。
   - *edgeencryption.ekm.provider.classname* = com.snc.edgeencryption.encryption.CloudEdgeNaeKeyProvider
   - *edgeencryption.thirdparty.vendor.library.path* = <手順 1 の JAR ファイルをコピーしたディレクトリへのディレクトリパス>

     注: Java 11 の *edgeencryption.thirdparty.vendor.library.path*
3. 変更内容を保存します。
4. London 以降へのアップグレードを続行します。

Unbound Technology のキーのセットアップ

エッジ暗号化 で Unbound Technology (旧 Dyadic Security) のキーを使用するには、base64 エンコードされ、ラップされた暗号化キーをテキストファイルとして エッジ暗号化 プロキシサーバーに格納し、ラッピングキーエイリアスを指定します。Unbound Technology の実装によって、ラッピングキーのコントロールが維持されます。

始める前に

必要なロール：security_admin

Unbound Technology の実装で、ラッピング キーとラップされたキーの両方を識別します。ラッピングとパディングには RSA/ECB/OAEPWITHSHA-256ANDMGF1PADDING アルゴリズムを使用します。ラップされたキーを、base64 エンコードされたテキスト形式でエクスポートします。キー エイリアスを使用し、ファイル拡張子のない名前としてファイルを保存します。

注: Unbound Technology 暗号化キーを エッジ暗号化 とともに使用する場合、Unbound クライアント マシンでコマンドライン インストーラを使用してプロキシサーバーをインストールします。エッジ暗号化 プロキシサーバーは、Unbound テクノロジ クライアントと同じマシンで実行する必要があります。

手順

1. ラップされた暗号化キーを、base64 エンコードされたテキスト形式で <プロキシのインストール ディレクトリー>/keys ディレクトリーに追加します。
   ファイルの名前は、ファイル拡張子のないキー エイリアスにする必要があります。
2. edgeencryption.properties ファイルを更新します。
   a. <プロキシのインストール ディレクトリー>/conf ディレクトリーに移動します。
   b. edgeencryption.properties ファイルを開きます。
   c. ファイルストアのプロパティを入力し、*edgeencryption.keyfile.directory* の値を keys に設定します。

      このプロパティの指示によって、プロキシサーバーは <Java ホーム ディレクトリ>/keys ディレクトリで暗号化キーを探します。

      エッジ暗号化 のプロパティの詳細については、「エッジ暗号化プロキシサーバーのプロパティ」を参照してください。
   d. Dyadic プロバイダー設定のプロパティをコメント解除し、*edgeencryption.ekm.provider.rsa.wrapping.key.alias* の値を Unbound 実装のラッピングキーエイリアスに設定します。
   e. ファイルを保存して閉じます。

次のタスク

暗号化キー エイリアスをインスタンスに追加します。暗号化キーエイリアスは、<プロキシインストールディレクトリ>/keys ディレクトリに追加されているラップされた暗号化キーのファイル名です。たとえば、ディレクトリ内のファイルの名前が myunboundkey の場合、この名前を [キーエイリアス] フィールドに追加します。「インスタンスで暗号化キーを設定する」を参照してください。

ファイルに格納する暗号化キーを作成する

シンプルなテキスト ファイルをキーストアとして使用できます。ファイルごとに 1 つずつ暗号化キーを保持します。

始める前に
必要なロール：admin

このタスクについて
この手順では、キー ストレージと暗号化キーの両方が作成されます。

注: キー ファイルの名前は、インスタンス内の暗号化キー テーブルで指定されたキー エイリアスに合わせる必要があります。「インスタンスで暗号化キーを設定する」を参照してください。

手順

1. プロキシサーバーのインストール ディレクトリーの /keys フォルダーにファイルを作成します。
2. 暗号化キーをファイルに追加します。

オプション	説明
AES 128	ファイルに暗号化キー (正確な 16 バイト) を配置します。
AES 256	ファイルに暗号化キー (正確な 32 バイト) を配置します。

3. edgeencryption.properties ファイルを更新します。
   a. <インストール ディレクトリー>/conf/ ディレクトリーに移動します。
   b. edgeencryption.properties ファイルを開きます。
   c. ファイルストアのプロパティを入力します。
   d. ファイルを保存して閉じます。

インスタンスで暗号化キーを設定する

エッジ暗号化 では、プロキシをオフラインにせずに暗号化キーを管理するためのツールが提供されています。

始める前に
必要なロール：security_admin

インスタンスで新しい暗号化キーをセットアップする前に、次の手順を実行します。

1. 暗号化キーを作成します。
2. すべての暗号化プロキシで新しいキーを使用できるようにします。各プロキシにファイルまたは Java KeyStore ファイルをコピーします。または、各プロキシから Java KeyStore または Enterprise Key Management (EKM) デバイスにアクセスできることを確認します。

このタスクについて
キー エイリアスは一意である必要があります。キー エイリアスはそれぞれ、各プロキシで同じキーのサイズとタイプにする必要があります。そうしないと、キーをデフォルトとして割り当てることはできません。

手順

1. 移動先 すべて > エッジ暗号化の設定 > 暗号化キーの設定 > キーの設定.
2. フォームの [新しいキーの追加] セクションで、次の手順を実行して新しいキーを追加します。

> **重要:** SafeNet でバージョニングされたキーを使用している場合は、[キーバージョン] 用の追加の列が表示されます。[キーバージョン] は編集できません。関連リンクで [最新のキーバージョンを取得] リンクをクリックして、Edge プロキシから各キーの最新バージョンを取得します。

リストで左側の列に **X** 印のある行を削除することができます。以前にデフォルトとして使用されたキー、または [ステータス] が [利用可能] のキーは、削除できません。

a. [新規行を挿入] という行をダブルクリックします。

b. 編集ボックスで、キーの名前を入力し、チェックマークをクリックします。

キー エイリアスは小文字と数字です。[更新] をクリックすると大文字が小文字に変わります。キー エイリアスは一意である必要があります。

> **注:** Unbound Technology のキーを使用している場合は、暗号化キー エイリアスを追加します。 暗号化キーエイリアスは、<プロキシインストールディレクトリ>/keys ディレクトリに追加されているラップされた暗号化キーのファイル名です。たとえば、ディレクトリ内のファイルの名前が myunboundkey の場合、この名前を [キーエイリアス] フィールドに追加します。

c. 同じ行で、[キー サイズ] 列をダブルクリックします。

d. 選択ボックスで、キー サイズとして **[128** ビット] または **[256** ビット] を選択し、チェックマークをクリックします。

e. 同じ行で、[タイプ] 列をダブルクリックします。

f. 選択ボックスで、キー タイプとして [ファイル]、[キーストア]、**[SafeNet]**、または **[Unbound]** を選択し、チェック マークをクリックします。

g. キーの追加を終えたら、[次の手順] をクリックします。
次に進む前に、各キーのエイリアス、キー サイズ、およびキー タイプを指定する必要があります。

3. フォームの [キーのステータス] セクションで、キーの [ステータス] を確認し、[利用可能] であることを確認します。

4. キーが [利用可能] な場合は、[次の手順] をクリックします。
これには数分かかる場合があります。

> **注:** SafeNet でバージョニングされたキーを使用している場合は、[キーバージョン] 用の追加の列が表示されます。[キーバージョン] は編集できません。

インスタンスでは、どのプロキシでも利用可能なすべての暗号化キーのステータスが追跡されます。すべてのプロキシでキーが使用可能になると、そのステータスが [利用可能] になります。数分経ってもステータスが変わらない場合は、キーがすべてのプロキシで利用可能かどうかを確認してください。ステータスが [利用不可] のままである場合は、1 つ以上のプロキシにキーがありません。

暗号化キーの状況

ステータス	説明
利用可能	すべてのオンライン プロキシにキーがあります。
利用不可	これは新しいキーであり、プロキシがまだキーをロードしていないか、少なくとも 1 つのプロキシがキーをロードできませんでした。

5. フォームの [デフォルトキーの変更] セクションで、次のいずれかの操作を行います。
   - キー エイリアスを入力します。
   - 虫眼鏡アイコンをクリックし、エイリアスを選択します。

   注: SafeNet でバージョニングされたキーを使用している場合は、[キーバージョン] 用の追加のフィールドが表示されます。[キーバージョン]はグレー表示され、編集することはできません。最新のキーバージョンのみを選択してください。以前のバージョンを選択した場合は、[更新] または [次のステップ] をクリックすると、次のメッセージが表示されます。

   ```
   One of the default keys chosen is not the latest version available for the key. Please use the latest version.
   ```

   デフォルトのキーが SafeNet キーの最新バージョンでない場合は、関連リンクに [デフォルトキーを最新バージョンに更新] リンクが表示されます。このリンクをクリックして、最新バージョンを使用するようにデフォルトキーを更新してください。

6. フォームの [キー ローテーションのスケジュール] セクションで、新しい暗号化キーを使用して既存のデータを暗号化するように、一括キー ローテーション ジョブまたは単一キー ローテーション ジョブをスケジュールします。

   一括キー ローテーション ジョブまたは単一キー ローテーションジョブを実行しないと、データに再度アクセスするまで、既存のデータは古いキーで暗号化されたままになります。

エッジ暗号化プロパティ ファイルでその他のプロパティを設定する

ネットワークに エッジ暗号化 プロキシサーバーをインストールしてキーストアとキーをセットアップしたら、エッジ暗号化 のその他のプロパティを設定します。

始める前に

必要なロール：admin

手順

1. <インストールディレクトリー>/conf/edgeencryption.properties ファイルを開き、次の エッジ暗号化 プロキシサーバープロパティを設定します。
   - ターゲット (インスタンス) のプロパティ
   - ユーザー アカウントのプロパティ
   - プロキシのプロパティ
   - 順序保存の暗号化タイプまたは暗号化パターンを使用する場合は、プロキシデータベースのプロパティを設定します。
   - クリアー テキストおよび静的 IV のプロパティ
2. ファイルを保存して閉じます。

Web プロキシを設定する

ネットワークで Web プロキシを使用している場合は、Web プロキシを使用するように エッジ暗号化 プロキシをセットアップできます。

始める前に
必要なロール：admin

このタスクについて
ネットワークで Web プロキシを使用していない場合は、設定ファイルで Web プロキシ プロパティをコメント アウトしたままにしておきます。

エッジ暗号化 プロキシサーバーは、Web プロキシへの HTTP 接続と、Web プロキシによるベーシック認証をサポートしています。

手順

1. <インストール ディレクトリー>/conf/ ディレクトリーに移動します。
2. edgeencryption.properties ファイルを開きます。
3. Web プロキシ プロパティを設定します。
4. edgeencryption.properties ファイルを保存して閉じます。
5. Web プロキシで顧客固有のサーバー証明書を使用している場合は、 エッジ暗号化 プロキシサーバーで使用される JVM にその証明書を追加して、Web プロキシと エッジ暗号化 プロキシサーバー間の信頼を確立します。
   a. cd コマンドを使用して、<Java home directory>/jre/lib/security/cacerts に移動します。
   b. コマンド keytool -keystore cacerts -importcert -alias <chooseAlias> -file <certificateFile> を実行します。

プロキシサーバーの初期メモリ制限と上限メモリ制限を設定する

初期メモリ制限と上限メモリ制限を設定して、プロキシサーバーが利用できるメモリ量を指定します。これらの制限を設定することで、エッジ暗号化 の実装でパフォーマンスの問題が回避されます。

始める前に
必要なロール：admin

このタスクについて
原則として、初期メモリ制限と上限メモリ制限の両方を同じ値に設定してください。どのマシンでも、2 GB の物理メモリをオペレーティング システム (OS) に割り当てます。次に、初期メモリ制限プロパティと上限メモリ制限プロパティを使用して、残りの物理メモリをヒープに割り当てます。たとえば、8 GB のメモリを搭載したマシンでは、OS に 2 GB を割り当て、残りの 6 GB (6144 メガバイト) を初期および上限のメモリに割り当てます。

> **重要:** エッジ暗号化 プロキシサーバーが実行中の場合は、これらのプロパティを更新した後にプロキシサーバーを停止して再起動する必要があります。

手順

1. プロキシサーバーのディレクトリーで、<インストール ディレクトリー>/conf/wrapper.conf を開きます。
2. 初期メモリ制限を設定するには、ファイルの最後に次の行を追加します。

```
wrapper.java.additional.<number>=-Xms<min_memory_in_MB>m
```

   <数字> を、wrapper.conf ファイルで定義されている wrapper.java.additional.<数字> プロパティのシーケンスで次に利用可能な <数字> に設定します。

Example
たとえば、次に示す wrapper.java.additional.<数字> プロパティのリストがあるとします。

```
wrapper.java.additional.1=
wrapper.java.additional.2=
```

このリストの最大の <数字> は **2** です。wrapper.java.additional.<数字>=-Xms<min_memory_in_MB>m 行を追加するときは、<数字> を次に利用可能な数字である **3** に設定します。

重要: 番号のシーケンスにはギャップを残さないでください。

<min_memory_in_MB> を、2 GB のメモリを OS に割り当てた後に残っているメモリのメガバイト数に設定します。

3. 上限メモリ制限を設定します。
上限メモリ制限はベース システムで設定されていないため、プロキシサーバーは利用可能なすべてのメモリを使用できます。他のサービスがサーバー上で実行されている場合は、上限メモリ制限を設定することもできます。

ファイルの最後に次の行を追加します。

wrapper.java.additional.<数字>=-Xmx<max_memory_in_MB>m

<数字> を、wrapper.conf ファイルで定義されている wrapper.java.additional.<数字> プロパティのシーケンスで次に利用可能な <数字> に設定します。

Example
たとえば、次に示す wrapper.java.additional.<数字> プロパティのリストがあるとします。

```
wrapper.java.additional.1=
wrapper.java.additional.2=
```

このリストの最大の <数字> は **2** です。wrapper.java.additional.<number>=-Xmx<max_memory_in_MB>m 行を追加するときは、<数字> を次に利用可能な数字である **3** に設定します。

注: 番号のシーケンスにはギャップを残さないでください。

<max_memory_in_MB> を、2 GB のメモリを OS に割り当てた後に残っているメモリのメガバイト数に設定します。

4. ファイルを保存して閉じます。

Example: 例：プロキシサーバーの初期および上限のメモリ制限を設定する

```
wrapper.java.additional.1 = -Djava.io.tmpdir=../tmp
wrapper.java.additional.2 = -Dcloudedge.home.dist=..
# must ensure UTF8 encoding when running on Windows
wrapper.java.additional.3 = -Dfile.encoding=UTF8
# additional properties for heap settings
wrapper.java.additional.4 = -Xms6144m
wrapper.java.additional.5 = -Xmx6144m
```

次のタスク
エッジ暗号化プロキシを起動する。

エッジ暗号化プロキシを起動する

エッジ暗号化 プロキシのインストールと設定を終えたら、コマンド ラインからプロキシを起動できます。

始める前に
必要なロール：admin

暗号化プロキシを起動する前に、次のことを確認してください。

- エッジ暗号化 プラグインがインスタンスで有効になっている。
- このマシンの edgeencryption.properties ファイルがすでに設定されている。
- 順序保存の暗号化タイプまたは暗号化パターンを使用する場合は、プロキシ データベースが実行されている。

注: 初めて edgeencryption.properties ファイルをセットアップするとき、またはプロパティを変更するときに、パスワードの暗号化プロパティの設定を望まない場合があります。すべてが正常に機能していることを確認したら、パスワードの暗号化プロパティを設定し、プロキシをシャット ダウンしてから、プロキシを再起動することができます。

手順

1. プロキシサーバーを実行します。

オプション	説明
Linux マシン	**a.** cd ServerName_port を実行します。 **b.** ./startup.sh を実行します。
Windows マシン	コマンド ラインから admin として次の手順を実行します。 **a.** cd ServerName_port/bin を実行します。 **b.** edgeencryption.bat start を実行します。

2. プロキシサーバーのログを調べて、プロキシが実行されていることを確認します。

プロパティファイルでのパスワードの難読化

edgeencryption.properties ファイルでパスワードを難読化して、クリアテキストのパスワードを公開せずにプロパティファイルを共有することができます。

始める前に
必要なロール：admin

このプロパティを設定する前に、エッジ暗号化 プロキシサーバーがセットアップされていて正常に実行されていることを確認してください。このプロパティを設定する前に、「エッジ暗号化プロキシを停止する」の手順を実行してください。

このタスクについて
このプロパティを設定すると、最初の起動時に接続やアクセスの問題が発生した場合に、それらのデバッグが困難になることがあります。プロキシを正常にセットアップして、テストに成功した後でのみ、本番環境でこのプロパティを設定してください。

手順

1. <インストール ディレクトリー>/conf/ ディレクトリーに移動します。
2. conf ディレクトリーで、パスフレーズとして使用できる複雑な文字列またはフレーズを含むテキスト ファイルを作成します。プロキシはこのファイルを使用して、edgeencryption.properties ファイルのパスワードを難読化します。
   パスフレーズは、パスワード自体と関係のないランダムかつ複雑なフレーズにしてください。
3. edgeencryption.properties ファイルを開きます。
4. パスワードの暗号化プロパティを設定します。
5. edgeencryption.properties ファイルを保存して閉じます。

次のタスク
このプロパティを設定した後に、「エッジ暗号化プロキシを起動する」の手順を実行できます。

プロキシの手動追加

最初の エッジ暗号化 プロキシを正しく設定してテストを終えたら、Linux または Windows マシンで追加のプロキシをセットアップできます。複数のプロキシを同じマシンにインストールすることはお勧めしません。

始める前に
必要なロール：admin

このタスクについて
追加したマシンにさらにプロキシサーバーを追加して、最適な環境を確保します。必要な追加のプロキシ数の決定については、「エッジ暗号化環境のサイジング」を参照してください。

> **注:** 必ずすべてのプロキシに同じ暗号キーと同じ RSA キー ペアを持たせてください。これらは、暗号化設定と暗号化ルールへのデジタル署名に使用されます。プロキシ データベースをインストールの一部としてセットアップする場合、すべてのプロキシで同じプロキシ データベースを使用する必要があります。

手順

1. 適切なコマンドを使用してプロキシをインストールします。
   詳細については、「エッジ暗号化 プロキシサーバーをインストールする (インタラクティブ インストーラー)」を参照してください。
2. すべての暗号化キーと edgeencryption.properties ファイルを最初のプロキシから新しいプロキシにコピーします。
   暗号化キーは、プロキシのキーストア、/keys ディレクトリー、または SafeNet KeySecure キーストアにあります。
3. 新しいプロキシで edgeencryption.properties ファイルを開きます。
4. 次のプロパティを変更します。

プロパティ	説明
edgeencryption.proxy.name	プロキシサーバーの一意の名前。
edgeencryption.proxy.host	プロキシを実行しているコンピューターのサーバー名、IP アドレス、または完全修飾ドメイン名。

プロパティ	説明
edgeencryption.proxy.http.port	HTTP 通信のプロキシのポート。マシン上のすべてのプロセス全体で一意である必要があります。
edgeencryption.proxy.https.port	HTTPS 通信のプロキシのポート。マシン上のプロセス全体で一意である必要があります。

5. Windows マシンにプロキシサーバーをインストールする場合は、新しいプロキシで conf/wrapper.conf ファイルを開き、次の表に示すプロパティを追加することで、サービスの名前を変更する必要があります。

   注: この手順は、プロキシサーバーを起動する前に実行する必要があります。

プロパティ	説明
wrapper.ntservice.name	エッジ暗号化 プロキシ サービスの一意の名前。
wrapper.ntservice.displayname	エッジ暗号化 プロキシ サービスの表示名。
wrapper.ntservice.description (オプション)	プロキシサーバーの説明。

6. ファイルを保存して閉じます。
7. 適切なコマンドを使用してプロキシを起動します。
   詳細については、「エッジ暗号化プロキシを起動する」を参照してください。

エッジ暗号化 プロキシサーバーを認証する

エッジ暗号化 がプロキシサーバーからの要求を処理できるよう、プロキシサーバーが信頼できるソースであることを指定します。

始める前に

プロキシサーバーが認証されていない場合は、次のメッセージがコンソールログに含まれています。

```
WARN This Edge Encryption proxy has not yet been authenticated by the instance.
Please navigate to the matching Proxy record on your ServiceNow instance and authenticate it.
```

このプロキシにアクセスしようとすると、「This site can't be reached」というメッセージが表示されます。

アップグレードプロセス中にプロキシの運用ステータスを維持するために、プロキシ更新が成功するまで認証は必要ありません。

必要なロール：admin

手順

1. 移動先 すべて > エッジ暗号化の設定 > プロキシ.
2. プロキシを選択し、[認証] をクリックします。

結果

プロキシが [未認証] から [処理待ち]、[認証済み] へと移動します。認証を開始すると、ステータスが [未認証] から [処理待ち] に変化します。認証が完了すると、ステータスが [処理待ち] から [認証

済み] に変化してプロキシにアクセスできるようになり、エッジ暗号化 はプロキシからの要求を受け入れられるようになります。

注: プロキシを停止して再起動する場合、プロキシは [認証済み] のままになり、正常に再起動します。

エッジ暗号化プロキシを停止する

コマンド ラインから エッジ暗号化 プロキシを停止できます。

始める前に
必要なロール：admin

手順

1. プロキシサーバーを停止します。

オプション	説明
Linux マシン	./shutdown.sh を実行します。
Windows マシン	edgeencryption.bat stop を実行します。 Windows サービスを削除するには、edgeencryption.bat remove を実行します。

2. プロキシサーバーのログを調べて、プロキシが停止したことを確認します。

Linux でエッジ暗号化プロキシをアンインストールする

エッジ暗号化 プロキシをアンインストールすることができます。プロキシをアップグレードする場合、現在のバージョンをシャット ダウンしてアンインストールする必要はありません。

始める前に
必要なロール：admin

エッジ暗号化 プロキシを実行しているコンピューターにアクセスできる必要があります。

このタスクについて
エッジ暗号化 プロキシをシャット ダウンする前に、プロキシを使用してインスタンスに接続しているユーザーがいないことを確認します。

Linux 上で実行される暗号化プロキシは、単一のプロセスとして動作します。暗号化プロキシを別のホスト マシンに再展開する、プロキシのバージョンを更新する、Java のバージョンを更新する、暗号化プロキシを複数のプロキシサーバーに展開するときに暗号化プロキシの一意の名前を変更する、などの作業を実行する際に、このプロセスを終了できます。

手順

1. edgeencryption.properties ファイルを保存してから配布ディレクトリーを削除することもできます。
2. shutdown.sh シェル スクリプトを実行します。
3. プロキシサーバーのログを調べて、プロキシサーバーがシャット ダウンされたことを確認します。
4. 配布フォルダー内のファイルを削除します。

Windows でエッジ暗号化プロキシをアンインストールする

エッジ暗号化 プロキシをアンインストールすることができます。プロキシをアップグレードする場合、現在のバージョンをシャット ダウンしてアンインストールする必要はありません。

始める前に

必要なロール：admin

エッジ暗号化 プロキシを実行しているコンピューターにアクセスできる必要があります。

エッジ暗号化 プロキシをシャット ダウンする前に、プロキシを使用してインスタンスに接続しているユーザーがいないことを確認します。

手順

1. edgeencryption.properties ファイルを保存してから配布ディレクトリーを削除することもできます。
2. `edgeencryption.bat stop` を実行します。
3. `edgeencryption.bat remove` を実行します。
4. プロキシサーバーのログを調べて、プロキシサーバーがシャット ダウンされたことを確認します。
5. 配布フォルダー内のファイルを削除します。

エッジ暗号化 でのマルチプロバイダー SSO のセットアップ

マルチプロバイダー SSO をセットアップして、エッジ暗号化 プロキシサーバーの URL またはインスタンスの URL を経由するログインを有効にします。エッジ暗号化 を有効にした状態でマルチプロバイダーのシングルサインオン (SSO) を実装する場合、一部のユーザーは エッジ暗号化 プロキシサーバーを経由してインスタンスにログインする必要がありますが、他のユーザーはその必要はありません。

始める前に

- Integration - Multiple Provider Single Sign-On Installer プラグイン (com.snc.integration.sso.msoi.installer) を有効にします。
- エッジ暗号化 プラグイン (com.glide.edgeencryption) を有効にし、1 つまたは複数のプロキシサーバーがネットワークにセットアップされていることを確認します。
- ユーザーがマルチプロバイダー SSO を使用してログインする際に経由する エッジ暗号化 プロキシサーバーの URL を決定します。エッジ暗号化 プロキシサーバーの URL を決定するには、「エッジ暗号化のインストール」を参照してください。

必要なロール：admin

このタスクについて

ログインするユーザーは、Edge プロキシを使用するかどうかにかかわらず、適切な URL を使用してログインする必要があります。

- すべてのユーザーを エッジ暗号化 プロキシサーバー経由でルーティングする場合は、ID プロバイダー レコードをセットアップし、**[ServiceNow** ホームページ]、[エンティティ ID/発行者]、[対象者 **URI]** の各フィールドでプロキシサーバー URL を定義します。
- 一部のユーザーをプロキシサーバー経由でルーティングし、他のユーザーをインスタンスにルーティングするには、2 つの ID プロバイダー レコードを作成します。どちらのレコードについても **[ID** プロバイダー **URL]** フィールドで同じ値を使用します。ただし、レコードの一方はプロキシサーバー経由でルーティングし、もう一方はインスタンスにルーティングします。

自動翻訳

- インスタンス名を介したログイン：Edge 以外のプロキシ用 IdP レコードの https://<インスタンス名>.service-now.com/login_with_sso.do?glide_sso_id=<sys_id
- Edge プロキシを介したログイン：Edge プロキシ用 IdP レコードのhttps://<edge ホスト名>:<ポート>/login_with_sso.do?glide_sso_id=<sys_id

手順

1. ID プロバイダー レコードで ID プロバイダー URL の重複を有効にします。
   一意性の制約により、2 つの異なる ID プロバイダー レコードでの ID プロバイダー URL の重複は防止されています。フィールドを False に設定することで、複数の ID プロバイダー レコードで ID プロバイダ URL の重複を有効にすることができます。
   a. 移動先 システム定義 > ディクショナリ.
   b. ID プロバイダー テーブル [saml2_update1_properties] にある **[idp]** フィールドの定義レコードを開きます。
   c. [一意] フィールドを追加するようにフォームを設定します。
   d. [一意] フィールドの値が **False** に設定されていることを確認します。
2. 移動先 マルチプロバイダー **SSO** > **ID** プロバイダー.
3. 同じ ID プロバイダーに対して 2 つの ID プロバイダー レコードを作成します。一方ではインスタンスの URL を使用し、もう一方では エッジ暗号化 プロキシサーバーの URL を使用します。
   ID プロバイダーレコードを作成するには、「外部 ID プロバイダーの作成」を参照してください。
   a. エッジ暗号化 プロキシサーバーの URL については、次の値をフォームに入力します。

フィールド	値
ID プロバイダー URL	IdP メタデータからインポートします。
ServiceNow のホームページ	プロキシサーバーのホームページの URL。例：https://<プロキシのホスト名>:<ポート>/navpage.do
エンティティ ID/発行者	https://<プロキシのホスト名>:<ポート>
対象者 URI	https://<プロキシのホスト名>:<ポート>

   b. [送信] をクリックします。
   c. インスタンスの URL については、次の値をフォームに入力します。

フィールド	値
ID プロバイダー URL	IdP メタデータからインポートします。
ServiceNow のホームページ	https://<インスタンス>.service-now.com/navpage.do
エンティティ ID/発行者	https://<インスタンス>.service-now.com/navpage.do
対象者 URI	https://<インスタンス>.service-now.com/navpage.do

   d. [送信] をクリックします。

4. オプション: 複数の ID プロバイダーを使用する場合は、MultiSSO インストレーション イグジットを変更します。

   a. 移動先 システム定義 > インストレーションイグジット.
   インストレーション イグジットの現在のリストが表示されます。

   b. **MultiSSO** インストレーション イグジットを開きます。

   c. [スクリプト] フィールドで次のステートメントを探します。

```
var samlResponseTxt = request.getParameter("SAMLResponse");
if (!GlideSession.get().isLoggedIn() && GlideStringUtil.notNil(samlResponseTxt)) {
    var idpRecord = this.getIdPRecord(request);
    if (idpRecord) {
        SSO_Helper.debug("IdP found based on SAML response: " +
 idpRecord.getUniqueValue());
        return new SSO_Helper(idpRecord.getUniqueValue(), false, null, true);
    }
}
```

   d. このステートメントを次のコードに置き換えます。

```
var samlResponseTxt = request.getParameter("SAMLResponse");
if (!GlideSession.get().isLoggedIn() && GlideStringUtil.notNil(samlResponseTxt)) {
   /* // You have two profiles that use the same IdP entity id it cannot use
   // the IdP issuer / entity id from the response otherwise it may result in the
   // wrong IdP profile. IdP initiated login will not work
   var idpRecord = this.getIdPRecord(request);
   if (idpRecord) {
      SSO_Helper.debug("IdP found based on SAML response: " +
 idpRecord.getUniqueValue());
      return new SSO_Helper(idpRecord.getUniqueValue(), false, null, true);
      }*/
   return new SSO_Helper(null, true);
   }
```

   注: この設定では IdP により開始されるログインは機能しません。

   e. [更新] をクリックします。

5. オプション: 複数の会社を使用している場合は、ユーザーをマルチプロバイダー SSO 用に構成し、ユーザーに応じて ID プロバイダーレコードの sys_id を更新します。

   (Optional) 詳細については、「マルチプロバイダー SSO のユーザーの構成」を参照してください。

   - ユーザーが エッジ暗号化 プロキシサーバー経由でログインするように設定するには、エッジ暗号化 プロキシサーバー URL を使用している ID プロバイダー レコードの sys_id を使用します。
   - ユーザーがインスタンスにログインするように設定するには、インスタンス URL を使用している ID プロバイダー レコードの sys_id を使用します。

自動翻訳

ログイン **URL**

URL	ログイン先
https://<プロキシのホスト名>:<ポート>/login_with_sso.do?glide_sso_id=<プロキシサーバー URL の IdP レコードの sys_id>	プロキシサーバー経由でログインします。
https://<インスタンス名>.service-now.com/login_with_sso.do?glide_sso_id=<インスタンス URL の IdP レコードの sys_id>	インスタンス経由でログインします。

エッジ暗号化プロキシサーバーのプロパティ

<インストール ディレクトリー>/conf/ フォルダーにある edgeencryption.properties 設定ファイルには、環境設定に使用するプロパティが含まれています。

プロキシサーバーのプロパティを変更した後は、プロキシサーバーを再起動する必要があります。

クリアー テキストおよび静的 **IV** のプロパティ

edgeencryption.customer.assigned.known.cleartext	すべてのプロキシで同じキーを使用していることをインスタンスに検証させるためのクリアー テキスト。起動時に、プロキシはクリアー テキストを暗号化し、暗号化したテキストをインスタンスに送信します。インスタンス側はクリアー テキストを知らず、キーがインスタンスに送信されることもありません。このプロパティはすべてのプロキシで同じにする必要があります。
edgeencryption.encrypter.static.iv	等価性保存および順序保存の暗号化で使用される静的 IV (初期化ベクトル)。このプロパティはすべてのプロキシで同じである必要があり、正確に 16 バイト (16 文字の ASCII 文字) である必要があります。

デジタル署名のプロパティ

edgeencryption.proxy.signature.keystore.path	パスと Java KeyStore のファイル名。
edgeencryption.proxy.signature.keystore.password	パスワード。デフォルトのパスワードは <changeme> です。Java KeyStore のインストール後にパスワードを変更します。
edgeencryption.proxy.signature.keystore.keyalias	RSA キーペアが生成されるときに -alias 引数として与えられるキーエイリアス。

ファイルストアのプロパティ

edgeencryption.keyfile.directory	このディレクトリは、キー ファイルが格納される場所を指定します。Java KeyStore または SafeNet KeySecure キーストアを使用してい

	る場合は、このプロパティをコメント アウトしたままにしておきます。 例： `edgeencryption.keyfile.directory=keys` Unbound Technology のキーを使用している場合は、このプロパティをコメント解除し、値としてキーのディレクトリを設定します。

全般的な設定のプロパティ

edgeencryption.config.poll.interval	ポーリング間隔 (秒単位)。デフォルト設定では、プロキシが暗号化設定の変更を認識するまでに 5 秒かかります。値を大きくするほど、インスタンスがオフラインのプロキシを検出するのに時間がかかります。 ⚠ 警告: このプロパティは変更しないでください。プロキシ ポーリング間隔のデフォルト設定を変更すると、プロキシがオンラインになったときに検出が遅れることがあります。
edgeencryption.rules.dir	プロキシで暗号化ルールが格納されるフォルダ。
edgeencryption.encryption.order_preserving.cache.enable	順序保存の暗号化タイプをサポートするためにキャッシュを使用するかどうかを決定する設定。
edgeencryption.encryption.order_preserving.cache.size	最大キャッシュ サイズ (バイト単位)。
edgeencryption.jobs.concurrency	このプロキシで同時に実行できる一括暗号化ジョブの最大数。
edgeencryption.jobs.requests_per_second	このプロキシからインスタンスに送信できる 1 秒あたりの HTTP ジョブ要求の数。
edgeencryption.attachments.request.timeout.seconds	添付ファイルのアップロード要求のタイム アウト (秒単位)。
edgeencryption.request.buffer.size	暗号化要求のサイズ。暗号化要求がこのサイズより大きい場合、超過分はディスクに保存されます。 ⚠ 警告: このプロパティは変更しないでください。
edgeencryption.httpclient.request.buffer.size	クライアント要求のサイズ。クライアント要求がこのサイズより大きい場合、超過分はディスクに保存されます。 ⚠ 警告: このプロパティは変更しないでください。

edgeencryption.httpclient.header.size	要求/応答ヘッダーのサイズ。 • 最小値：8K • 最大値：32K ⚠ 警告: このプロパティは変更しないでください。
edgeencryption.proxy.idle.timeout	トランザクションがタイム アウトになるまでの時間 (秒単位)。 デフォルト値：300 (秒)
edgeencryption.proxy.keepalive.interval	プロキシからインスタンスに発行される ping の間隔 (秒単位)。プロキシとインスタンス間の接続を確認するために、定期的に ping が発行されます。 • デフォルト値：10 (秒) • 最小値：5 (秒)
edgeencryption.register.retry.count	登録を試みるためにプロキシからインスタンスに対して ping を実行する最大回数。 デフォルト値：0 (無制限)
edgeencryption.tokenization.exclusion.list	暗号化パターンは、これらのフィールドにある文字列をトークン化できません。

Java KeyStore のプロパティ

edgeencryption.keystore.path	Java KeyStore へのパス。ファイルストアまたは SafeNet KeySecure キーストアを使用している場合は、このプロパティをコメントアウトしたままにしておきます。 例： `edgeencryption.keystore.path = keystore/keystore.jceks`
edgeencryption.keystore.password	プロキシが Java KeyStore への接続に使用するパスワード。ファイルストアまたは SafeNet KeySecure キーストアを使用している場合は、このプロパティをコメントアウトしたままにしておきます。

ログ記録のプロパティ

ログ記録のプロパティは、<installation directory>/conf/ ディレクトリの lo4gj2.properties ファイルにあります。これらのプロパティは、トラブルシューティングのため、または ServiceNow サポートから指示された場合にのみ変更され ます。詳細については 、「エッジ暗号化プロキシのデバッグログを増やす方法」を参照してください。

NAE デバイスキーストアのプロパティ

edgeencryption.nae.retries	再試行回数。
edgeencryption.nae.enabled	NAE デバイスが使用可能かどうかを示す設定。
edgeencryption.nae.server	NAE サーバーの名前。
edgeencryption.nae.port	NAE サーバーによって使用されるポート。
edgeencryption.nae.protocol	NAE サーバーによって使用されるプロトコル。
edgeencryption.nae.keystore.path	NAE サーバー上のキーストアへのパス。
edgeencryption.nae.keystore.password	NAE キーストアのパスワード。
edgeencryption.nae.username	NAE デバイスでの認証に使用するユーザー名。
edgeencryption.nae.password	NAE デバイスでの認証に使用するパスワード。
edgeencryption.nae.client.certificate	NAE サーバー上のキーストアにある証明書。ユーザー名とパスワードの代わりに証明書を使用して認証を行う場合に、このプロパティを設定します。

パスワードのプロパティ

edgeencryption.encrypter.properties.password	conf フォルダーにあるファイルの名前。このファイルには、edgeencryption.properties ファイルにあるパスワードをセキュアなプロセスの中で難読化するのに使用される文字列が含まれています。 注: conf フォルダーにあるファイルの名前。このファイルには、edgeencryption.properties ファイルにあるパスワードをセキュアなプロセスの中で難読化するのに使用される文字列が含まれています。

プロキシのプロパティ

edgeencryption.proxy.host	プロキシを実行しているコンピューターのサーバー名、IP アドレス、または完全修飾ドメイン名。このプロパティとポートによって、クライアントがプロキシサーバーへのアクセスに使用する URL が定義されます。
edgeencryption.proxy.name	プロキシ名。プロキシごとに一意である必要があります。
edgeencryption.proxy.http.port	HTTP 通信のプロキシのポート。
edgeencryption.proxy.https.port	HTTPS 通信のプロキシのポート。

プロキシ設定のロックのプロパティ

edgeencryption.proxy.locked	True の場合、プロキシはインスタンスからの暗号化設定の変更または暗号化ルールの変更を受け入れません。すべての暗号化設定およびルールが最終的に決定した後、本番インスタンスでこのプロパティを設定します。

プロキシ データベースのプロパティ

edgeencryption.db.url	プロキシ データベースの場所。同じインスタンスに接続するすべての暗号化プロキシで同じである必要があります。
edgeencryption.db.user	プロキシ データベースにアクセスするためのユーザー名。同じインスタンスに接続するすべての暗号化プロキシで同じである必要があります。
edgeencryption.db.password	プロキシ データベースにアクセスするためのパスワード。同じインスタンスに接続するすべての暗号化プロキシで同じである必要があります。
edgeencryption.db.name	プロキシ データベース名。同じインスタンスに接続するすべての暗号化プロキシで同じである必要があります。このプロパティのデフォルトは edgeencryption です。
edgeencryption.db.bootstrap.file	プロキシ データベースのブートストラップ ファイル。このファイルは sql/ ディレクトリーからの相対パスです。同じインスタンスに接続するすべての暗号化プロキシで同じである必要があります。 ⚠ **警告:** 通常、このパラメーターは変更しないでください。

プロキシサーバーのパフォーマンスのプロパティ

プロキシサーバーのパフォーマンスのプロパティは、デフォルトでは設定ファイルに存在しません。デフォルト値を変更するには、プロパティを追加してプロキシサーバーを再起動する必要があります。詳細については、「エッジ暗号化の診断とパフォーマンス」を参照してください。

edgeencryption.stat.collection.enabled	エッジ暗号化 プロキシサーバーのパフォーマンス ダッシュボードで使用される統計情報の収集を有効にします。 デフォルト値：true エッジ暗号化 プロキシサーバーのパフォーマンス ダッシュボードで使用される統計情報の収集を無効にするには、このプロパティを追加して、値を False に設定します。

edgeencryption.stat.collection.interval	エッジ暗号化 プロキシサーバーが統計情報を収集する間隔 (秒単位)。この値は 30 秒未満にすることはできません。 デフォルト値：30 (秒)

SSL 証明書のプロパティ

SSL 証明書プロパティの値を変更した場合は、プロキシを再起動します。システムでは起動時に HTTPS キーペアを使用して、プロキシサーバーの接続を確立し、プロキシがクライアント要求に応答する方法を決定します。

edgeencryption.proxy.https.cert.alias	プロキシサーバーから接続中のクライアントに提供される証明書のエイリアス。
edgeencryption.proxy.https.keystore.path	HTTPS 証明書が格納されているキーストアへのパス。
edgeencryption.proxy.https.keystore.password	HTTPS 証明書が格納されているキーストアのパスワード。

ターゲット (インスタンス) のプロパティ

edgeencryption.target.host	インスタンスのホスト名。同じインスタンスに接続するすべての暗号化プロキシで同じである必要があります。このプロパティは、プロキシのインストール時に設定されます。たとえば、instancename.servicenow.com などです。
edgeencryption.target.port	インスタンスのポート。同じインスタンスに接続するすべての暗号化プロキシで同じである必要があります。このプロパティは、プロキシのインストール時に設定されます。
edgeencryption.target.protocol	インスタンスのプロトコル。同じインスタンスに接続するすべての暗号化プロキシで同じである必要があります。このプロパティは、プロキシのインストール時に設定されます。次のオプションが含まれます。 • http • https

Unbound Technology プロバイダーのプロパティ

edgeencryption.ekm.provider.classname	実装の内部クラス名。 ⚠ 警告: このプロパティは変更しないでください。
edgeencryption.thirdparty.vendor.library.path	Unbound クライアント マシン上の Unbound API JAR ファイルへのパス。

edgeencryption.ekm.provider.rsa.wrapping.key.alias	Unbound Technology の実装のラッピング キー エイリアス。すべてのプロキシで同じにする必要があります。

ユーザー アカウントのプロパティ

edgeencryption.target.username	プロキシがインスタンスへのログインに使用するユーザー名。ユーザーには edge_encryption ロールが必要です。「エッジ暗号化ユーザー アカウントを設定する」を参照してください。
edgeencryption.target.password	プロキシがインスタンスへのログインに使用するパスワード。

Web プロキシのプロパティ

edgeencryption.webproxy.host	Web プロキシの名前または IP アドレス。
edgeencryption.webproxy.port	Webプロキシのポート。
edgeencryption.webproxy.user	Web プロキシへの接続に使用されるユーザー名。Web プロキシで認証を使用しない場合は、このプロパティをコメント アウトしたままにしておきます。
edgeencryption.webproxy.password	Web プロキシへの接続に使用するパスワード。Web プロキシで認証を使用しない場合は、このプロパティをコメント アウトしたままにしておきます。

不使用になったプロキシ暗号化プロパティ

edgeencryption.encrypter.default.key128

現在の AES 128 キーの名前を指定します。AES 128 キーは、使用しなくても利用可能でなければなりません。すべてのプロキシで同じにする必要があります。

インスタンスでこれらのキーのメンテナンスを実行します。

edgeencryption.encrypter.default.key256

現在の AES 256 キーの名前を指定します。すべてのプロキシで同じにする必要があります。

インスタンスでこれらのキーのメンテナンスを実行します。

edgeencryption.encrypter.key

キー名をキーごとに指定し、デフォルトのキーを指定するために使用します。これは、暗号化された各アイテムに含まれるメタデータと統合されたキーエイリアスであり、したがってインスタンスに格納されます。キーの名前は小文字を使用する必要があります。

edgeencryption.encrypter.type

暗号化キーストアシステムのタイプを指定します。

edgeencryption.encrypter.file

キーに関連付けられたテキストファイルのパスとファイル名を指定します。

edgeencryption.encrypter.password

キーストアにアクセスするためのパスワードを指定します。

CyberArk と Edge プロキシサーバーとの統合

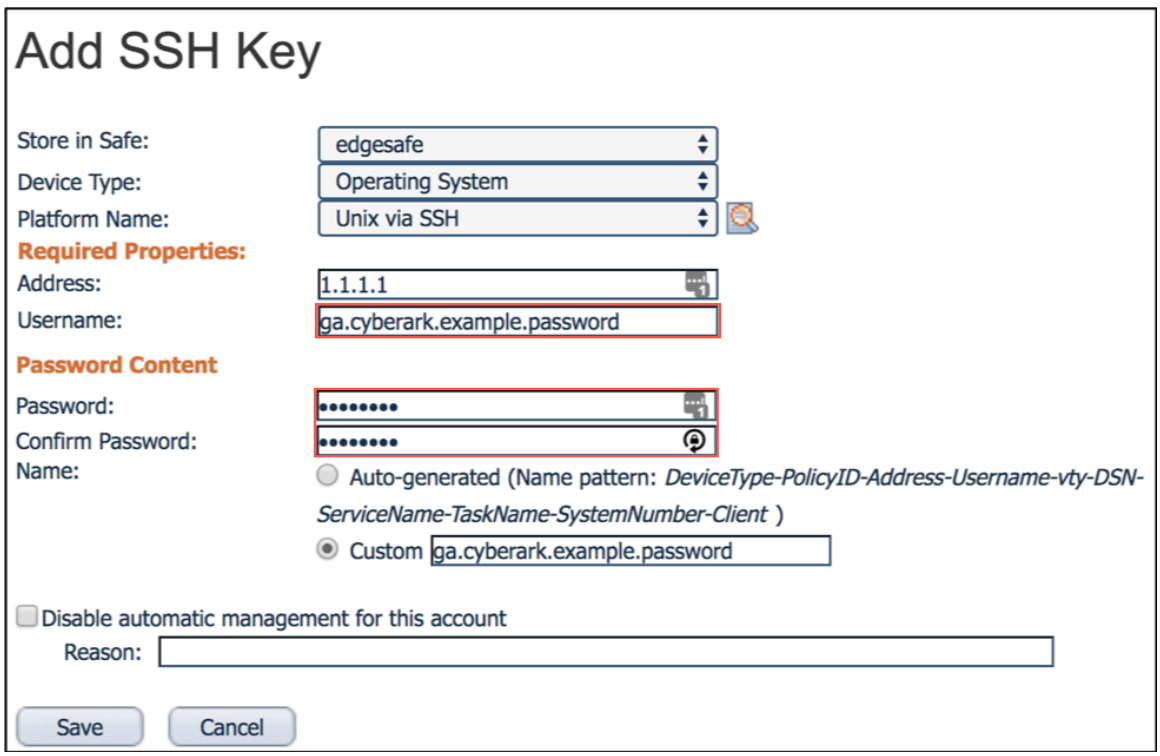
CyberArk を使用して、パスワードを集中管理された安全なデジタルボールトに保存し、以前にクリアテキストで保存され、ファイルアクセスによって保護されたパスワード、または以前に第 2 ファイルで暗号化されたパスワードを保護します。

CyberArk AIM (アプリケーション ID 管理) は、ハードコーディングされたパスワードや可視のパスワードをなくすことで、不正なアクセスを防止します。AIM は、独立した強化サーバー上のデジタル ボールトにパスワードを格納します。ボールトではパスワードはデジタル認証情報として表現されます。AIM クライアント (Edge プロキシサーバー) は、CyberArk のデジタル認証情報を使用して、独立したサーバーにアクセスし、セキュリティ保護されたパスワードを取得します。パスワードは Edge プロキシサーバーにもインスタンスにも格納されません。

CyberArk のデジタル ボールトの認証情報

CyberArk と Edge プロキシサーバーとの統合をセットアップするには、事前に CyberArk を購入して設定を行う必要があります。

CyberArk に認証情報を追加するには、認証情報の **[Platform Name]** を **[Unix via SSH]** に設定して、**[Custom]** の認証情報 **[Name]** を作成するか、**[Auto-generated]** の認証情報 **[Name]** を書き留めておきます。この認証情報を使用するように Edge プロキシを設定すると、プロキシサーバーは、この認証情報 **[Name]** をプロキシ内の設定に合わせます。



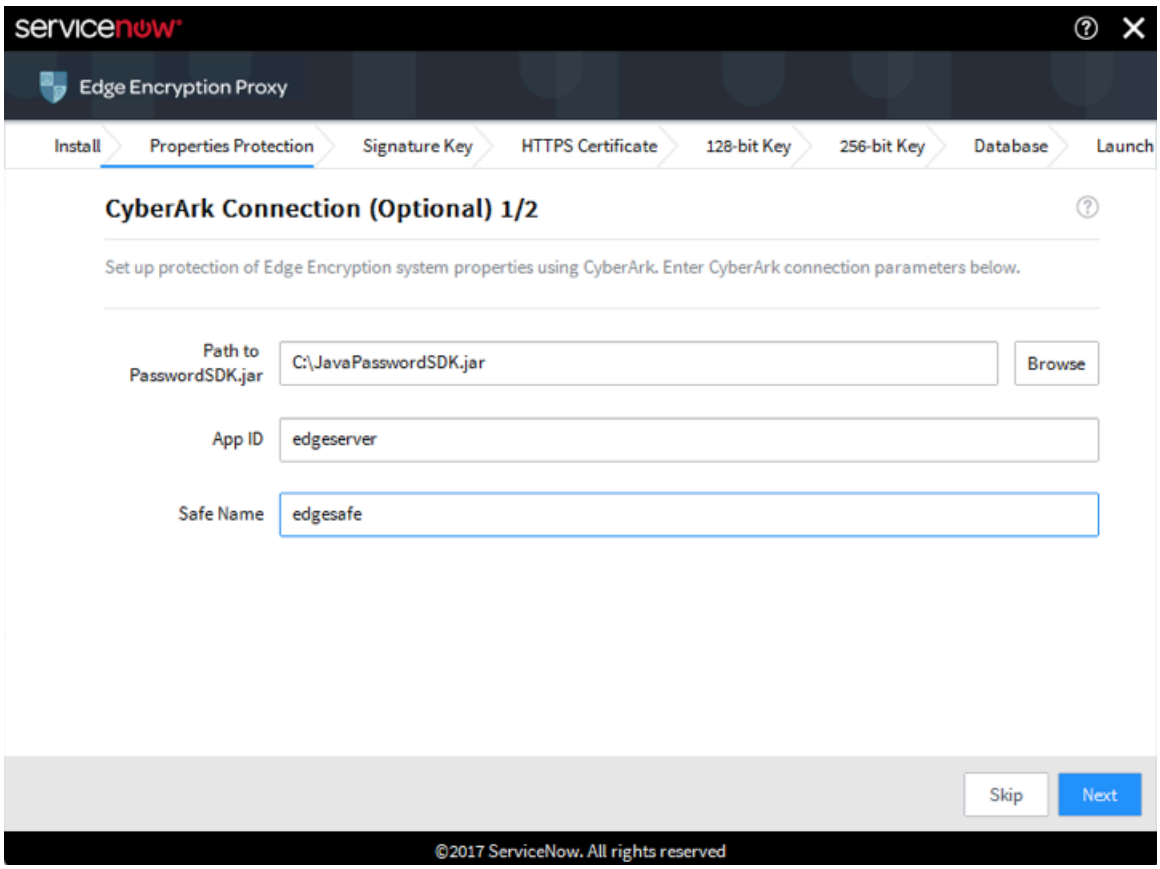
認証情報の各エントリには、セキュリティ保護されている **[Password]** のほか、アプリケーションがそのパスワードへのアクセスに使用する認証情報 **[Name]** が保持されています。

注: CyberArk の認証情報は暗号化キーではありません。

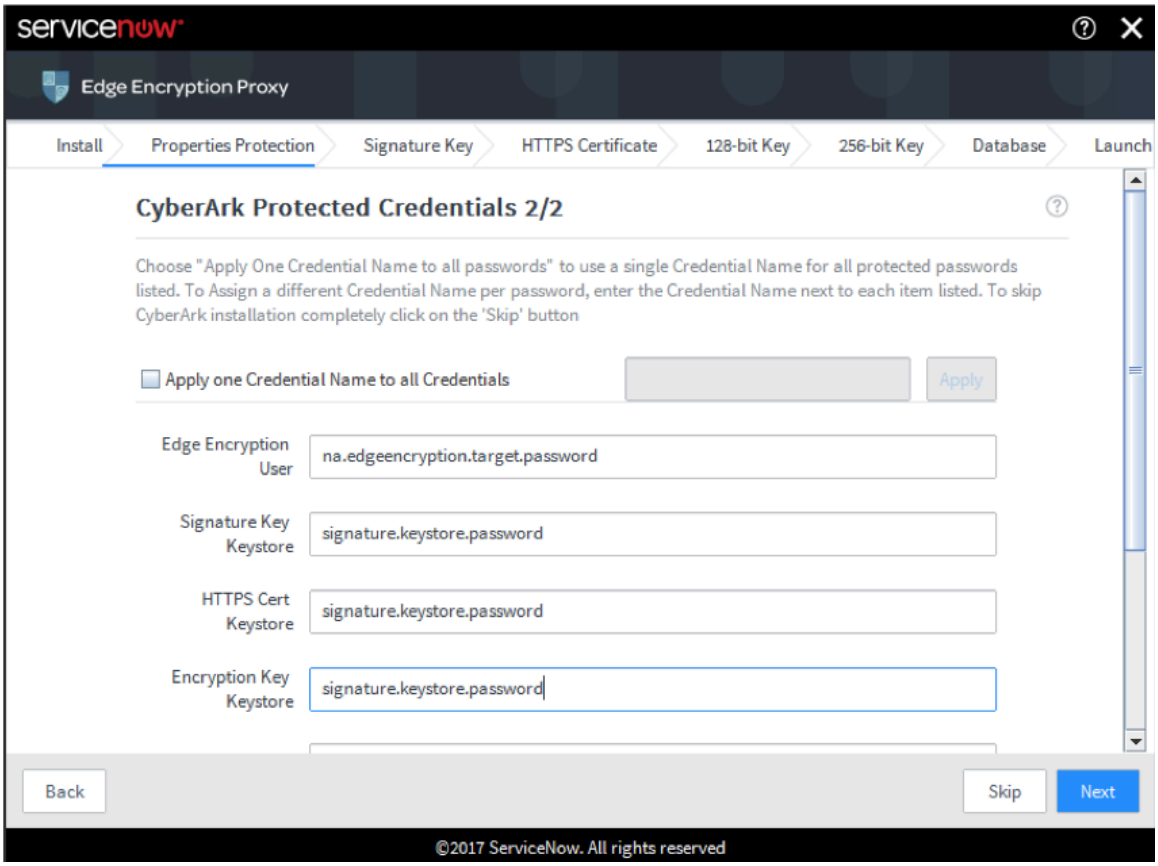
Edge プロキシのインストール時に CyberArk を追加する

プロキシ インストーラーには、CyberArk との統合を行うための新しい設定ページが含まれています。プロキシ インストーラーによるプロキシのインストール時に CyberArk の追加を希望しない場

合は、このページを省略できます。または、設定ファイルで CyberArk の統合を手動でセットアップして設定を行うこともできます。



プロキシ インストーラーには、CyberArk で保護される認証情報用の新しいページも含まれています。このページでは、1 つまたは複数の認証情報名を使用して、さまざまなプロパティの構成を行うことができます。プロキシ インストーラーによるプロキシのインストール時に CyberArk の追加を希望しない場合は、このページを省略できます。



CyberArk のパスワード保護

Edge プロキシ インストーラーにあるパスワード フィールドのうち、CyberArk のボールトで CyberArk の認証情報が設定されていて、インストーラーの [CyberArk で保護される認証情報] ページで指定されているものは、グレー表示され、「CyberArk による保護」というメッセージが表示されます。

Edge プロキシサーバーでロードバランサーを使用する

ロードバランサーを使用して、エッジ暗号化プロキシ設定のプロキシサーバー間の負荷を分散することができます。 ロードバランサーとプロキシサーバーが異なるポートを使用している場合は、ユーザーがブラウザで応答を表示できるように、ロードバランサーのホスト名と HTTPS ポートを指定します。

重要: すべての本番環境で、冗長性のためにエッジ暗号化プロキシサーバーを少なくとも 2 つ用意してください。

ロードバランサーを使用しない Edge 要求の処理

ロードバランサーを使用していない場合、要求は次のように処理されます。

1. ユーザーがブラウザーから要求を出します。
2. ブラウザーが Edge プロキシサーバーに要求を送信します。
3. プロキシサーバーが ServiceNow インスタンスに要求を送信します。
4. ServiceNow インスタンスがプロキシサーバーに応答を返します。
5. プロキシサーバーは、ユーザーのブラウザーに応答を返す前に、応答ヘッダーに自分自身のポート番号を追加します。

ユーザーは、応答ヘッダーで指定されたポート番号でプロキシサーバーからの応答を表示できます。これで、要求は正常に完了します。

ロードバランサーを使用した Edge 要求の処理

これに対して、ロードバランサーを使用している場合、ユーザーのブラウザーはプロキシサーバーと通信するのではなくロードバランサーと直接通信します。要求は次のように処理されます。

注: この例では、プロキシサーバーのポート番号として 1025 を使用しています。

1. ユーザーがブラウザーから要求を出します。
2. ブラウザーが要求をロードバランサー仮想 IP (VIP) (仮想サーバーとも呼ばれます) に送信します。
3. VIP は、プロキシサーバーを指し示すように設定されています (10.2.200.148:1025 など)。そのため、ロードバランサーは要求をプロキシサーバーに転送します。
4. プロキシサーバーが ServiceNow インスタンスに要求を送信します。
5. ServiceNow インスタンスがプロキシサーバーに応答を返します。
6. プロキシサーバーは、応答内のロケーション ヘッダーを、risk-servicenow.dev.echonet:1025 のプロパティで設定された値に書き換えます。
   - ホスト：edgencryption.proxy.host
   - **HTTP** ポート：edgeencryption.proxy.http.port
   - **HTTPS** ポート：edgeencryption.proxy.https.port
7. プロキシサーバーは、プロキシサーバー ポートを指し示しているロケーション ヘッダーを使用して、応答をロードバランサーに転送します。

この結果は、ロードバランサーとプロキシサーバーが同じポートを使用しているかどうかによって異なります。

- ロードバランサーとプロキシサーバーが同じポートを使用している場合、ユーザーは応答ヘッダーで識別される同じポートから応答を受け取るため、要求は成功します。
- ロードバランサーとプロキシサーバーが異なるポートを使用している場合、ユーザーのブラウザーはロードバランサーとのみ通信しますが、応答はプロキシサーバー上にあるため、要求は失敗します。

解決策

この問題は、単にロードバランサーとすべての Edge プロキシサーバーを同じポート上で使用して解決できます。ただし、これは理想的な解決策ではありません。より良い解決策として、ロードバランサーがどのポートを使用しているかをシステムが認識できるようにします。

プロキシサーバーとロードバランサーが異なるポートを使用している場合、次のプロパティを使用すると、Edge プロキシサーバーが応答メッセージをロードバランサーに再ルーティングできます。

- *edgeencryption.proxy.rewrite.location.host* は、ロードバランサー経由で ServiceNow にアクセスするために使用するホスト名を指定します。
- *edgeencryption.proxy.rewrite.location.https.port* は、ロードバランサー経由で ServiceNow にアクセスするために使用する HTTPS ポートを指定します。

ロードバランサーの設定を行う

ロードバランサーとプロキシサーバーが異なるポートを使用している場合は、ユーザーがブラウザで応答を表示できるように、ロードバランサーのホスト名と HTTPS ポートを指定します。

始める前に
必要なロール：

- Windows ホストでのローカルアドミニストレーターまたはドメインアドミニストレーター
- Linux ホスト上のファイルシステムへのフルアクセス権限があるサービスユーザー

手順

1. アドミン、ドメインアドミン、またはサービス ユーザーとして、プロキシサーバー ホストにログインします。
2. Edge プロキシのインストール ディレクトリーに移動し、conf/edgeencryption.properties を選択します。
3. 次のプロパティを設定します。

プロパティ	説明
edgeencryption.proxy.rewrite.location.host	Edge の構成にプロキシサーバー間の負荷分散を行うロードバランサーが含まれている場合は、応答をロードバランサーに書き換えて要求を完了させることができます。 ○ プロキシのセットアップでロードバランサーが存在する場合は、ロードバランサー経由で ServiceNow にアクセスするために使用するホスト名を指定します。 ○ オプション：プロキシのセットアップでロードバランサーが存在しない場合は、プロキシサーバーが使用するホスト名に設定することができます。
edgeencryption.proxy.rewrite.location.https.port	Edge の構成にプロキシサーバー間の負荷分散を行うロードバランサーが含まれている場合は、ロードバランサー経由で ServiceNow にアクセスするために使用する HTTPS ポートを指定します。 ○ 構成にロードバランサーが存在する場合は、ロードバランサー経由で ServiceNow にアクセスするために使用する HTTPS ポートを指定します。 ○ オプション：構成にロードバランサーが存在しない場合は、プロキシサーバーが使用する HTTPS ポートに設定することができます。

4. ファイルを保存します。

結果
これで、ユーザーはブラウザーでレスポンスを表示できるようになり、要求を完了できます

エッジ暗号化のアップグレード

インスタンスのアップグレードとプロキシサーバーのアップグレードのどちらを行う場合も、エッジ暗号化 環境で特別な考慮が必要になります。

インスタンスのアップグレード

エッジ暗号化環境でインスタンスのアップグレードを行うには、インスタンスのアップグレード後に Edge によるコントロールが正しく機能するように注意する必要があります。

インスタンスのアップグレード中に、次のものを追加、編集、または削除しないでください。

- エッジ暗号化 の設定
- Edge Encryption (エッジ暗号化) のルール
- Edge Encryption (エッジ暗号化) のトークン化パターン
- Edge Encryption (エッジ暗号化) のスケジュール済みジョブ
- エッジ暗号化のキー設定
- エッジ暗号化のスケジュール設定済みのアップグレード
- エッジ暗号化の拒否リスト IP 設定

インスタンスのアップグレード中に実行されたスケジュール設定済みのジョブは、どれも完了しません。中断されたジョブを完了するには、インスタンスのアップグレードの完了後にジョブを再実行します。ジョブのスケジュールを変更した場合、インスタンスのアップグレード前に発生した処理は失われず、まだ処理が完了していないデータのみの処理がジョブで続行されます。

プロキシサーバーのアップグレード

インスタンスが エッジ暗号化 プロキシサーバーをアップグレードできるようにプロキシをスケジュールするか、任意のタイミングでプロキシサーバーを手動でアップグレードします。

⚠ **警告:** Windows でのアップグレードでは、ファイルロックの問題が発生し、アップグレードが失敗する可能性があります。アップグレードを正常に完了するには、インストールディレクトリの下のファイルを開いた状態にしないでください。また、インストールディレクトリに既存のシェルが存在しないようにする必要があります。特に、インストールディレクトリでコマンドラインから (`bin\edgeencryption.bat install/start` 経由で) プロキシを起動する場合は、そのシェルを閉じるか、後でインストールディレクトリから移動させる必要があります。インストールディレクトリにあるファイルを、エディターや他のアプリケーションで開かないでください。

サードパーティのライブラリー

Gemalto などのサードパーティライブラリーは、同じディレクトリに格納されている場合、インスタンスとプロキおよびサーバーのアップグレード時に失われます。アップグレード中にサードパーティのライブラリーが失われないようにするには、以下を実行します。

1. 次のプロパティを edgeencryption.properties に手動で追加します。

   ```
   edgeencryption.ekm.provider.classname =
   com.snc.edgeencryption.encryption.CloudEdgeNaeKeyProvider
   ```

2. *`edgeencryption.thirdparty.vendor.library.path`* ベンダーライブラリーの場所プロパティを追加し、「/path/to/jars」に設定します。

   たとえば、

   ```
   edgeencryption.thirdparty.vendor.library.path = /app/servicenow/libs
   ```

3. SafeNet JAR をそのパスにコピーします。

自動翻訳

エッジ暗号化 インストール外でサードパーティライブラリーをインストールした後、アップグレード中にそのライブラリーが失われることはありません。

スケジュール設定済みのアップグレード

重要: ServiceNow インスタンスのアップグレード中に、プロキシサーバーのバージョンもアップグレードしてインスタンスのバージョンに合わせ、互換性の問題が発生する可能性を減らします。

予定された時刻にインスタンスがプロキシサーバーをアップグレードできるように、アップグレードをスケジュールします。この機能は、アップグレード後にデフォルトで使用できます。スケジュール設定済みのアップグレードでは、次のイベントが発生します。

1. プロキシサーバーがインスタンスに問い合わせて、アップグレード可能な新しいバージョンがあるかどうかを確認します。通常、新しいバージョンは、インスタンスがアップグレードされたときに使用可能になります。
2. 新しいバージョンのプロキシサーバーが使用可能になると、アドミニストレーターはログイン時に通知を受け取ります。
3. アドミニストレーターはプロキシサーバーごとに、エッジ暗号化プロキシサーバーのアップグレードをスケジュールすることができます。

   注: security_admin ロールを持つユーザーのみが、プロキシサーバー経由でアップグレード スケジュールを作成できます。

4. アップグレードがスケジュールされると、プロキシサーバーは予定された時刻に自動的にアップグレードされます。アップグレード中、プロキシサーバーは一時的にオフラインになります。

   注: プロキシサーバーはアップグレード中に再起動するため、一時的にオフラインになります。その時間の長さは環境、およびプロキシ サービスの停止と再起動にかかる時間によって異なります。

5. スケジュール設定済みのアップグレード中に、新しいプロキシ ディレクトリが作成され、設定ファイルが新しいディレクトリにコピーされます。新しいプロパティが既存のプロパティ ファイルに書き込まれます。古いプロキシ ディレクトリにある次のファイルまたはディレクトリが、新しいプロキシ ディレクトリにコピーされます。
   - /conf ディレクトリー
   - /keys ディレクトリー
   - /keystore ディレクトリー
   - java/jre/lib/security/cacerts ファイル

   その結果、キー、キーストア、設定、および証明書が保持されます。

   注: 新しいプロキシ ディレクトリーにコピーされるのは上記のファイルのみです。プロキシサーバー ディレクトリにあるその他のカスタマイズされたファイルは、スケジュール設定済みのアップグレード中は保存されません。アップグレードのログ ファイルは、元のプロキシ ディレクトリーの <original-proxy-directory>/tmp/upgrade-wrapper/bin フォルダーにあります。

スケジュール設定済みのアップグレードの必須条件

エッジ暗号化プロキシのアップグレードをスケジュールする際は、事前に次の点を確認してください。

1. *JAVA_HOME* 環境変数が、エッジ暗号化プロキシのディレクトリー構造以外の場所にあるマシンにインストールされた Java を指し示していること。
2. *JAVA_HOME* 環境変数が、バージョン 1.8_u144 以降としてインストールされた Java を指し示していること。
3. エッジ暗号化プロキシの wrapper.conf ファイルの `-Djava.io.tmpdir` パラメーターが、エッジ暗号化プロキシのディレクトリー構造「以外の場所」にあるディレクトリーを指し示していて、プロキシがそのディレクトリーに対して読み取り/書き込み/実行の権限を持っていること。オプションで、Java がデフォルトの tmp 場所を使用するように、パラメーター全体をコメント アウトすることもできます。

手動アップグレード

アップグレード スケジュールを作成する代わりに、コマンド ラインから個々のプロキシサーバーを手動でアップグレードすることができます。「Linux で実行中のエッジ暗号化プロキシサーバーを手動でアップグレードする」または「Windows で実行中のエッジ暗号化プロキシサーバーを手動でアップグレードする」を参照してください。

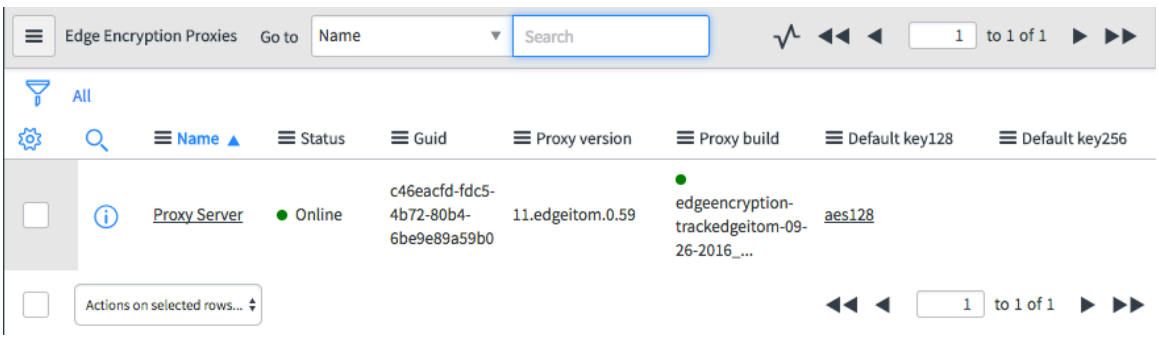
プロキシのビルド ステータス

に移動すると、プロキシサーバーが古くなっているかどうかを簡単に特定できます エッジ暗号化の設定 > プロキシ > すべて. プロキシのビルド ステータスは [プロキシのビルド] 列に次の色で表示されます。

緑
: プロキシサーバーは最新です。

黄
: プロキシサーバーは古くなっていて、アップグレードが必要です。

オレンジ
: アップグレードに失敗しました。ダウンタイムが発生しないように、プロキシサーバーは古いバージョンに戻されています。



スケジュール設定済みのプロキシ アップグレードが失敗した場合のトラブルシューティング

スケジュール設定済みのプロキシ アップグレードが失敗した場合、プロキシサーバーはアップグレード前のバージョンに戻ります。元のデータ、キー、および設定ファイルはすべて保持されます。このプロセスには数分かかることがあります。カスタマーサービス & サポートに連絡して、アップグレードを確実に成功させてください。

失敗の原因を調べるには、アップグレード スケジュールで [障害の原因] をチェックします。また、失敗したアップグレードのインストール ディレクトリーは保持されているため、ログ ファイルをトラブルシューティングに使用できます。

注: 余分なプロキシ ディレクトリを削除する前に、ログ ファイルを確認してディレクトリーが最新であることを必ず確認してください。ログ ファイルに最近のアクティビティが記録されている場合、プロキシからインスタンスに接続している可能性があります。

スケジュール設定済みのプロキシ アップグレードが繰り返し失敗する場合、プロキシサーバーを手動でアップグレードすることができます。「Linux で実行中のエッジ暗号化プロキシサーバーを手動でアップグレードする」および「Windows で実行中のエッジ暗号化プロキシサーバーを手動でアップグレードする」を参照してください。

Java の最低要件

エッジ暗号化 プロキシサーバーをインストールまたは実行するホストマシンでは、サポートされているバージョンの Java を保持する必要があります。現在サポートされているバージョンは、17.x バージョンシリーズの Java 17.0.3 以降です。

注: Java 11 は Yokohama リリースをもってサポートされなくなりました。Yokohama 以降のバージョンの エッジ暗号化 プロキシをインストールする前に、エッジ暗号化 プロキシを使用して環境を Java 17 にアップグレードします。

Java 8 update 141 (8u141) 以下で AES 256 ビット暗号化を使用する場合、Java Cryptography Extension (JCE) 管轄ポリシーファイルを、各 エッジ暗号化 プロキシサーバー ホストのシステムの Java ホーム ディレクトリにコピーしてインストールする必要があります。スケジュール設定済みアップグレードまたは手動アップグレードを実行する前に、これらのファイルを <Java-home-directory>/jre/lib/security フォルダに追加します。AES 256 ビット暗号化ポリシーファイルをインストールするには、「AES 256 ビット暗号化キーを設定する」を参照してください。

プロキシのバージョンが混在する環境

古いバージョンのプロキシサーバーを実行しながら最新バージョンのプロキシサーバーを使用する環境は推奨されませんが、すべてのプロキシサーバーがインスタンスと同じバージョン ファミリーである環境はサポートされています。たとえば、Zurich リリースのインスタンスを使用している環境では、Zurich のパッチまたはホットフィックスを適用したプロキシサーバーがサポートされます。ただし、次の制限があります。

- あるプロキシサーバーがサポートしている機能を別のプロキシがサポートしていない場合、使用しているプロキシサーバーによっては一貫性のない動作が生じます。
- プロキシサーバーが古くなっている場合、最近のセキュリティ強化が組み込まれていない可能性があります。

以前のリリースのプロキシサーバーがインスタンスの新しいリリースに登録されている場合は、プロキシサーバーが古くなっているという通知が定期的に表示されます。最適で安全な環境を確保するため、ServiceNow では、常にインスタンスでサポートされている最新バージョンのソフトウェアにプロキシサーバーをアップグレードすることをお勧めしています。

エッジ暗号化プロキシサーバーのアップグレードをスケジュールする

インスタンスが古いプロキシサーバーをアップグレードできるように、アップグレード スケジュールを作成します。

始める前に

アップグレードをスケジュールするには、プロキシサーバー経由でインスタンスにログインする必要があります。
Java 8 update 141 (8u141) 以下で AES 256 ビット暗号化を使用する場合、Java Cryptography Extension (JCE) 管轄ポリシーファイルを、各 エッジ暗号化 プロキシサーバー ホストのシステムの Java ホーム ディレクトリにコピーしてインストールする必要があります。スケジュール設定済みアップグレードまたは手動アップグレードを実行する前に、これらのファイルを <Java-home-

directory>/jre/lib/security フォルダに追加します。AES 256 ビット暗号化ポリシーファイルをインストールするには、「AES 256 ビット暗号化キーを設定する」を参照してください。

必要なロール：security_admin

このタスクについて

アップグレードがスケジュールされると、プロキシサーバーは予定された時刻に自動的にアップグレードされます。アップグレード中、プロキシサーバーは一時的にオフラインになります。

> **注:** プロキシサーバーはアップグレード中に再起動するため、一時的にオフラインになります。その時間の長さは環境、およびプロキシ サービスの停止と再起動にかかる時間によって異なります。

スケジュール設定済みのアップグレード中に、新しいプロキシ ディレクトリが作成され、設定ファイルが新しいディレクトリにコピーされます。新しいプロパティが既存のプロパティ ファイルに書き込まれます。古いプロキシ ディレクトリにある次のファイルまたはディレクトリが、新しいプロキシ ディレクトリにコピーされます。

- /conf ディレクトリー
- /keys ディレクトリー
- /keystore ディレクトリー
- java/jre/lib/security/cacerts ファイル

その結果、キー、キーストア、設定、および証明書が保持されます。

> **注:** 新しいプロキシ ディレクトリーにコピーされるのは上記のファイルのみです。プロキシ サーバー ディレクトリにあるその他のカスタマイズされたファイルは、スケジュール設定済みのアップグレード中は保持されません。アップグレードのログ ファイルは、元のプロキシ ディレクトリーの <original-proxy-directory>/tmp/upgrade-wrapper/bin フォルダーにあります。

複数のプロキシサーバーが古くなっている場合は、プロキシサーバーごとに個別にアップグレードをスケジュールする必要があります。

> **注:** 同じマシンで複数のプロキシサーバーをホストすることは避けてください。ただし、この構成が環境に含まれている場合は、同じマシンで同時に複数のプロキシへのアップグレードをスケジュールしないでください。

手順

1. 移動先 すべて > エッジ暗号化の設定 > プロキシ > スケジュールをアップグレード.
2. **[New]** をクリックします。
3. フォームに入力します。

エッジ暗号化プロキシのアップグレード スケジュール フォーム

フィールド	説明
プロキシサーバー	アップグレード対象のプロキシサーバー。
ターゲットバージョン	このバージョンにプロキシサーバーをアップグレードします。この値は読み取り専用で、インスタンスで使用可能な最新のプロキシ バージョンに設定されます。
予定開始時間	アップグレードを開始する日時。

フィールド	説明
有効	スケジュール設定済みのアップグレードが有効かどうか。このフィールドが選択されていない場合、スケジュールされた日時にアップグレードが実行されません。
ステータス	アップグレードのステータス。この値は読み取り専用です。ステータスの候補は次のとおりです。 ○ 処理待ち ○ 実行中 ○ 完了 ○ 失敗

4. [送信] をクリックします。

次のタスク

アップグレードの一般的な時間は 15 分未満です。アップグレードの実行後、アップグレードの詳細を表示して詳しく調べることができます。アップグレードが失敗した場合は、[障害の原因] を確認して、次のステップを決定してください。

アップグレードの詳細

フィールド	説明
前のバージョン	サーバーがアップグレードされる前のバージョン。
後のバージョン	サーバーがアップグレードされた後のバージョン。
実際の開始時間	アップグレードが開始された時間。
終了時間	アップグレードが終了した時間。
障害の原因	アップグレードが失敗した理由。

Linux で実行中のエッジ暗号化プロキシサーバーを手動でアップグレードする

Linux で実行されているプロキシを更新します。

始める前に

Java 8 update 141 (8u141) 以下で AES 256 ビット暗号化を使用する場合、Java Cryptography Extension (JCE) 管轄ポリシーファイルを、各 エッジ暗号化 プロキシサーバー ホストのシステムの Java ホーム ディレクトリにコピーしてインストールする必要があります。スケジュール設定済みアップグレードまたは手動アップグレードを実行する前に、これらのファイルを <Java-home-directory>/jre/lib/security フォルダに追加します。AES 256 ビット暗号化ポリシーファイルをインストールするには、「AES 256 ビット暗号化キーを設定する」を参照してください。

必要なロール：ホストマシンでの security_admin またはローカルアドミニストレーター

手順

1. エッジ暗号化 プロキシのアップデート アーカイブ ファイルをインストール ディレクトリーにコピーします。
   a. 移動先 エッジ暗号化の設定 > インストールとダウンロード > ダウンロード
   b. [コマンドラインインストーラをダウンロード] リンクを選択します。
   c. ダウンロードが開始されたら、ダウンロード場所としてインストールディレクトリを選択します。
2. インストール ディレクトリーに移動します。
3. 次のコマンドを実行します。

   `java -jar edgeencryption-dist-<バージョン>-linux-x86-64.jar -m dist-upgrade -c <プロキシのディレクトリー>`

オプション	説明
プロキシのディレクトリー	プロキシが最初にインストールされたインストール ディレクトリー内のディレクトリー。このディレクトリーはインストールによって作成されます。

   ヘルプ画面が必要な場合は、このコマンドを引数なしで実行します：`java -jar edgeencryption-dist-<version>-linux-x86-64.jar`。

   現在のタイムスタンプで新しいプロキシディレクトリが作成されます。古いプロキシディレクトリのバックアップは、新しいプロキシのインストールディレクトリに backup.dist-upgrade_timestamp として保持されます。古いプロキシがシャット ダウンし、新しいプロキシが起動します。古いプロキシサーバーで開いている接続/トランザクションはすべて終了します。
4. 新しいディレクトリーにあるプロキシ ログとインスタンスを調べて、新しいプロキシが実行されていることを確認します。

Windows で実行中のエッジ暗号化プロキシサーバーを手動でアップグレードする

Windows で実行されているプロキシを更新します。

始める前に
Java 8 update 141 (8u141) 以下で AES 256 ビット暗号化を使用する場合、Java Cryptography Extension (JCE) 管轄ポリシーファイルを、各 エッジ暗号化 プロキシサーバー ホストのシステムの Java ホーム ディレクトリにコピーしてインストールする必要があります。スケジュール設定済みアップグレードまたは手動アップグレードを実行する前に、これらのファイルを <Java-home-directory>/jre/lib/security フォルダに追加します。AES 256 ビット暗号化ポリシーファイルをインストールするには、「AES 256 ビット暗号化キーを設定する」を参照してください。

必要なロール：ホストマシンでの security_admin またはローカルアドミニストレーター

手順

1. エッジ暗号化 プロキシのアップデート アーカイブ ファイルをインストール ディレクトリーにコピーします。
   a. 移動先 エッジ暗号化の設定 > インストールとダウンロード > ダウンロード
   b. [コマンドラインインストーラをダウンロード] リンクを選択します。
   c. ダウンロードが開始されたら、ダウンロード場所としてインストールディレクトリを選択します。
2. Windows のコマンド プロンプト プログラムをアドミニストレーター権限で起動します。

3. インストール ディレクトリーに移動します。
4. 次のコマンドを実行します。

```
java -jar edgeencryption-dist-<バージョン>-all.jar -m dist-upgrade -c <プロキシディレクトリ>
```

オプション	説明
プロキシのディレクトリー	プロキシが最初にインストールされたインストール ディレクトリー内のディレクトリー。このディレクトリーはインストールによって作成されます。

ヘルプ画面が必要な場合は、コマンド `java -jar edgeencryption-dist-<version>-linux-x86-64.jar` を引数なしで実行します。

現在のタイムスタンプで新しいプロキシディレクトリが作成されます。古いプロキシディレクトリのバックアップは、新しいプロキシのインストールディレクトリに backup.dist-upgrade_timestamp として保持されます。古いプロキシがシャット ダウンし、新しいプロキシが起動します。古いプロキシサーバーで開いている接続/トランザクションはすべて終了します。

5. 新しいディレクトリーにあるプロキシ ログとインスタンスを調べて、プロキシが更新されて実行されていることを確認します。

エッジ暗号化プロキシサーバーのアップグレードをロール バックする

プロキシのアップグレードが失敗した場合、以前のバージョンに戻すことができます。

始める前に
必要なロール：admin

このタスクについて
Zurich リリースのスケジュール設定済みのアップグレード機能を使用していて、アップグレードが失敗した場合、プロキシサーバーは自動的に古いバージョンにロール バックします。古いプロキシサーバーは無修正でバックアップ ディレクトリーに格納されます。

手動のアップグレードをロール バックする場合は、次の手順に従います。

手順

1. プロキシをシャット ダウンします。
2. 新しいプロキシ ディレクトリーを削除します。
3. バックアップ ディレクトリーの名前をプロキシ名に変更します。
   バックアップ ディレクトリーは、<プロキシ名>_backup という名前のプロキシ インストール ディレクトリーです。
4. プロキシを起動します。
5. プロキシ ログとインスタンスを調べて、プロキシがオンラインであることを確認します。

エッジ暗号化の設定

エッジ暗号化 プロキシサーバーをインストールして実行したら、プロキシサーバーを介して エッジ暗号化 を管理します。

インスタンスで暗号化設定と暗号化パターンを作成する前に、「エッジ暗号化のインストール」のすべての手順を完了する必要があります。

注: エッジ暗号化 の設定にアクセスするには、プロキシサーバーを介してログインし、security_admin ロールに昇格する必要があります。

暗号化キーをローテートする

インスタンスから暗号化キーのローテーションを実行します。新しいキーを追加し、デフォルトキーの割り当てを変更してから、一括キーローテーションまたは単一キーローテーションをスケジュールします。

暗号化キーをデフォルトキーとして設定する前に、各プロキシでキーを使用できるようにします。これにより、キーがデフォルトキーとして割り当てられるときに、プロキシでデータを暗号化するためのキーが確保されます。すべてのプロキシは、キーがデフォルトキーとして割り当てられる前に、そのキーにアクセスできる必要があります。

警告: プロキシからキーを削除する場合は、事前に一括キーローテーションジョブをセットアップして実行し、インスタンスでキーを使用しているデータが存在しないようにしてください。そのキーを使用して暗号化されている情報がまだある場合は、キーを削除した後で情報を復号化することはできません。

Edge のフィルタリングとソートの動作

デフォルトキーを変更した場合は、必ずキーローテーション (一括または単一のいずれかのローテーション) を実行してください。そうでない場合、レコードをソートおよびフィルタリングしたときに予期しない結果が生じることがあります。たとえば次のシナリオを考えます。

1. 1 つの暗号化キーを使用して暗号化レコードを作成します。
2. 新しいキーを作成し、これをデフォルトとして設定します。
3. 新しい暗号化キーを使用して新しい暗号化レコードセットを作成します。

Edge プロキシ経由で接続されているときに任意の暗号化フィールドでフィルタリングする場合、すべてのレコードを適切にフィルタリングで除外できなかったり、レコードが予期せず表示されたりする可能性があります。フィルターは、現在のデフォルトキーを使用して暗号化されたレコードに対してのみ機能します。前のデフォルトキーを使用して暗号化されたレコードは、引き続きリストビューに表示されます。

Edge プロキシ経由で接続されているときに任意の暗号化フィールドでソートする場合、人間が判読可能な同一のテキストが含まれる 2 つのレコードグループが暗号化フィールドに表示されます。

単一のキー ローテーション ジョブをスケジュールする

指定したキー エイリアスを使用して暗号化されたデータを検索し、現在のデフォルトの暗号化キーでデータを再暗号化するジョブをスケジュールします。データは復号化されてから、デフォルトキーで再暗号化されます。

始める前に

必要なロール：security_admin
このジョブをスケジュールする前に、次でデフォルトキーを更新してください: エッジ暗号化の設定 > 暗号化キーの設定 > デフォルトキーを設定.

手順

1. 移動先 エッジ暗号化の設定 > メンテナンス > 単一のキーローテーションをスケジュール.
2. 必要に応じて、フォームのフィールドに入力します。

フィールド	値
名前	内容を端的に表す名前を入力します。
ジョブタイプ	[単一のキー ローテーション] を選択します。
キー	廃止するキーを入力します。このキーが次のデフォルトキーでなくなったことを確認します エッジ暗号化の設定 > 暗号化キーの設定 > デフォルトキーを設定.
推定レコード数	処理するレコードの推定合計数。単一のキー ローテーションを実行しているときは使用できません。
履歴レコードを処理する	フィールドが監査されている場合は、監査テーブルの履歴レコードを処理するように選択します。監査テーブルのフィールドの履歴レコードを暗号化する場合、新しい値と古い値の両方が暗号化されます。 このフィールドは読み取り専用で有効です。 監査フィールドの詳細については、「監査」を参照してください。
推定最大監査レコード数	処理する監査対象レコードの推定最大数。単一のキー ローテーションを実行しているときは使用できません。
有効	このジョブを非アクティブ化する場合は、このチェックボックスをオフにします。
実行	ジョブの実行間隔を選択します。
開始中	ジョブを初めて実行する日時を入力します。

3. フォーム ヘッダーのメニュー アイコンをクリックし、[保存] を選択します。
   監査対象フィールドを処理するときは [推定レコード数] はサポートされません。

一括キー ローテーション ジョブをスケジュールする

以前のキーで暗号化されたデータを検索し、現在のデフォルトの暗号化キーでデータを再暗号化するジョブをスケジュールします。データは復号化されてから、現在のデフォルトキーで再暗号化されます。

始める前に
必要なロール：security_admin

手順

1. 移動先 すべて > エッジ暗号化の設定 > メンテナンス > 一括キーローテーションをスケジュール.
2. 必要に応じて、フォームのフィールドに入力します。

フィールド	値
名前	内容を端的に表す名前を入力します。
ジョブタイプ	[一括キー ローテーション] を選択します。

フィールド	値
推定レコード数	処理するレコードの推定合計数。一括キー ローテーションを実行しているときは使用できません。
履歴レコードを処理する	フィールドが監査されている場合は、監査テーブルの履歴レコードを処理するように選択します。監査テーブルのフィールドの履歴レコードを暗号化する場合、新しい値と古い値の両方が暗号化されます。 このフィールドは読み取り専用で有効です。 監査フィールドの詳細については、「監査」を参照してください。
推定最大監査レコード数	処理する監査対象レコードの推定最大数。一括キー ローテーションを実行しているときは使用できません。
有効	このジョブを非アクティブ化する場合は、このチェックボックスをオフにします。
実行	ジョブの実行間隔を選択します。
開始中	ジョブを初めて実行する日時を入力します。

3. フォーム ヘッダーのメニュー アイコンをクリックし、[保存] を選択します。
監査対象フィールドを処理するときは [推定レコード数] はサポートされません。

添付ファイルのキー ローテーション ジョブをスケジュールする

指定したキー エイリアスを使用して暗号化された添付ファイルを検索し、現在のデフォルトの暗号化キーで添付ファイルを再暗号化するジョブをスケジュールします。添付ファイルは復号化されてから、デフォルトキーで再暗号化されます。

始める前に
必要なロール：security_admin

手順

1. 移動先 すべて > エッジ暗号化の設定 > メンテナンス > 添付ファイルのキーローテーションをスケジュール.
2. 必要に応じて、フォームのフィールドに入力します。

フィールド	値
名前	内容を端的に表す名前を入力します。
ジョブ タイプ	[添付ファイルのキー ローテーション] を選択します。
有効	このジョブを非アクティブ化する場合は、このチェックマークをオフにします。
テーブル	テーブルを選択します。
実行	ジョブの実行間隔を選択します。
開始中	ジョブを初めて実行する日時を入力します。

3. フォーム ヘッダーのメニュー アイコンをクリックし、[保存] を選択します。

自動翻訳

4. 更新するレコードの推定数を確認するには、[推定レコード数] をクリックします。
5. ジョブを直ちに実行するには、[今すぐ実行] をクリックします。

暗号化設定を使用したフィールド暗号化

暗号化設定を作成してフィールドを暗号化します。

エッジ暗号化 の設定を行うには、プロキシ経由でインスタンスに接続する必要があります。本番インスタンスに変更を適用する前に、非本番インスタンスですべての変更をテストしてください。

暗号化キーの定義

1 つ以上のプロキシをセットアップし、デフォルトの暗号化キーを設定した後、すべてのプロキシがそのキーを使用できることがインスタンスによって確認されます。すべてのプロキシに暗号化キーが存在しない限り、その暗号化キーをデフォルトキーにすることはできません。デフォルトキーが定義されると、暗号化設定を作成できるようになります。

暗号化するフィールドと添付ファイルの割り当て

「暗号化するフィールドと添付ファイルを割り当てる」とは、フィールドまたは添付ファイルに暗号化タイプを割り当てることを意味します。フィールドを暗号化対象としてマークする前に、次の問題について検討してください。

- どのシステム機能が影響を受ける可能性があるかを判断します。
- すべてのスクリプトについてフィールドを使用していないかを調べます。
- フィールドのサイズを望ましいサイズに調整します。フィールド暗号化を完了した後は、フィールドサイズを変更できません。

暗号化するフィールドをマークすると、暗号化データが格納されるように、フィールドサイズが拡張されます。テーブルのレコード数によってはフィールドサイズを拡張するプロセスに長時間かかることがあります。

API サポート

フィールド暗号化 は、暗号化フィールドに暗号化されたデータを挿入できるように、setDisplayValue() および setValue() API を更新します。また、getDisplayValue() と getValue() がクリアテキスト値を返すようにもなります。

次のスクリプトは、インシデントの簡単な説明が暗号化されている場合の API の変更を示しています。

```
var gr = new GlideRecord('incident'); //creates a new incident
gr.setValue('short_description','test123'); //sets the value to test123
var sys_ID = gr.insert(); //inserts the record in the Incident table.
gs.info(gr.getValue('short_description')); //displays the unencrypted value
```

getValue() を使用して暗号化テキストを取得すると、スクリプトで暗号テキストを返されなくなります。スクリプトは、ユーザーが暗号化モジュールにアクセスできるという想定で、プレーンテキストを返します。ユーザーが暗号化モジュールにアクセスできない場合、getValue() は暗号テキストを返します。

フィールド暗号化を作成する

暗号化するフィールドを選択し、暗号化タイプを特定します。

始める前に
必要なロール：security_admin

手順

1. 移動先 すべて > エッジ暗号化の設定 > 暗号化設定 > 新規作成.
2. フォームを完了します。

フィールド	説明
テーブル	暗号化するフィールドが含まれているテーブル。
タイプ	テーブルの列を暗号化するか、テーブルの添付ファイルを暗号化するか。[列] を選択します。
列	暗号化するフィールド。[タイプ] が [列] の場合にのみ表示されます。 文字列、日付、日付/時刻、ジャーナル、ジャーナル入力、および URL フィールドのみがサポートされます。 ◦ 文字列および **URL** フィールド：親テーブルまたは子テーブルに暗号化設定を追加できます。 ◦ 日付および日付/時刻フィールド：親テーブルのみに暗号化設定を追加できます。子テーブルには、新しい暗号化設定を追加できません。 **注:** 暗号化している日付および日付/時刻フィールドの影響を受けるレコード数に応じて、暗号化設定の作成には数分かかることがあります。インスタンスのトランザクション量が低いときは、日付および日付/時刻フィールド暗号化を作成してください。
暗号化タイプ	使用する暗号化タイプ。

注: 特定のテーブルとフィールドの組み合わせには、一度に 1 つの有効な設定のみを持たせることができます。

3. [送信] をクリックします。

次のタスク
暗号化設定レコードを追加したら、既存のデータを暗号化する暗号化ジョブを作成できます。暗号化ジョブを実行しない場合は、次回のデータ変更時に Edge によって既存のデータが暗号化されます。詳細については、「暗号化ジョブをスケジュールする」を参照してください。

変数暗号化設定を作成する

暗号化するサービスカタログ変数を選択し、暗号化タイプを特定します。

始める前に
必要なロール：security_admin

手順

1. 移動先 すべて > エッジ暗号化の設定 > 変数暗号化設定.
2. [エッジ暗号化の変数設定] リストで、[新規] をクリックします。

3. フォームに入力します。

フィールド	説明
変数	暗号化する変数。
暗号化タイプ	使用する暗号化タイプ。

4. [送信] をクリックします。

次のタスク
暗号化設定レコードを追加したら、既存のデータを暗号化する暗号化ジョブを作成できます。暗号化ジョブを実行しない場合は、次回のデータ変更時に Edge によって既存のデータが暗号化されます。詳細については、「暗号化ジョブをスケジュールする」を参照してください。

暗号化設定を非アクティブ化する

暗号化するフィールドまたはテーブルの添付ファイルを設定した後で、暗号化設定を無効にして暗号化を停止することができます。暗号化を無効にした後、「フィールドの復号化ジョブ」または「添付ファイルの添付ファイル復号化ジョブ」を実行して、暗号化データをインスタンスから削除することができます。

始める前に
必要なロール：security_admin

このタスクについて

⚠ **警告:** 暗号化設定を無効にしても暗号化レコードは削除されず、暗号化タイプは変更できません。

手順

1. 移動先 エッジ暗号化の設定 > エッジ暗号化構成 > すべて.
   [エッジ暗号化の設定] リストが表示されます。
2. 無効にする暗号化設定をクリックします。
   [エッジ暗号化の設定] フォームが表示されます。
3. [有効] ボックスをクリックします。
   [有効] ボックスがクリアーされます。
4. [更新] をクリックします。
   [エッジ暗号化の設定] リストが表示されます。

次のタスク
復号化ジョブまたは添付ファイルの復号化ジョブを実行して、インスタンスのデータを復号化することができます。ジョブを実行しない場合は、次回の変更時に暗号化データが復号化されます。

暗号化ジョブをスケジュールする

フィールドに設定されたデフォルトの暗号化キーを使用して、指定したフィールド暗号化されていないデータを検索して暗号化するジョブをスケジュールできます。フィールドを暗号化対象として設定した後に暗号化ジョブを作成しない場合は、新しい値のみが暗号化されます。

始める前に
必要なロール：security_admin

手順

1. 移動先 エッジ暗号化の設定 > 暗号化設定 > すべて フィールドのジョブを作成するか、 エッジ暗号化の設定 > 変数暗号化設定 変数のジョブを作成します。
2. 暗号化ジョブをスケジュールするフィールドをクリックします。
3. [関連リンク] で、[一括暗号化ジョブをスケジュール] をクリックします。

   [スケジュール設定済みの暗号化ジョブ] フォームが表示され、すべてのフィールドが入力されます。フォームの下部には、以前のジョブの実行があった場合のレコードが表示されます。
4. 必要に応じて、フォームのフィールドに入力します。

フィールド	値
名前	内容を端的に表す名前を入力します。
有効	このジョブを非アクティブ化する場合は、このチェックボックスをオフにします。
ジョブタイプ	[暗号化] を選択します。
テーブル	テーブルを選択します。
列	列を選択します。
推定レコード数	処理するレコードの推定総数です。[推定レコード数] を選択後に入力します。
履歴レコードを処理する	フィールドが監査されている場合は、監査テーブルの履歴レコードを処理するように選択します。監査テーブルのフィールドの履歴レコードを暗号化する場合、新しい値と古い値の両方が暗号化されます。 監査フィールドの詳細については、「監査」を参照してください。
推定最大監査レコード数	処理する監査レコードの推定最大数です。[推定レコード数] を選択後に入力します。このフィールドは、[履歴レコードを処理する] が選択されている場合にのみ表示されます。 **注:** 推定数は、実際に処理されたレコードの数よりも大きくなる場合があります。
実行	ジョブの実行間隔を選択します。
開始中	ジョブを初めて実行する日時を入力します。

5. フォーム ヘッダーのメニュー アイコンをクリックし、[保存] を選択します。
6. 更新するレコードの推定数を確認するには、[推定レコード数] をクリックします。
7. ジョブを直ちに実行するには、[今すぐ実行] をクリックします。

復号化ジョブをスケジュールする

暗号化されたフィールドのデータを復号化し、インスタンスにクリアー データを格納するジョブをスケジュールできます。

始める前に

注: 復号化ジョブを実行するには、フィールド暗号化レコードを非アクティブとしてマークする必要があります ([有効] ボックスをクリアーします)。

必要なロール：security_admin

手順

1. 移動先 エッジ暗号化の設定 > 暗号化設定 > すべて フィールドのジョブを作成するか、 エッジ暗号化の設定 > 変数暗号化設定 変数のジョブを作成します。
2. 復号化するフィールドをクリックします。
3. [関連リンク] で、[一括復号化ジョブをスケジュール] をクリックします。

   [スケジュール設定済みの暗号化ジョブ] フォームが表示され、すべてのフィールドが入力されます。フォームの下部には、以前のジョブの実行があった場合のレコードが表示されます。
4. 必要に応じて、フォームのフィールドに入力します。

フィールド	値
名前	内容を端的に表す名前を入力します。
ジョブタイプ	[復号化] を選択します。
有効	このジョブを非アクティブ化する場合は、このチェックボックスをオフにします。
テーブル	テーブルを選択します。
列	列を選択します。
推定レコード数	処理するレコードの推定総数です。[推定レコード数] を選択後に入力します。
履歴レコードを処理する	フィールドが監査されている場合は、監査テーブルの履歴レコードを処理するように選択します。監査テーブルのフィールドの履歴レコードを暗号化する場合、新しい値と古い値の両方が暗号化されます。 監査フィールドの詳細については、「監査」を参照してください。
推定最大監査レコード数	処理する監査レコードの推定最大数です。[推定レコード数] を選択後に入力します。このフィールドは、[履歴レコードを処理する] が選択されている場合にのみ表示されます。 注: 推定数は、実際に処理されたレコードの数よりも大きくなる場合があります。
実行	ジョブの実行間隔を選択します。
開始中	ジョブを初めて実行する日時を入力します。

5. フォーム ヘッダーのメニュー アイコンをクリックし、[保存] を選択します。
6. 更新するレコードの推定数を確認するには、[推定レコード数] をクリックします。
7. ジョブを直ちに実行するには、[今すぐ実行] をクリックします。

標準暗号化を使用して添付ファイルを暗号化する

特定のテーブルについて添付ファイルを暗号化できます。

テーブルに対するすべての添付ファイルには同じ暗号化タイプが使用されます。テキスト検索を実行するときに、暗号化された添付ファイルは検索されません。添付ファイルには標準の暗号化タイプのみ使用できます。順序保存または等価性保存の暗号化タイプは使用できません。

エッジ暗号化 プロキシをバイパスするセッションの場合は次のようになります。

- 添付ファイルの暗号化を有効にしたレコード：
  - ユーザーは、添付ファイルと添付ファイル名が存在することを確認できます。
  - ユーザーは、新しい添付ファイルを追加できません。
- 添付ファイルの暗号化を有効にしていないレコード：
  - ユーザーは、既存の添付ファイルを開いてダウンロードできます。
  - ユーザーは、新しい添付ファイルを追加できます。

暗号化プロキシを使用するセッションの場合、ユーザーは、既存の添付ファイルを開いてダウンロードでき、新しい添付ファイルを追加できます。

添付ファイルの暗号化を設定する

その添付ファイルを暗号化するテーブルを選択して、暗号化タイプを特定します。

始める前に

必要なロール：security_admin

手順

1. 移動先 すべて > エッジ暗号化の設定 > エッジ暗号化構成 > 新規作成.
2. 必要に応じて、フォームのフィールドに入力します。

エッジ暗号化の設定

フィールド	説明
テーブル	その添付ファイルを暗号化するテーブルを選択します。
タイプ	テーブルの列を暗号化するか、テーブルの添付ファイルを暗号化するか。[添付ファイル] を選択します。
列	暗号化するテーブル フィールド。 このフィールドは、[タイプ] が [列] の場合に表示され、[タイプ] が [添付ファイル] の場合は表示されません。
暗号化タイプ	使用する暗号化タイプ。添付ファイルの場合は、標準 AES128 と標準 AES256 のみが使用できます。

3. [送信] をクリックします。

次のタスク

暗号化レコードを追加したら、既存の添付ファイルを暗号化する添付ファイルの暗号化ジョブを作成できます。添付ファイルの暗号化ジョブを実行しない場合は、新しい添付ファイルを添付したときに添付ファイルが暗号化されます。

注: テーブルの [コレクションディクショナリー] エントリで *edge_encryption_clear_attachment_allowed* 属性 を **True** にマークすると、エッジ暗号化で暗号化されていない添付ファイルがテーブルに追加されます。この属性を有効にする場合は、「添付ファイルの暗号化ジョブ」をセットアップして、追加された暗号化されていない添付ファイルが暗号化されるようにする必要があります。

添付ファイルの暗号化ジョブをスケジュールする

テーブルに設定されたデフォルトの暗号化キーを使用して、指定したテーブルの暗号化されていない添付ファイルを検索して暗号化するジョブをスケジュールできます。

始める前に

必要なロール：security_admin

手順

1. 移動先 エッジ暗号化の設定 > 暗号化設定 > すべて.
2. 暗号化ジョブをスケジュールするテーブルをクリックします。
3. [関連リンク] で、[一括暗号化ジョブをスケジュール] をクリックします。

   [スケジュール設定済みの暗号化ジョブ] フォームが表示され、すべてのフィールドが入力されます。フォームの下部には、以前のジョブの実行があった場合のレコードが表示されます。
4. 必要に応じて、フォームのフィールドに入力します。

フィールド	値
名前	内容を端的に表す名前を入力します。
有効	このジョブを非アクティブ化する場合は、このチェックボックスをオフにします。
ジョブ タイプ	[添付ファイルの暗号化] を選択します。
テーブル	テーブルを選択します。
実行	ジョブの実行間隔を選択します。
開始中	ジョブを初めて実行する日時を入力します。

5. フォーム ヘッダーのメニュー アイコンをクリックし、[保存] を選択します。
6. 更新するレコードの推定数を確認するには、[推定レコード数] をクリックします。
7. ジョブを直ちに実行するには、[今すぐ実行] をクリックします。

添付ファイルの復号化ジョブをスケジュールする

指定したテーブルの暗号化された添付ファイルを復号化し、クリアーな添付ファイルをインスタンスに格納するジョブをスケジュールできます。

始める前に

注: 復号化ジョブを実行する前に、テーブルの暗号化レコードを非アクティブとしてマークする必要があります ([有効] ボックスをクリアーします)。

必要なロール：security_admin

手順

1. 移動先 エッジ暗号化の設定 > 暗号化設定 > すべて.
2. 復号化する添付ファイルを持つテーブルをクリックします。
3. [関連リンク] で、[一括添付ファイル復号化ジョブをスケジュール] をクリックします。

   [スケジュール設定済みの暗号化ジョブ] フォームが表示され、すべてのフィールドが入力されます。フォームの下部には、以前のジョブの実行があった場合のレコードが表示されます。
4. 必要に応じて、フォームのフィールドに入力します。

フィールド	値
名前	内容を端的に表す名前を入力します。
ジョブ タイプ	[添付ファイルの復号化] を選択します。
有効	このジョブを非アクティブ化する場合は、このチェックマークをオフにします。
テーブル	テーブルを選択します。
実行	ジョブの実行間隔を選択します。
開始中	ジョブを初めて実行する日時を入力します。

5. フォーム ヘッダーのメニュー アイコンをクリックし、[保存] を選択します。
6. 更新するレコードの推定数を確認するには、[推定レコード数] をクリックします。
7. ジョブを直ちに実行するには、[今すぐ実行] をクリックします。

フィールドまたは添付ファイルの暗号化タイプを変更する

フィールドまたは添付ファイルの暗号化タイプを変更するには、既存の暗号化設定レコードで新しい暗号化タイプを選択します。特定のテーブルとフィールドの組み合わせには、一度に 1 つの有効な設定のみを持たせることができます。

始める前に
必要なロール：security_admin

手順

1. 移動先 エッジ暗号化の設定 > 暗号化設定 > すべて.
   [エッジ暗号化の設定] リストが表示されます。
2. 暗号化設定を変更するレコードを開きます。
3. [暗号化タイプ] ドロップダウンをクリックし、新しい暗号化タイプを選択します。

   注: 添付ファイルの場合は、標準 AES128 と標準 AES256 のみが使用できます。
4. 必要に応じて、暗号化または添付ファイルの暗号化ジョブを実行します。
   暗号化ジョブを実行する必要はありません。暗号化ジョブを実行しない場合、フィールドまたは添付ファイルは、次回フィールドまたは添付ファイルが変更されたときに新しい暗号化タイプを使用して暗号化されます。

暗号化パターンを使用して文字列をトークン化する

文字列パターンをトークンに置き換え、それからトークンをインスタンスに送信して格納することができます。

始める前に

暗号化パターンを使用するには、ネットワークに MySQL データベースをインストールしてセットアップする必要があります。これは、順序保存の暗号化に使用されるものと同じデータベースです。暗号化パターンを作成または編集するには、プロキシ経由でインスタンスに接続する必要があります。

必要なロール：security_admin

このタスクについて

ベース システムのパターンを使用するか、独自のパターンを作成できます。ベースシステムのパターンは詳細パターンです。暗号化パターンには次の制限があります。

- すべてがアルファベット文字のパターンは使用できません。
- 最小パターン サイズは 5 文字です。システムプロパティを使用してこの設定を変更することができます。
- 暗号化パターンでは * および + の数量指定子は禁止されています。
- 暗号化パターンは、完全な単語に一致し、より大きい文字列に埋め込まれた文字列の一部分には一致しません。単語は、パターンに含めることができない空白と文字で定義されます。
- 同じ文字列が複数回インスタンスに送信された場合は、同じトークンに置き換えられます。
- 完全一致のテキスト検索がサポートされています。クエリー文字列は、インスタンスに送信されたときにトークンと交換され、トークンに対して検索が実行されます。検索結果がプロキシサーバーに返されると、トークンがクリアー テキストに置き換えられます。語幹解釈などの機能はサポートされていません。

パターンを使用しているときは、クリアー テキストがネットワークの外に出ることは決してありません。インスタンスに送られる要求の中で、プロキシサーバーがパターンを照合するときに、プロキシによって文字列が同じサイズのトークンに置き換えられます。このトークンが、クリア テキスト文字列の代わりにインスタンスに送られます。インスタンスからプロキシサーバーに応答が送られるときは、プロキシによってトークンが文字列に置き換えられます。この文字列は、プロキシサーバーを介して表示されるときにクリア テキストとして表示されます。

注: 暗号化されたフィールドでは暗号化パターンのチェックは行われません。

手順

1. 移動先 すべて > エッジ暗号化の設定 > 暗号化パターン > 新規作成.
   または、[詳細パターン] に移動して、事前設定されたパターンを有効化または編集することもできます。
2. パターン名を入力します。
3. [エッジ パターン入力タイプ] を定義します。

オプション	説明
基本	一連の文字タイプ。[基本パターンの入力] タブで、[追加] をクリックし、文字タイプを選択します。 文字が追加されるたびに [サンプル パターン] にパターンが表示されます。

オプション	説明
	[新規ブロック] をクリックし、次の文字を次の行に移動します。これで、文字を長いパターンにグループ化できます。 パターンの最後の文字を削除するには **[X]** をクリックします。
詳細	Java の正規表現式。[詳細] を選択した場合、入力タイプを [基本] に戻すことはできません。 [サンプルの一致] フィールドに、正規表現式をテストするためのサンプル パターンを入力します。[パターン] フィールドに、Java の正規表現式を入力します。[検証] をクリックし、式がサンプル パターンと一致することを検証します。

入力タイプは、パターンの入力方法を定義するものです。パターンの使用方法には影響を与えません。

4. [送信] をクリックします。

順序保存の暗号化データの修復または復元

security-admin ロールを持っている場合、 エッジ暗号化 プロキシにより実行され、順序保存の暗号化を使用しているフィールドを修復または復元するジョブをスケジュールできます。

ジョブを次のようにスケジュールします。

- 順序トークンを修復する。
- プロキシデータベースを再作成する。

これらのジョブの実行には時間がかかる場合があり、エッジ暗号化 プロキシのパフォーマンスに影響することがあります。週末の深夜など、システムを使用しているユーザーがまったくいないか最小限の数であるときに、これらのジョブをスケジュールしてください。

順序トークンの修復ジョブをスケジュールする

順序トークンが失われているフィールドを検索して修復するジョブをスケジュールできます。

始める前に
必要なロール：security_admin

このタスクについて
これらのジョブを使用して、テーブル内の個々のフィールドを修復したり、順序保存の暗号化を使用しているすべてのフィールドを修復したりします。このジョブは、インスタンスの実行中にプロキシデータベースがオフラインになったときに実行します。そのようにすると、順序トークンが失われている順序保存のフィールドが得られます。

手順

1. 移動先 すべて > エッジ暗号化の設定 > メンテナンス > 順序トークンの修復をスケジュール.
2. 必要に応じて、フォームのフィールドに入力します。

フィールド	値
名前	内容を端的に表す名前を入力します。
ジョブ タイプ	[順序トークンの修復] を選択します。
すべてのフィールド	すべてのテーブルを修復するには、このチェック ボックスをオンにします。
テーブル	テーブルを選択します。
列	列を選択します。
有効	このジョブを非アクティブ化する場合は、このチェックボックスをオフにします。
実行	ジョブの実行間隔を選択します。
開始中	ジョブを初めて実行する日時を入力します。

3. フォーム ヘッダーのメニュー アイコンをクリックし、[保存] を選択します。
4. 更新するレコードの推定数を確認するには、[推定レコード数] をクリックします。

プロキシ データベースの復元ジョブをスケジュールする

このジョブは、プロキシ データベースがデータを失ったときに実行します。このジョブは、トークン (順序保存の暗号化タイプ) で暗号化されたすべてのレコードを検索し、それらをプロキシに送信して、プロキシ データベースを再ビルドできるようにします。

始める前に
必要なロール：security_admin

手順

1. 移動先 すべて > エッジ暗号化の設定 > メンテナンス > データベースの復元をスケジュール.
2. 必要に応じて、フォームのフィールドに入力します。

フィールド	値
名前	このジョブの内容を端的に表す名前を入力します。
ジョブ タイプ	[データベースの復元] を選択します。
有効	このジョブを非アクティブ化する場合は、このチェックボックスをオフにします。
実行	ジョブの実行間隔を選択します。
開始中	このジョブを初めて実行する日時を入力します。

3. フォーム ヘッダーのメニュー アイコンをクリックし、[保存] を選択します。
4. 更新するレコードの推定数を確認するには、[推定レコード数] をクリックします。

IP アドレス拒否リストを設定する

ネットワーク内の IP アドレスがインスタンスに要求を送信しないようにする

始める前に
必要なロール：security_admin
エッジ暗号化 プロキシサーバーはネットワークに常駐しているため、ネットワーク ソフトウェアによる脆弱性スキャンの影響を受ける場合があります。IP スキャナーまたは他の要求が ServiceNow インスタンスに転送されるのを防ぐために、IP アドレス、IP 範囲、またはネットワークマスクを拒否リストに追加できます。拒否リストにあるアドレスからのプロキシサーバーへの接続はすべて終了させられ、インスタンスには転送されません。

IP アドレスを拒否リストに登録するには、プロキシサーバー経由でインスタンスにログインする必要があります。

重要: ネットワークの IP アドレスを拒否リストに登録する際は、事前にネットワークトポロジについての知識が必要です。IP アドレスを拒否リストに追加した場合、その IP アドレスを持つユーザーはすべて、エッジ暗号化 プロキシサーバーへのアクセスがブロックされます。

手順

1. 移動先 すべて > エッジ暗号化の設定 > メンテナンス > 拒否リスト **IP** アドレス.
   暗号化プロキシ IP 拒否リスト [edge_encryption_ip_blacklist] リストビューが開きます。
2. [新規] をクリックします。
3. フォームに入力します。

フィールド	説明
プロキシサーバー	拒否リストに登録されたアドレスからの要求を転送できないようにする エッジ暗号化 プロキシサーバーです。
IP、IP 範囲、またはネットマスク	この IP アドレス、範囲、またはネットワーク マスクからの要求が ServiceNow インスタンスに転送されません。値の例を次に示します。 ○ IP アドレス：10.10.10.5 ○ IP 範囲：10.10.10.1-15 ○ ネットワーク マスク：10.10.10.0/24 **注:** IPv4 または IPv6 アドレスのいずれかを使用できます
有効	レコードが有効かどうか。有効なレコードからの IP アドレスのみが、インスタンスに要求を送信できなくなります。
説明	拒否リストレコードの説明です。

4. [送信] をクリックします。
5. IP アドレスを拒否リストに登録する必要がある他のすべてのプロキシについて、上記の手順を繰り返します。

結果
エッジ暗号化 プロキシサーバーは、拒否リストに登録された IP アドレス、範囲、またはネットワークマスクからの接続を終了し、要求をインスタンスに転送できません。

レコードプロデューサーからのデータの暗号化

レコードプロデューサーレコードから暗号化ルールを作成し、レコードプロデューサーからの挿入を許可するように エッジ暗号化 プロキシサーバーを設定します。

始める前に
必要なロール：security_admin
レコードプロデューサーを使用すると、エンドユーザーはインシデントレコードなどのタスクベースのレコードをサービスカタログおよびサービスポータルから作成できます。レコードプロデューサーが、暗号化対象としてマークされたフィールドにデータを挿入しようとすると、無効な挿入のメッセージが表示され、データはフィールドに保存されません。

レコードプロデューサーからのデータを暗号化するには、ターゲットフィールドに対して暗号化設定を定義する必要があります。レコード プロデューサーから暗号化ルールを作成する前に、ターゲットフィールドとテーブルの暗号化設定を作成済みであることを確認してください。「フィールド暗号化を作成する」を参照してください。レコード プロデューサーから添付ファイルを暗号化するには、「添付ファイルの暗号化を設定する」の手順を実行します。

手順

1. エッジ暗号化 プロキシサーバー経由でインスタンスにログインします。
2. 移動先 サービスカタログ > カタログ定義 > レコードプロデューサー.
3. レコードプロデューサーレコードを作成 するか、既存のレコードプロデューサーレコードを開きます。
4. [関連リンク] で、[エッジ暗号化のルールを作成] を選択します。
   暗号化対象としてにマークされたフィールドにレコード プロデューサーから送信されたデータを暗号化するための、2 つの非アクティブな暗号化ルールが自動的に作成されます。

暗号化ルール	説明
<RecordProducerName>	サービスカタログからの POST パラメーターを処理し、インスタンス内のフィールドに変数をマップするために作成されたルール。
<RecordProducerName>Json	サービスポータルからの JSON ペイロードを処理し、インスタンス内のフィールドに変数をマップするために作成されたルール。

5. レコード プロデューサーによって作成された必要な暗号化ルールを有効にします。

   a. 移動先 エッジ暗号化の設定 > ルール > すべて.

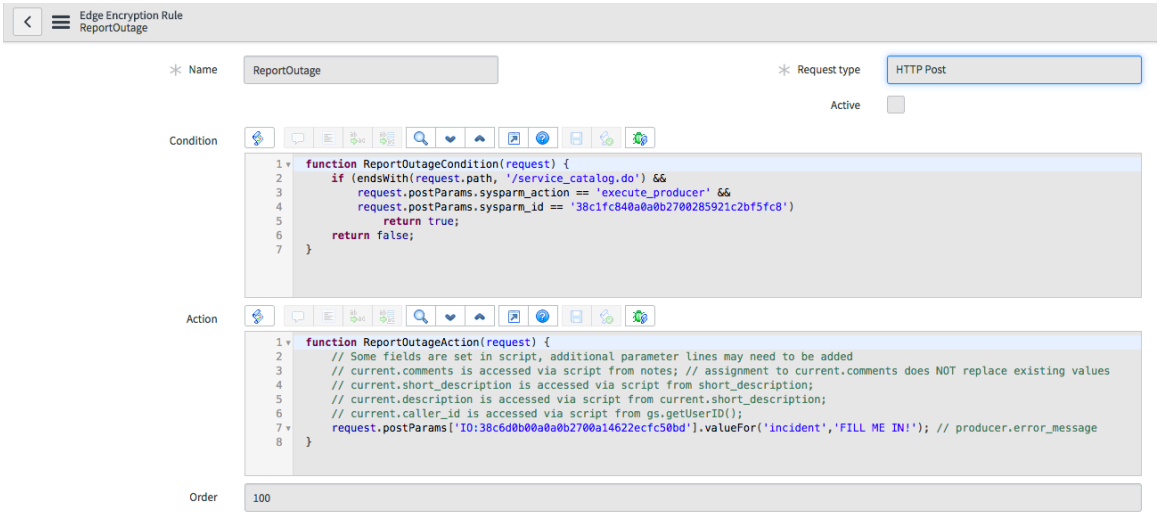
   b. レコード プロデューサーの使用場所に応じて、レコード プロデューサーによって作成された関連する暗号化ルールを開き、[有効] フラグを選択します。
      レコード プロデューサーをサービスカタログで使用する場合は、<RecordProducerName> 暗号化ルールを有効にします。レコード プロデューサーをサービスポータルで使用する場合は、<RecordProducerName>Json 暗号化ルールを有効にします。

6. オプション: 暗号化ルールの [アクション] フィールドを調べ、必要なフィールド名またはステートメントがあれば追加します。

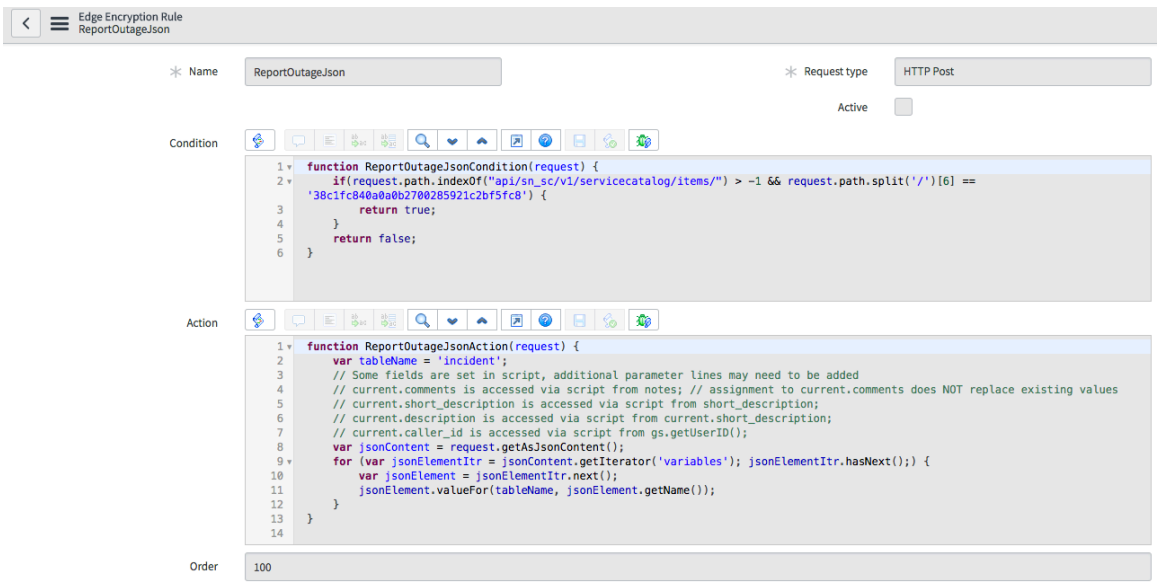
   (Optional) レコード プロデューサーが変数をテーブルのフィールドに直接マップする場合は、暗号化ルールによって変数が正しいフィールドに自動的にマップされます。ただし、変数がプラットフォームのさまざまなスクリプトを介して間接的にマップされる場合は、それぞれの変数が正しいフィールドにマップされるように、状況に応じてルールを更新する必要があります。

Example

(Optional) 次の暗号化ルールはレポート機能停止レコード プロデューサーから作成されたもので、サービスカタログからの POST パラメーターを処理してインスタンス内のフィールドに変数をマップします。'FILL ME IN' の部分をターゲットフィールドに置き換えます。



次の暗号化ルールはレポート機能停止レコード プロデューサーから作成されたもので、サービスポータルからの JSON ペイロードを処理してインスタンス内のフィールドに変数をマップします。スクリプト化された変数をターゲットフィールドにマップするステートメントを追加します。



レコード プロデューサーからのペイロードを調べてみると、error_message 要素に short_description フィールドの値が格納されています。次のステートメントを追加することで、スクリプト化された変数 error_message を short_description フィールドにマップできます。

```
if (jsonElement.getName() == 'error_message')
      jsonElement.valueFor(tableName, 'short_description');
```

[アクション] フィールドの値は次のようになります。

```
function ReportOutageJsonAction(request) {
 var tableName = 'incident';
```

```
// Some fields are set in script, additional parameter lines may need to be added
// current.comments is accessed via script from notes; // assignment to current.comments
does NOT replace existing values
// current.short_description is accessed via script from short_description;
// current.description is accessed via script from current.short_description;
// current.caller_id is accessed via script from gs.getUserID();
var jsonContent = request.getAsJsonContent();
for (var jsonElementItr = jsonContent.getIterator('variables'); jsonElementItr.hasNext();) {
 var jsonElement = jsonElementItr.next();
 if (jsonElement.getName() == 'error_message')
  jsonElement.valueFor(tableName, 'short_description');
            } else {
        jsonElement.valueFor(tableName, jsonElement.getName());
}
}
```

結果
2 つの暗号化ルールにより、サービスカタログとサービスポータルのどちらでも、暗号化対象としてマークされたフィールドにレコード プロデューサーから値を挿入できます。

カスタムの暗号化ルールを定義する

場合によっては、HTTP 要求の中で、要求がインスタンスに送られる途中で機密情報を特定して暗号化する必要があります。このような要求のデータを識別、解釈し、暗号化するための暗号化ルールを記述して、要求内のフィールドをインスタンスのテーブルフィールド名にマップすることができます。

暗号化ルールとは

暗号化ルールは、エッジ暗号化 プロキシサーバーで実行され、要求内のフィールドを ServiceNow インスタンス上のテーブル内のフィールドにマップするスクリプトです。暗号化ルールでは、カスタム ペイロードでのデータの暗号化方法を エッジ暗号化 プロキシサーバーに指示します。

注: 暗号化ルールでは ECMAScript 3 またはそれ以前のみがサポートされています。

どのような場合にカスタム ルールを使用するか

エッジ暗号化 プラグインの一部として、1 組の暗号化ルールが組み込まれています。これらのルールは、次のような多くのコアプラットフォームのユースケースを処理します。

- リスト編集フォームからのフィールドの編集
- レコードフォームからのレコードの更新
- ダイレクト Web サービスの管理
- REST アプリケーションプログラムインターフェイス (API) からのデータの処理

標準のフォームとリストを使用して作成されたアプリケーションは、カスタム暗号化ルールがなくても機能します。

暗号化を必要とするデータが含まれているスクリプトを開発する場合は、データを検索して Glide テーブルフィールド名にマップするための暗号化ルールを作成します。例：

- スクリプトプロセッサー
- スクリプト Web サービス
- Scripted REST API、UI、または Ajax スクリプト

暗号化ルールの形式

ルールには 3 つの要素が含まれています。

- 条件：要求のタイプを識別します。
- アクション：要求内のフィールドをテーブル内のフィールドにマップし、暗号化設定が定義されたフィールドにマップされる値を暗号化します。
- 順序：ルールの優先度。優先度が最も低く、条件が満たされたルールのみが実行されます。ビジネスルールと同様に、ルールは最低から最高の順に実行されます。

添付ファイル要求を除き、HTTP 要求は エッジ暗号化 プロキシサーバーによって評価されます。エッジ暗号化 プロキシサーバーは、すべての条件が False を返すか、または 1 つの条件が True を返すまで、暗号化ルールのすべての条件を優先度順に評価します。条件が True を返すと、要求に対してアクションが実行され、結果がインスタンスに転送されます。他の条件は評価されません。したがって、暗号化ルールの条件はできるだけ具体的にする必要があります。汎用的なルールでは、別のルールによる処理を想定した要求に対して True と評価され、正しくないアクションによって要求が処理される可能性があります。汎用的な条件が避けられない場合は、より具体的なルールが先に評価されるように、ルールを高い優先度の値でマークする必要があります。

暗号化ルールを作成する際のガイドライン

効率的で最適化された暗号化ルールを作成することで、スクリプトの検証処理の時間を短縮できます。

全体的なガイドライン：ルールが長くなる場合は、ブロック数を最小限に抑え、可能な限りルールを分割してください。理想的には、カスタムルールで複数のケースを網羅するのではなく、特定のユースケースに適用し、アクションスクリプトで `if` ステートメントや `switch` ステートメントを使用します。

1. ルールを可能な限り分割します。例を次に示します。
   - テーブルごとに異なるルールを作成し、それぞれのルールがそれぞれのテーブルでのみ実行されるようにします。
   - ターゲットとするレコードプロデューサーごとに、または少なくともレコードプロデューサーのサブセットごとに、異なるルールを作成します。1 つのルールで数十もの `sys_ids` を対象にするのではなく、より少数のレコードプロデューサーを対象にした複数の異なるルールを作成できます。`sys_id` ごとに 1 つずつルールを作成することもできます。

     注: 複数のルールを作成すると、それだけ多くのメンテナンスが必要になります。複数のシンプルなルールの方が、長く複雑なルールよりも効率的に検証できます。

2. ブロック数を最小限に抑えます。スクリプトの評価中は処理エンジンによって個々のブロックがスキャンされます。そのため、ブロックが多数あると検証が遅れる原因になります。例を次に示します。
   - すべての `if` ブロックを配列参照に置き換え、配列参照内のすべてのブロックをただ 1 つの `if` ブロックに置き換えます。
   - 可能な限り、`if` ブロックを結合してグループ化します。

暗号化ルール API

暗号化ルールは JavaScript で記述されており、エッジ暗号化 の API を利用して、要求の本体に含まれている機密情報を検索して暗号化します。API は、xPath に似た式を使用して、JSON と XML の両方のコンテンツを参照します。

エッジ暗号化 API は、要求が出力ストリームに書き込まれているときはストリームから離れて要求を処理します。ストリームを解析することにより、暗号化ルールをネットワークで実行することができます。ただし、本体から複数回コンテンツを取得して解析した場合、予期しない結果につながる可能性があります。この潜在的な問題を回避するには、アクションによる要求の処理は 1 回のパスで行う必要があります。

暗号化ルールを作成する際、Glide API、スクリプトインクルード、ビジネス ルール、または `current` などのグローバルパラメーターは使用できません。ルールは HTTP オブジェクトに対して作成されるため、グローバルの `request` オブジェクトを使用できます。

暗号化ルールを作成する際は、許可リストマネージャーやスコープ対象のアプリケーションから API を使用することはできません。

エラー処理

暗号化ルールの条件またはアクションから例外がスローされた場合は、プロキシのログでトラブルシューティング情報がないか確認してください。

クライアント要求を調べる

カスタムの暗号化ルールを作成する前に、エッジ暗号化 プロキシサーバーに入力されるクライアント要求の形式を決定する必要があります。

始める前に

必要なロール：admin

このタスクについて

暗号化ルールは、クライアント要求に対して繰り返され、状況に応じて何を暗号化する必要があるかを判断するので、作成するルールの対象となる要求のタイプについて理解する必要があります。クライアント要求の形式によって、暗号化ルールの構造と、ルールで使用できる API が決まります。

手順

1. クライアント要求を調べます。

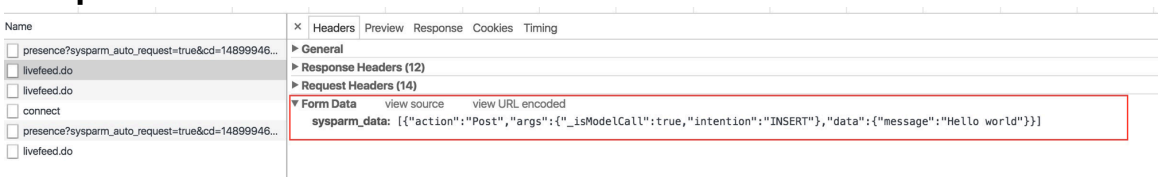
   要求のソースに応じて、次のツールを利用して要求を調べ、形式を判断することができます。

要求のソース	利用可能なツール
クライアント ブラウザー	ブラウザーの開発者用コンソールを使用して、クライアント要求を調べます。次のツールが役に立ちます。 ○ Firefox ネットワーク モニター ○ Chrome ネットワーク パネル
サードパーティ/外部ソース	HTTP プロトコル アナライザーを使用して要求を調べます。次のツールが役に立ちます。 ○ Wireshark ○ HTTP Scoop

要求のソース	利用可能なツール
	または多くの場合、外部ソースのドキュメントを使用して要求の形式を判断することもできます。

2. クライアント要求から、パケットを調べて次のものを判断します。
   - クライアント要求のメソッド
   - 要求の URL パス
   - URL パラメーター
   - POST パラメーター (存在する場合)
   - 要求本体の形式 (含まれている場合)

Example



結果
要求を調べることで、暗号化ルールの中でフィルタリングや反復処理を必要とするフィールドを把握できます。*request* オブジェクトのフィールドを理解するには、「要求」を参照してください。

暗号化ルールを作成する

暗号化ルールは、HTTP 要求の中で暗号化する必要のある内容を検索するために、プロキシによって使用されます。

始める前に
必要なロール：security_admin
暗号化ルールを作成する前に、「クライアント要求を調べる」の手順を実行して形式を決める必要があります。

このタスクについて
暗号化ルールを作成または編集するには、暗号化プロキシ経由でインスタンスに接続する必要があります。

手順

1. 移動先 すべて > エッジ暗号化の設定 > ルール > 新規作成.
2. [名前] ボックスに名前を入力します。
3. [要求タイプ] で HTTP メソッドを選択します。
   - **HTTP POST**
   - **HTTP Get**
   - **HTTP PUT**
   - **HTTP Patch**
   - **HTTP Delete**

> 注: Jakarta 以前のインスタンスでは **[HTTP Get]** メソッドと **[HTTP Post]** メソッドのみを使用できます。

4. [条件] ボックスに、どの場合にルールを実行するかを定義する JavaScript ステートメントを入力します。
5. [アクション] ボックスに、条件が True のときに実行する JavaScript 関数を入力します。
6. [順序] ボックスに、ルールの相対的な優先度を入力します。
7. [送信] をクリックするか、フォームを保存します。

暗号化ルールの条件

暗号化ルールの条件は、ルールを実行する必要があるかどうかを決定します。

ルールにより HTTP 要求を処理する必要がある場合は、暗号化ルールの条件が True を返す必要があります。それ以外の場合は False を返す必要があります。

条件を作成する際は、要求ごとに 1 つのルールしか実行されないことに注意してください。そのため、意図した状況の下で実行されるように、必要に応じて条件を汎用的にしたり具体的にしたりする必要があります。

> 注: 条件内の内容のチェックを実行するときは注意が必要です。過度のチェックはプロキシサーバーに大きな負荷がかかることがあります。また、複雑な要求を処理するときに遅延が増大する要因となる場合があります。

条件では、メソッドのタイプ、コンテンツ タイプ、URL パス、または URL クエリー文字列パラメータを使用して、ルールで要求を処理する必要があるかどうかを判断できます。条件からは、要求オブジェクトを介してこれらのフィールドにアクセスできます。暗号化ルールの条件を作成するにあたっては、必ず事前にクライアント要求を調べ、ルールをトリガーするために必要な条件を理解しておいてください。

> 注: 効率的なルールを作成するため、ルールで評価する必要のない要求を除外するための簡単な方法を検討してください。最初にそれらの要求に対して False を返すように条件を作成します。これによりパフォーマンスが向上し、要求が正しいルールにルーティングされるスピードが速まります。

暗号化ルールのオブジェクトと API を暗号化ルールの条件で使用できます。

path と **postParams** を使用した例

```
/*This condition checks if the request coming in has a path ending in
"/sample_processor.do" and if a post parameter exists in that request called myPostParam */

function SampleCondition(request) {
  if (endsWith(request.path, "/sample_processor.do") && request.postParams.myPostParam) {
     return true;
  }
  return false;
}
```

urlParams と **contentType** を使用した例

```
/* This condition checks if a url parameter exists in the query called
myUrlParam and if the content type contains 'xml'
(if so, you can expect the body to be an XML payload).
Then, it checks if the xml payload contains myXmlTag */
```

```
function SampleCondition2(request) {
  if (request.urlParams.myUrlParam && request.contentType.indexOf('xml') > -1 &&
 request.xmlContains('myXmlTag')) {
      return true;
  }
 return false;
}
```

暗号化ルールのアクション

暗号化ルールにより、クライアント要求内のフィールドがインスタンス上のテーブルのフィールドにマップされ、暗号化対象としてマークされたフィールドが識別されます。

暗号化ルールのアクションは、暗号化ルールの条件が True を返すときのみ実行されます。暗号化ルールでは、要求ペイロードで暗号化するデータが識別されます。ルールは要求オブジェクト内のコンテンツに対して繰り返されるので、要求本体の形式と構造を理解して、要求内の何を暗号化する必要があるかを判断する必要があります。暗号化するデータは、次の場所に存在する可能性があります。

- POST または URL パラメーター。
- POST または URL パラメーター内部の JSON または XML の内容。
- JSON ペイロード。
- XML ペイロード。

暗号化ルールのアクションを記述するにあたっては、事前に次のことを確認してください。

- クライアント要求を調べる。
- 機密データが request オブジェクト内のどこにあるのかを特定します。
- データを挿入するフィールドおよびテーブル名を決めます。または、それらを要求から動的に抽出する方法について理解します。

暗号化ルールのオブジェクトと API を暗号化ルールのアクションと条件で使用できます。

暗号化ルールのオブジェクトと **API**

暗号化ルール API を使用して、エッジ暗号化 プロキシサーバー経由でインスタンスに送られる要求の中の値を解析し、暗号化します。

暗号化ルールに使用できる API は、request オブジェクトの形式によって異なります。たとえば、request オブジェクトの contentType パラメータが XML の場合は、XML API を使用してペイロード内の値を解析し暗号化することができます。要求内のオブジェクトのタイプを決めたら、API を使用して暗号化ルールを作成できます。

暗号化ルール API は、暗号化ルールの条件スクリプトとアクション スクリプトの両方で使用できます。

要求

request オブジェクトは、エッジ暗号化 のルールのアクション スクリプトと条件スクリプトで使用できるグローバル オブジェクトです。

request オブジェクトは、エッジ暗号化 プロキシサーバーに送られるクライアント要求を表す JavaScript オブジェクトです。作成する暗号化ルールでは、*request* オブジェクトを解析し、*request* オブジェクトの値をインスタンス上のテーブルのフィールドにマップし、*request* オブジェクト内に機密データがある場合にそのデータを暗号化する必要があります。

`request` オブジェクトには、クライアント要求から得られる次の属性とデータが含まれています。

request オブジェクトのフィールド

フィールド	説明
path	URL のパス部分。
requestMethod	GET、POST、PUT、PATCH、DELETE。
contentType	Content-Type ヘッダー フィールド。
urlParams	クエリー文字列内のパラメーター。これは文字列に評価される可能性もあります。
postParams	これがフォーム ポストの場合は、ポスト パラメーターが含まれています。

request - getAsJsonContent()

要求を `JsonNode` タイプの反復可能なオブジェクトとして返します。

このメソッドは、要求本体が有効な JSON ペイロードである場合に、エッジ暗号化 のルールでのみ使用できます。要求本体にどのような形式が含まれているかわからない場合は、`request` オブジェクトの contentType フィールドを調べます。

要求が `JsonNode` オブジェクトとして返されたら、JSON API を使用してオブジェクトを反復処理し、フィールドを暗号化できます。

パラメーター

名前	タイプ	説明
なし		

返される内容

タイプ	説明
JsonNode	反復可能な `JsonNode` としての要求。

request - getAsXmlContent()

要求の内容を `XMLContent` タイプの反復可能なオブジェクトとして返します。

このメソッドは、要求本体が有効な XML ペイロードである場合に、エッジ暗号化 のルールでのみ使用できます。要求本体にどのような形式が含まれているかわからない場合は、`request` オブジェクトの contentType フィールドを調べます。

要求が `XMLContent` オブジェクトとして返されたら、XML API を使用してオブジェクトを反復処理し、フィールドを暗号化できます。

パラメーター

名前	タイプ	説明
なし		

返される内容

タイプ	説明
XMLContent	*XMLContent* タイプの反復可能なオブジェクトとしての要求。

request - XMLContains(String path)

指定したパスが XML DOM に存在する場合に True を返します。

このメソッドは、要求本体が有効な XML ペイロードである場合にのみ使用できます。要求本体にどのような形式が含まれているかわからない場合は、*request* オブジェクトの contentType フィールドを調べます。

パラメーター

名前	タイプ	説明
path	文字列	検索する XPath ステートメント。

返される内容

タイプ	説明
ブーリアン	指定したパスが XML DOM に存在するかどうか。

POST パラメーターと **URL** パラメーターの **API**

POST パラメーターと URL パラメーターは、request.postParams および request.urlParams を使用して、*request* オブジェクトのプロパティとしてアクセスできます。

request.postParams.myParam を呼び出すことで、任意の単一のパラメーターに *postParams* および *urlParams* の親オブジェクトとしてアクセスできます。この方法でアクセスするパラメーターはすべて、基底クラス *ParameterValue* のオブジェクトです。このクラス内の任意の API を任意のパラメーターで呼び出すことができます。

クライアント要求を調べる作業の後、場合によっては *request* オブジェクトからパラメーター値にアクセスして暗号化する必要があります。クライアント要求内のデータに応じて、複数の方法で値を暗号化し、インスタンスのフィールドにマップすることができます。

既知のテーブルおよびフィールドの値を暗号化する

暗号化データを保持するインスタンスのテーブルおよびフィールドの名前がわかっている場合は、それらを暗号化ルールで明示的に定義できます。たとえば、インスタンスでインシデントを作成するために要求が処理されることが分かっていて、説明フィールドの **text** パラメーターを暗号化する必要があるとします。この場合、次のアクションを作成できます。

```
function SampleAction1() {
   request.postParams.text.valueFor('incident', 'description');
}
```

動的に定義したテーブルおよびフィールドの値を暗号化する

逆に、暗号化データを設定するフィールドの名前がわからない場合は、**tableName** および **fieldName** を使用してそれらを動的に定義することができます。

次の例は、インスタンスのさまざまなタスク テーブル (インシデント、問題、change_request など) にデータを格納する可能性のある汎用的な要求を処理します。

```
function SampleAction2() {
    var tableName = request.urlParams.table;
    for (var parameter in request.postParams) {
        var currentParam = request.postParams[parameter];
        var fieldName = currentParam.toString();
        if (fieldName == 'text') {
            currentParam.valueFor(tableName, 'description')
        } else {
            currentParam.valueFor(tableName, fieldName);
        }
    }
}
```

このアクションでは次のことを実行します。

- URL パラメーターから宛先テーブルを取得します。
- URL パラメーターを反復処理します。
- 暗号化対象としてマークされたフィールドに一致する名前を持つすべての URL パラメーターを暗号化するよう、エッジ暗号化 プロキシサーバーに指示します。
- `text` という具体的なパラメーターを探し、インシデント テーブルの説明フィールド暗号化に基づいて値を暗号化するよう エッジ暗号化 プロキシに指示します。

この例で、valueFor() メソッドは実際には暗号化を実行していません。そうではなく、このメソッドは、要求オブジェクト内のテーブル/フィールドのペアが暗号化設定で暗号化対象としてマークされているかどうかを確認し、必要に応じて暗号化するよう、エッジ暗号化 プロキシサーバーに指示しています。

パラメーター内の **JSON** または **XML** を暗号化する

POST パラメーターまたは URL パラメーターには、JSON または XML コンテンツが含まれています。この場合、パラメーター内のコンテンツを処理し、値を反復処理し、必須フィールドを暗号化することができます。次の例で、**tableName** は引き続き POST パラメーターからアクセスされますが、フィールドの値は JSON オブジェクトの **data** です。

```
function SampleAction3() {
    var tableName = request.postParams.table;
    var data = request.postParams.data;
    var dataIterator = data.getAsJsonContent().iterator();
    while (dataIterator.hasNext()) {
        var jsonElement = dataIterator.next();
        var fieldName = jsonElement.getName();
        if (fieldName == 'text') {
            jsonElement.valueFor(tableName, 'description');
        } else {
            jsonElement.valueFor(tableName, fieldName);
        }
    }
}
```

POST パラメーター内の XML を処理する暗号化ルールのアクションの例

```
function SampleAction4() {
    var tableName = request.postParams.table;
    var data = request.postParams.data;
    var dataIterator = data.getAsXmlContent().getIteratorOverAllChildren();
```

```
    while (dataIterator.hasNext()) {
        var jsonElement = dataIterator.next();
        var fieldName = jsonElement.getName();
        if (fieldName == 'text') {
            jsonElement.valueFor(tableName, 'description');
        } else {
            jsonElement.valueFor(tableName, fieldName);
        }
    }
}
```

クエリーを暗号化する

クライアント要求のパラメーターの中にエンコードされたクエリーが出現し、そこに機密データが含まれていることがあります。クエリーのフィールドを、インスタンスデータベースの暗号化された値に一致させるには、クエリーのフィールドが暗号化対象としてマークされているかどうかを確認するようプロキシに指示する暗号化ルールを作成する必要があります。encodedQueryFor() メソッドは、指定したテーブルのエンコードされたクエリーを解析し、クエリーのフィールドに暗号化設定があるかどうかを確認します。

この例では、ルールでパラメーターを反復処理して **filter** パラメーターを探します。このパラメーターは Glide のエンコードされたクエリーであると想定しています。

```
function SampleAction5() {
    var tableName = request.urlParams.table;
    for (var parameter in request.postParams) {
        var currentParam = request.postParams[parameter];
        var fieldName = currentParam.toString();
        if (fieldName == 'filter') {
            currentParam.encodedQueryFor(tableName);
        } else {
            currentParam.valueFor(tableName, fieldName);
        }
    }
}
```

たとえば、**filter** の値が `short_description=My sensitive information^number=INC000056^category=Outage` の場合、クエリーはインスタンス上で `short_description=<Encrypted(My sensitive information)>^number=INC000056^category=Outage` となります。

ParameterValue - toString()

POST または URL パラメーターの値を文字列に変換します。

パラメーター

名前	タイプ	説明
なし		

返される内容

タイプ	説明
文字列	文字列としてのパラメーター値。

ParameterValue - getAsJsonContent()

要求を `JsonNode` タイプの反復可能なオブジェクトとして返します。

このメソッドは、要求本体が有効な JSON ペイロードである場合に、エッジ暗号化 のルールでのみ使用できます。要求本体にどのような形式が含まれているかわからない場合は、`request` オブジェクトの contentType フィールドを調べます。

要求が `JsonNode` オブジェクトとして返されたら、JSON API を使用してオブジェクトを反復処理し、フィールドを暗号化できます。

パラメーター

名前	タイプ	説明
なし		

返される内容

タイプ	説明
JsonNode	反復可能な `JsonNode` としての要求。

ParameterValue - getAsXmlContent()

要求の内容を `XMLContent` タイプの反復可能なオブジェクトとして返します。

このメソッドは エッジ暗号化 のルールでのみ使用できます。このメソッドでは、要求本体が有効な XML ペイロードであると仮定されます。contentType を確認してそのことを確かめることができます。

要求が `XMLContent` オブジェクトとして返されたら、XML API を使用してオブジェクトを反復処理し、フィールドを暗号化できます。

パラメーター

名前	タイプ	説明
なし		

返される内容

タイプ	説明
XMLContent	`XMLContent` タイプの反復可能なオブジェクトとしての要求。

ParameterValue - encodedQueryFor(String tableName)

要素の値が、指定したテーブルのエンコードされたクエリーであることを指定します。

パラメーターでこの関数を呼び出すと、パラメーターの値が、指定したテーブルのエンコードされたクエリ文字列 であることがプロキシに通知されます。プロキシは、エンコードされたクエリーを解析し、エンコードされたクエリーのフィールドのうち暗号化する必要があるフィールドを暗号化します。

パラメーター

名前	タイプ	説明
tableName	文字列	クエリーの実行を想定するテーブル。

返される内容

タイプ	説明
なし	

ParameterValue - valueFor(String tableName, String fieldName)

要素の値が、指定したテーブルの指定したフィールドにマップされることを指定します。

このメソッドを要素値で呼び出すと、この要素の値が、指定したテーブルの指定したフィールドにマップされることがプロキシに通知されます。プロキシは、フィールドを暗号化する必要があるかどうかを確認します。

パラメーター

名前	タイプ	説明
tableName	文字列	テーブル名。
fieldName	文字列	フィールド名

返される内容

タイプ	説明
なし	

XML API

XML API は、*request* オブジェクトか *ParameterValue* プロパティのどちらかで *getAsXmlContent()* を呼び出した後に使用できます。

XML API を使用して暗号化ルールを記述する際は、次の一般的な形式に従うことができます。

1. *getAsXmlContent()* は、*request* オブジェクトまたは *ParameterValue* プロパティで呼び出します。これは、*XMLContent* 基底クラスの反復可能なオブジェクトを返します。
2. *getIterator()* または *getIterator(String xPath)* は、*XMLContent* オブジェクトで呼び出します。これは、XML 要素の反復処理に使用できる *XMLElementIterator* オブジェクトを返します。
3. *hasNext()* メソッドは、*XMLElementIterator* オブジェクトで別の要素が使用可能かどうかを判断するために呼び出します。
4. *next()* は、*XMLElementIterator* オブジェクトで次の XML 要素を返すために呼び出します。*next()* を呼び出すには先に *hasNext()* を呼び出す必要があります。

5. `valueFor(String tableName, String fieldName)` は、XML 要素で呼び出します。このメソッドは、この要素の値が指定したテーブルの指定したフィールドにマップされることをプロキシに通知します。プロキシは、フィールドを暗号化する必要があるかどうかを確認します。

   注: XML 要素で `valueFor(String tableName, String fieldName)` を呼び出す必要があるかどうかを判断するために、`getName()` メソッドを使用して要素の名前を返すことができます。

インスタンス上の既知のテーブルフィールドへのマッピング

この例では、インスタンス上で XML ペイロードが処理され、インシデント テーブルにレコードが挿入されます。説明フィールドには、インシデントの short_description が設定されます。

```
<data>
   <record>
      <name>'Test Record 1'</name>
      <description>'Test Record 1 Description'</description>
      <tag>critical</tag>
   </record>
   <record>
      <name>'Test Record 2'</name>
      <description>'Test Record 2 Description'</description>
      <tag>security</tag>
   </record>
</data>
```

暗号化ルールの次のアクションを適用できます。

```
function sampleXmlAction1() {
   var xmlContent = request.getAsXmlContent();
   // This loop iterates over all description tags that match the given path
   var xmlElementIterator = xmlContent.getIterator('data/record/description');
   while (xmlElementIterator.hasNext()) {
      var xmlElement = xmlElementIterator.next();
      xmlElement.valueFor('incident', 'short_decription');
   }
}
```

このアクションは、**description** タグを反復処理し、値を暗号化してインスタンスの incident.short_description に挿入するようプロキシサーバーに指示します。

注: このルールは、XML ペイロードのすべての **record** タグの中で **description** タグを探します。暗号化するタグが 1 つしか出現しない場合でも、ルールは xPath と反復構造体を使用します。ただし、ループ内で 1 回しか反復しません。

インスタンス上の未知のテーブルフィールドへのマッピング

この例では、ルールは **record** タグを反復処理しますが、**record** タグの中でどのようなタグが存在するのかはわかりません。わかっているのは、**record** タグ内のタグが、テーブルの URL パラメーターで指定された列の名前と一致する、ということだけです。

また、テーブルがインシデントの場合、ルールは **description** タグのデータを暗号化し、インスタンスの short_description フィールドに格納します。

```
function sampleXmlAction2() {
   var xmlContent = request.getAsXmlContent();
```

```
    var tableName = request.urlParam.table;
    // This first iterator will iterate over all record elements
    var xmlElementIterator = xmlContent.getIterator('data/record');
    while (xmlElementIterator.hasNext()) {
        encryptFieldsInRecord(xmlElementIterator.next());
    }
}
function encryptFieldsInRecord(xmlElement) {
    //Then, iterate over all tags representing fields in the table
    var fieldIterator = xmlElement.getIteratorOverAllChildren();
    while (fieldIterator.hasNext()) {
        var field = fieldIterator.next();
        var fieldName = childElement.getName();
        //if table is incident, then description is encrypted for the short_description field
        if (tableName == 'incident' && fieldName == 'description') {
            field.valueFor(tableName, 'short_description');
        } else {
            //if table is not incident, ask the proxy to check if the given field is encrypted for the
 given table
            field.valueFor(tableName, fieldName);
        }
    }
}
```

`encryptFieldsInRecord()` 関数では、要求に基づいて動的に割り当てられるテーブルおよびフィールドで *`valueFor()`* メソッドを呼び出します。テーブル名とフィールド名は変化する可能性がありますが、その場合でもルールは、定義された暗号化設定に基づいてテーブル内のフィールドを暗号化する必要があるかどうかを確認するようプロキシに指示します。

フィールドが暗号化対象として設定されていない場合、またはタグがテーブルのフィールドと一致しない場合、プロキシはそのタグをスキップします。タグが暗号化対象としてマークされたフィールドに一致する場合、エッジ暗号化 プロキシサーバーは値を暗号化します。

エンコードされたクエリーの使用

この例では、すべてのタグに **filter** 属性があります。この属性は、エンコードされたクエリーがタグに含まれているかどうかを示します。

```
<data>
    <record>
        <name filter="false">'Test Record 1'</name>
        <description filter="false">'Test Record 1 Description'</description>
        <query filter="true">category=1^name=edge</query>
    </record>
    <record>
        <name filter="false">'Test Record 2'</name>
        <description filter="false">'Test Record 2 Description'</description>
        <query filter="true">category=2^severity=3</query>
    </record>
</data>
```

暗号化ルールの次のアクションを適用できます。

```
function sampleXmlAction3() {
    var xmlContent = request.getAsXmlContent();
    var tableName = request.urlParam.table;
    // This first iterator will iterate over all record elements
```

```
  var xmlElementIterator = xmlContent.getIterator('data/record');
  while (xmlElementIterator.hasNext()) {
     encryptFieldsInRecord(xmlElementIterator.next());
  }
}
function encryptFieldsInRecord(xmlElement) {
  //this time we want to iterate over all tags representing fields in the table
  var fieldIterator = xmlElement.getIteratorOverAllChildren();
  while (fieldIterator.hasNext()) {
     var field = fieldIterator.next();
     var fieldname = childElement.getName();
     //let's look at the filter attribute, if true, then encrypt as encoded query
     if (field.getAttributeValue('filter') == 'true') {
        field.encodedQueryFor(tableName);
     } else {
        //if it is false then check if the field should be encrypted
        field.valueFor(tableName, fieldName);
     }
  }
}
```

filter 属性の値が True の場合、ルールは、エンコードされたクエリーの値を暗号化するようプロキシサーバーに指示します。False の場合、ルールは、フィールドを暗号化する必要があるかどうかを確認するようプロキシに指示します。

XMLContent

XML コンテンツを反復処理するメソッドを提供するグローバル オブジェクトです。

`XMLContent` オブジェクトにアクセスするには、`request`オブジェクトで getAsXmlContent() を呼び出します。

POST または URL パラメーターの XML データにアクセスするには、request.postParams.<パラメーター名>.getAsXmlContent() または request.urlParams.<パラメーター名>.getAsXmlContent() を呼び出します。

XMLContent - getIterator()

XML コンテンツの `XMLElementIterator` オブジェクトを返します。

パラメーター

名前	タイプ	説明
なし		

返される内容

タイプ	説明
XMLElementIterator	`XMLContent` オブジェクトの要素の反復処理に使用できるオブジェクト。

XMLContent - getIterator(String xPath)

指定したパラメーターに基づいて、XML コンテンツの `XMLElementIterator` オブジェクトを返します。

パラメーター

名前	タイプ	説明
xPath	文字列	*XMLContent* オブジェクトのどこを出発点にするかを指定する xPath 風の式。

返される内容

タイプ	説明
XMLElementIterator	*XMLContent* オブジェクトの要素の反復処理に使用できるオブジェクト。

XMLElementIterator

XM L要素を反復処理するためのメソッドを提供します。

XMLElementIterator オブジェクトを取得するには、*XMLContent* クラスの *getIterator()* メソッドを呼び出します。

XMLElementIterator - hasNext()

利用可能な別の要素があるかどうかを判断します。

パラメーター

名前	タイプ	説明
なし		

返される内容

タイプ	説明
ブーリアン	別の要素が利用可能な場合は True。

XMLElementIterator - next()

反復子の次の要素を返します。

next() を呼び出すには先に *hasNext()* を呼び出す必要があります。

パラメーター

名前	タイプ	説明
なし		

返される内容

タイプ	説明
XMLElement	次の XML 要素。

XMLElement

XML 要素を反復処理して値をテーブルのフィールドにマップするためのメソッドを提供します。

XMLElement オブジェクトを取得するには、*XMLElementIterator* オブジェクトの *next()* メソッドを呼び出します。

XMLElement - getIterator(String xPath)

指定したパラメーターに基づいて、XML 要素の *XMLElementIterator* オブジェクトを返します。

パラメーター

名前	タイプ	説明
xPath	文字列	*XMLElement* オブジェクトのどこを出発点にするかを指定する xPath 風の式。

返される内容

タイプ	説明
XMLElementIterator	*XMLElement* オブジェクトの要素の反復処理に使用できるオブジェクト。

XMLElement - getIteratorOverAllChildren()

指定したパラメーターに基づいて、XML 要素のすべての下位要素が含まれている *XMLElementIterator* オブジェクトを返します。

パラメーター

名前	タイプ	説明
なし		

返される内容

タイプ	説明
XMLElementIterator	*XMLElement* オブジェクトの要素の反復処理に使用できるオブジェクト。

XMLElement - valueFor(String tableName, String fieldName)

要素の値が、指定したテーブルの指定したフィールドにマップされることを指定します。

このメソッドを要素値で呼び出すと、この要素の値が、指定したテーブルの指定したフィールドにマップされることがプロキシに通知されます。プロキシは、フィールドを暗号化する必要があるかどうかを確認します。テーブル名およびフィールド名がわからない場合は、要求に基づいて動的に割り当てられるテーブルおよびフィールドで、*valueFor()* 関数を呼び出すことができます。

パラメーター

名前	タイプ	説明
tableName	文字列	テーブル名。
fieldName	文字列	フィールド名

返される内容

タイプ	説明
なし	

XMLElement - encodedQueryFor(String tableName)

要素の値が、指定したテーブルのエンコードされたクエリーであることを指定します。

要素でこの関数を呼び出すと、要素の値が、指定したテーブルのエンコードされたクエリ文字列 であることがプロキシに通知されます。プロキシは、エンコードされたクエリーを解析し、エンコードされたクエリーのフィールドのうち暗号化する必要があるフィールドを暗号化します。

パラメーター

名前	タイプ	説明
tableName	文字列	クエリーの実行を想定するテーブル。

返される内容

タイプ	説明
なし	

XMLElement - getName()

要素名を返します。

パラメーター

名前	タイプ	説明
なし		

返される内容

タイプ	説明
文字列	要素名。

XMLElement - getAttributeValue(String attribute)

指定した属性の値を返します。

パラメーター

名前	タイプ	説明
attribute	文字列	属性名。

返される内容

タイプ	説明
文字列	属性値。

JSON API

JSON API は、`request` オブジェクトか `ParameterValue` プロパティのどちらかで `getAsJsonContent()` を呼び出した後に使用できます。

JSON API を使用して暗号化ルールを記述する際は、次の一般的な形式に従うことができます。

1. `getAsJsonContent()` は、`request` オブジェクトで呼び出します。これは、`JsonNode` 基底クラスの反復可能なオブジェクトを返します。
2. `iterator()` または `getIterator(String xPath)` は、`JsonNode` オブジェクトで呼び出します。これは、JSON オブジェクトのノードの反復処理に使用できる `JsonNodeIterator` オブジェクトを返します。
3. `hasNext()` メソッドは、`JsonNodeIterator` オブジェクトで別の要素が使用可能かどうかを判断するために呼び出します。
4. `next()` は、`JsonNodeIterator` オブジェクトで次の JSON 要素を返すために呼び出します。`next()` を呼び出すには先に `hasNext()` を呼び出す必要があります。
5. `valueFor(String tableName, String fieldName)` は、JSON 要素で呼び出します。このメソッドは、この要素の値が指定したテーブルの指定したフィールドにマップされることをプロキシに通知します。プロキシは、フィールドを暗号化する必要があるかどうかを確認します。

   注: JSON 要素で `valueFor(String tableName, String fieldName)` を呼び出す必要があるかどうかを判断するために、`getName()` メソッドを使用して要素の名前を返すことができます。

インスタンス上の既知のテーブルフィールドへのマッピング

この例では、インスタンス上で JSON ペイロードが処理され、インシデント テーブルにレコードが挿入されます。説明フィールドには、インシデントの short_description が設定されます。

```
{
   data: {
      records: [
         {
            "name": "Test Record 1",
            "description": "Test Record 1 Description",
            "tag": "security"
         },
         {
            "name": "Test Record 1",
            "description": "Test Record 1 Description",
            "tag": "security"
         }],
      "query": "assigned_to=3D4860165813e63a00d00abd322244b092^category=vulnerability"
   },
   "source": "10.11.13.14"
}
```

次のルールを適用できます。

```
function sampleJsonAction1() {
   var jsonContent = request.getAsJsonContent();
   // This loop iterates over all description elements in the records array
   var jsonNodeIterator = jsonContent.getIterator('/data/records/description');
   while (jsonNodeIterator.hasNext()) {
      var jsonNode = jsonNodeIterator.next();
      jsonNode.valueFor('incident', 'short_decription');
   }
}
```

このアクションは、**description** ノードを反復処理し、値を暗号化してインスタンスの incident.short_description に挿入するようプロキシサーバーに指示します。

注: このルールは、JSON ペイロード内のすべての **description** ノードを探します。暗号化するノードが 1 つしか出現しない場合でも、ルールは xPath と反復構造体を使用します。ただし、ループ内で 1 回しか反復しません。

インスタンス上の未知のテーブルフィールドへのマッピング

この例では、ルールは **records** を反復処理しますが、どのノードを想定するかはわかりません。わかっているのは、**records** 内の各オブジェクトについて、ノードがテーブルの URL パラメーターで指定された列の名前と一致する、ということだけです。

また、テーブルがインシデントの場合、ルールは **description** ノードのデータを暗号化し、インスタンスの short_description フィールドに格納します。

```
function sampleJsonAction2() {
   var jsonContent = request.getAsJsonContent();
   var tableName = request.urlParam.table;
   // This first iterator will iterate over all record elements
   var jsonNodeIterator = jsonContent.getIterator('data/records');
   while (jsonNodeIterator.hasNext()) {
         encryptFieldsInRecord(jsonNodeIterator.next());
   }
}
function encryptFieldsInRecord(jsonNode) {
   //this time we want to iterate over all nodes
   var fieldIterator = jsonNode.iterator();
   while (fieldIterator.hasNext()) {
         var field = fieldIterator.next();
         var fieldname = childElement.getName();
         if (fieldName == 'description') {
            field.valueFor(tableName, 'short_description');
         } else {
            field.valueFor(tableName, fieldName);
         }
   }
}
```

`encryptFieldsInRecord()` 関数では、要求に基づいて動的に割り当てられるテーブルおよびフィールドで *`valueFor()`* メソッドを呼び出します。テーブル名とフィールド名は変化する可能性がありますが、その場合でもルールは、定義された暗号化設定に基づいてテーブル内のフィールドを暗号化する必要があるかどうかを確認するようプロキシに指示します。

フィールドが暗号化対象として設定されていない場合、またはノードがテーブルのフィールドと一致しない場合、プロキシはそのノードをスキップします。ノード名が暗号化対象としてマークされたフィールドに一致する場合、プロキシは値を暗号化します。

エンコードされたクエリーの使用

```
function sampleJsonAction3() {
   var jsonContent = request.getAsJsonContent();
   var tableName = request.urlParam.table;
   // This first iterator will iterate over all record elements
   var jsonNodeIterator = jsonContent.getIterator('data');
   while (jsonNodeIterator.hasNext()) {
      var jsonNode = jsonNodeIterator.next();
      if (jsonNode.getName() == 'records')
         encryptRecors(jsonNodeIterator.next());
      else if (jsonNode.getName() == 'query')
```

```
        jsonNode.encodedQueryFor(tableName);
    }
}
function encryptRecords(jsonNode) {
    //we iterate over all fields in the node
    var recordIterator = jsonNode.iterator();
    while (recordIterator.hasNext()) {
        encryptFieldsInRecord(recordIterator.next());
    }
}
function encryptFieldsInRecord(jsonNode) {
    //this time we want to iterate over all nodes
    var fieldIterator = jsonNode.iterator();
    while (fieldIterator.hasNext()) {
        var field = fieldIterator.next();
        var fieldname = childElement.getName();
        field.valueFor(tableName, fieldName);
    }
}
```

この例では、ルールは **data** を反復処理します。**records** を見つけるたびに、ルールは、2 つ目の例と同じロジックを実行して、各ノードのフィールドを反復処理します。**query** ノードを見つけると、`encodedQueryFor()` を呼び出して、クエリーで暗号化する必要がある値を暗号化します。

JsonNode

JSON コンテンツを反復処理するメソッドを提供するグローバル オブジェクトです。

`JsonNode` オブジェクトにアクセスするには、`request`オブジェクトで getAsJsonContent() を呼び出します。

POST または URL パラメーターから JSON コンテンツにアクセスするには、request.postParms.<パラメーター名>.getAsJsonContent() または request.urlParms.<パラメーター名>.getAsJsonContent() を呼び出します。

JsonNode - getIterator(String xPath)

JSON コンテンツの `JsonNodeIterator` オブジェクトを返します。

このメソッドはルート ノードでのみ使用できますが、これを使用して JSON オブジェクトに深くトラバースできます。後続のトラバースでは `iterator()` メソッドを使用する必要があります。

パラメーター

名前	タイプ	説明
xPath	文字列	xPath 式。

返される内容

タイプ	説明
JsonNodeIterator	JSON オブジェクトのノードを反復処理できるオブジェクト。

JsonNode - iterator()

現在のノードのすべての子ノードを反復処理する `JsonNodeIterator` オブジェクトを返します。

パラメーター

名前	タイプ	説明
なし		

返される内容

タイプ	説明
JsonNodeIterator	JSON オブジェクトのノードを反復処理できるオブジェクト。

JsonNode - getAsString()

現在のノード値を文字列として返します。

パラメーター

名前	タイプ	説明
なし		

返される内容

タイプ	説明
文字列	現在のノード値。

JsonNode - getAsString(String propertyName)

指定したプロパティの文字列値を返します。

パラメーター

名前	タイプ	説明
propertyName	文字列	プロパティの名前。

返される内容

タイプ	説明
文字列	プロパティ値。

JsonNode - getName()

現在の JSON ノードの名前を返します。

パラメーター

名前	タイプ	説明
なし		

返される内容

タイプ	説明
文字列	現在の JSON ノードの名前。

JsonNode - valueFor(String tableName, String fieldName)

JSON プロパティが、指定したテーブルの指定したフィールドにマップされることを指定します。

このメソッドを JSON プロパティで呼び出すと、このプロパティの値が、指定したテーブルの指定したフィールドにマップされることがプロキシに通知されます。プロキシは、フィールドを暗号化する必要があるかどうかを決定します。テーブル名およびフィールド名がわからない場合は、要求に基づいて動的に割り当てられるテーブルおよびフィールドで、`valueFor()` 関数を呼び出すことができます。

パラメーター

名前	タイプ	説明
tableName	文字列	テーブル名。
fieldName	文字列	フィールド名

返される内容

タイプ	説明
なし	

JsonNode - encodedQueryFor(String tableName)

JSON プロパティの値が、指定したテーブルのエンコードされたクエリーであることを指定します。

JSON ノードでこの関数を呼び出すと、その値が、指定したテーブルのエンコードされたクエリ文字列 であることがプロキシに通知されます。プロキシは、エンコードされたクエリーを解析し、エンコードされたクエリーのフィールドのうち暗号化する必要があるフィールドの値を暗号化します。

パラメーター

名前	タイプ	説明
tableName	文字列	クエリーの実行を想定するテーブル。

返される内容

タイプ	説明
なし	

JsonNodeIterator

`JsonNodeIterator` オブジェクトを取得するには、`JsonNode` クラスの `getIterator()` メソッドまたは `iterator()` メソッドを呼び出します。

JsonNodeIterator - hasNext()

利用可能な別のプロパティがあるかどうかを判断します。

パラメーター

名前	タイプ	説明
なし		

返される内容

タイプ	説明
ブーリアン	別のプロパティが利用可能な場合は True。

JsonNodeIterator - next()

反復子の次のプロパティを返します。

`next()` を呼び出すには先に *`hasNext()`* を呼び出す必要があります。

パラメーター

名前	タイプ	説明
なし		

返される内容

タイプ	説明
JsonNode	次の *`JsonNode`*。

print(String message)

メッセージをラッパー ログ ファイル <プロキシサーバーのディレクトリー>/logs/wrapper_<日付>.log に出力します。

このメソッドは エッジ暗号化 のルール アクション スクリプトでのみ使用できます。

パラメーター

名前	タイプ	説明
message	文字列	ラッパー ログ ファイルに書き込むメッセージ。

返される内容

タイプ	説明
なし	

禁止されているキーワード

エッジ暗号化 プロキシは、ルールを保存する前に、暗号化ルールスクリプトを検証します。JavaScript のキーワードの多くは暗号化ルールスクリプトでは使用できません。

禁止されているキーワード

キーワード
__DIR__
__FILE__
__LINE__
__parent__
__proto__
エラー
eval
getClass
getPrototypeOf
Java
javax
javafx
JavaImporter
load
loadWithNewGlobal
新規
パッケージ
オブジェクト
prototype
RegExp
setPrototypeOf
this
throw

エッジ暗号化の辞書属性

テーブルとフィールドに辞書属性を追加して、エッジ暗号化による動作方法を制御します。

辞書属性を true に設定するには、[属性] フィールドに「attribute=true」と入力する必要があります。レコードに辞書属性を追加するには、「辞書属性を使用したテーブルとフィールドの変更」を参照してください。

エッジ暗号化除外 [edge_encryption_excluded]

フィールドを暗号化から除外するかどうかを決定します。

true に設定すると、フィールドまたはテーブルを暗号化できません。false に設定すると、フィールドを暗号化できます。

- 値：true/false
- ターゲット要素：フィールドまたはテーブル
- デフォルト値：false

エッジ暗号化除外 [edge_encryption_enabled]

フィールドを暗号化設定による暗号化の対象とするかどうかを決定します。

true に設定すると、フィールドは暗号化の対象になります。false に設定すると、フィールドは暗号化の対象外になります。この属性はシステムによって使用され、また変更できないため、ユーザーに表示されません。

> **注:** この属性は、フィールドが暗号化されていることを示すものではなく、フィールドで暗号化ロジックをトリガーするものでもありません。この属性によって、フィールドがユーザーによって暗号化される可能性が決まります。

- 値：true/false
- ターゲット要素：フィールド
- デフォルト値：文字列フィールドの場合は true

エッジ暗号化クリアテキスト許可済み [edge_encryption_clear_text_allowed]

プロキシサーバー経由で実行されるユーザー アクション、またはスケジュール済みジョブなどのサーバー側の自動スクリプトに対して、サーバー側スクリプトから、フィールド内の暗号化された文字列に、暗号化されていないデータを追加できます。

true に設定すると、データを追加できます。false に設定すると、データを追加できません。

- 値：true/false
- ターゲット要素：フィールド
- デフォルト値：false

ドメインセパレーションと エッジ暗号化

ドメインセパレーションは、エッジ暗号化 で制限された環境でサポートされます。エッジ暗号化は、エッジ暗号化 プロキシで定義された特定の設定、ルール、およびキーを使用してお客様の環境の中からデータを暗号化する機能を提供します。エッジ暗号化 プロキシはドメインに対応しておらず、ドメイン固有の設定をサポートすることができません。 ドメインセパレーションでは、データ、プロセス、および管理タスクをドメインと呼ばれる論理的なグループに分けることができます。どのユーザーがデータを表示できるか、データにアクセスできるかなど、このアプリケーションのいくつかの側面を制御できます。

サポートレベル: サポートなし

- ドメインフィールドがデータテーブル に存在している可能性がありますが、データを管理するロジックがありません。
- このレベルでは、ドメイン分割は考慮されません。

サポートレベルの詳細については、「アプリケーションでのドメインセパレーションのサポート)」を参照してください。

エッジ暗号化におけるドメインセパレーションの仕組み

エッジ暗号化は、ドメイン固有のキー、設定、およびルールが不要となる場合に使用できます。

関連トピック

サービスプロバイダーのドメインセパレーション

エッジ暗号化 とのデータの統合

エッジ暗号化 を使用してサードパーティのデータをインスタンスと統合するには、サポートされている統合を使用してデータを エッジ暗号化 プロキシサーバー経由でルーティングする必要があります。サポートされている統合では、各ペイロードのデータをテーブルのフィールドにマップするベース システムの暗号化ルールが使用されます。

暗号化対象としてマークされたフィールドにデータをアップロードする

エッジ暗号化 では、Excel、CSV、XML、またはその他のファイル タイプと、暗号化設定が定義されたフィールドとの間で、データのインポートまたはエクスポートを行うことはできません。

ODBC ドライバー

エッジ暗号化 プロキシサーバー経由で、ODBC ドライバーを使用して要求を暗号化し、データをクエリーします。

詳細を見る： エッジ暗号化と ODBC ドライバーの統合

MID サーバー

MID サーバー プロキシサーバー経由でデータをルーティングするように エッジ暗号化を設定できます。ただし、いくつかの制限が適用されます。

詳細を見る： エッジ暗号化と MID サーバーの統合

REST/SOAP Web サービス

REST/SOAP Web Services を使用し、エッジ暗号化 プロキシサーバー経由でレコード データを更新または取得します。

詳細を見る：Web サービス

JSONv2 Web サービス

JSONv2 Web サービス API を使用し、エッジ暗号化 プロキシサーバー経由でレコード データを更新または取得します。ベースシステムの暗号化ルールでは、データ取得 API やデータ変更 API をサポートしています。

- データ変更 API を使用して 1 つのレコードを挿入するには、`insert()` メソッドや `insertMutiple()` メソッドを使用します。
- データ変更 API を使用して複数のレコードを挿入するには、`insertMultiple()` メソッドを使用します。

詳細を見る：JSONv2 Web サービス

上記に記載されていないサードパーティのカスタムの統合から得られるデータを暗号化するには、カスタムの暗号化ルールを作成します。「カスタムの暗号化ルールを定義する」を参照してください。

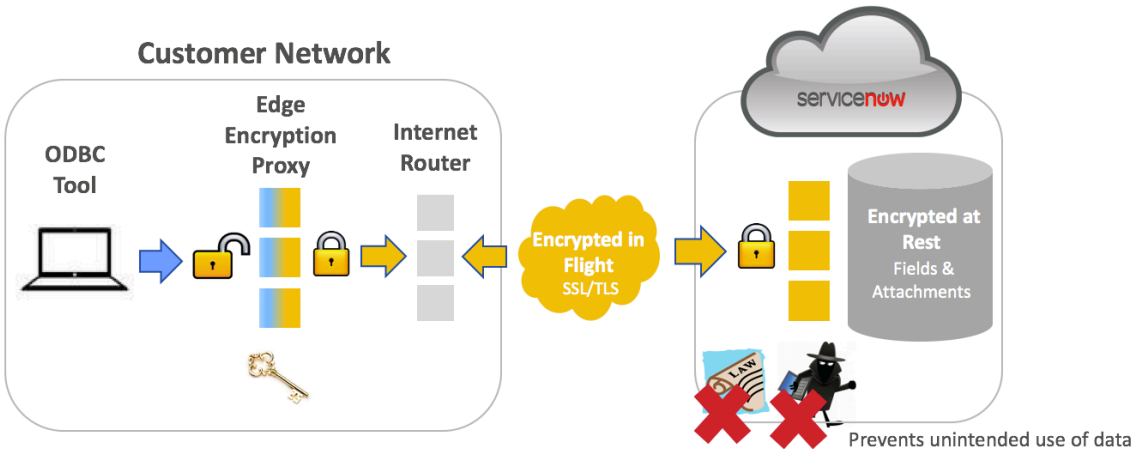
暗号化対象としてマークされたレコードに添付ファイルをアップロードする

REST および SOAP Web サービスを使用して設定した添付ファイル暗号化を使用して、テーブルに添付ファイルをアップロードできます。

エッジ暗号化と **ODBC** ドライバーの統合

エッジ暗号化 により暗号化されたデータをクエリーするように ODBC ドライバーを設定します。エッジ暗号化 が ODBC ドライバーと統合されているとき、ServiceNow プロキシサーバーは、エッジ暗号化 インスタンスへの ODBC ドライバーの要求を暗号化します。

インスタンスからの暗号化された応答は、エッジ暗号化 プロキシサーバーを通じて復号化されてから、ネットワークの ODBC ドライバーに渡されます。



統合を正常に機能させるためには、ODBC ドライバーが エッジ暗号化 プロキシサーバー証明書を信頼する必要があります。エッジ暗号化 プロキシサーバー証明書が、ODBC ドライバーが信頼している認証局によって署名されている場合、エッジ暗号化 プロキシサーバーは自動的に信頼されます。ただし、ODBC ドライバーが信頼している認証局がまだ エッジ暗号化 プロキシサーバー証明書に署名していない場合は、自己署名証明書を ODBC トラストストアにインポートする必要があります。

自己署名証明書を **ODBC** トラストストアにインポートする

ODBC ドライバーが信頼している認証局がまだ エッジ暗号化 プロキシサーバー証明書に署名していない場合は、自己署名証明書を ODBC トラストストアにインポートする必要があります。エッジ暗号化 プロキシサーバーから証明書をエクスポートし、ODBC トラストストアにインポートすることができます。

始める前に
必要なロール：admin

ODBC ドライバーが信頼している認証局が エッジ暗号化 プロキシサーバー証明書に署名したかどうかを判断するには、プロキシのホーム ディレクトリーのキーストア ディレクトリーで次のコマンドを実行して、ODBC ドライバーが信頼している認証局の一覧を表示します。

```
keytool -keystore "<ODBC directory>\ip\Java\jre\lib\security\cacerts" -list
```

注:

ほとんどの場合、クライアントはロードバランサーを介して エッジ暗号化 プロキシに接続するため、証明書はロードバランサーで構成された証明書になります。

クライアントと Edge プロキシサーバーの間にロードバランサーがない場合、証明書は Edge プロキシによって提示される証明書になり、*edgeencryption.properties* ファイルで構成されます。

```
edgeencryption.proxy.https.port = 9090
edgeencryption.proxy.https.keystore.path = keystore/keystore
edgeencryption.proxy.https.keystore.password = password
edgeencryption.proxy.https.cert.alias = jetty
```

プロパティの編集の詳細については、「エッジ暗号化プロパティファイルでその他のプロパティを設定する」を参照してください。

手順

1. プロキシのホーム ディレクトリーのキーストア ディレクトリーに移動します。
2. キーストアに自己署名証明書がないか確認します。
   a. キーストアに証明書がないかを確認するには、次のコマンドを実行してキーストア内のすべてのアイテムを一覧表示します。

      ```
      keytool -list -keystore keystore.jceks -storetype jceks -v
      ```

   b. アイテムのリストでキー エイリアスを探します。
3. キー エイリアスを使用して、証明書を .cer ファイルにエクスポートします。

   ```
   keytool -export -alias <key alias> -keystore keystore.jceks -storetype jceks -rfc -file <file name>.cer
   ```

4. ODBC トラストストア ディレクトリー ODBC\ip\Java\jre\lib\security\cacerts に移動します。
5. 証明書を ODBC トラストストアにインポートします。

   ```
   keytool -keystore cacerts -importcert -alias $<key alias> -file <file name>.cer
   ```

ODBC ドライバーのプロパティを設定する

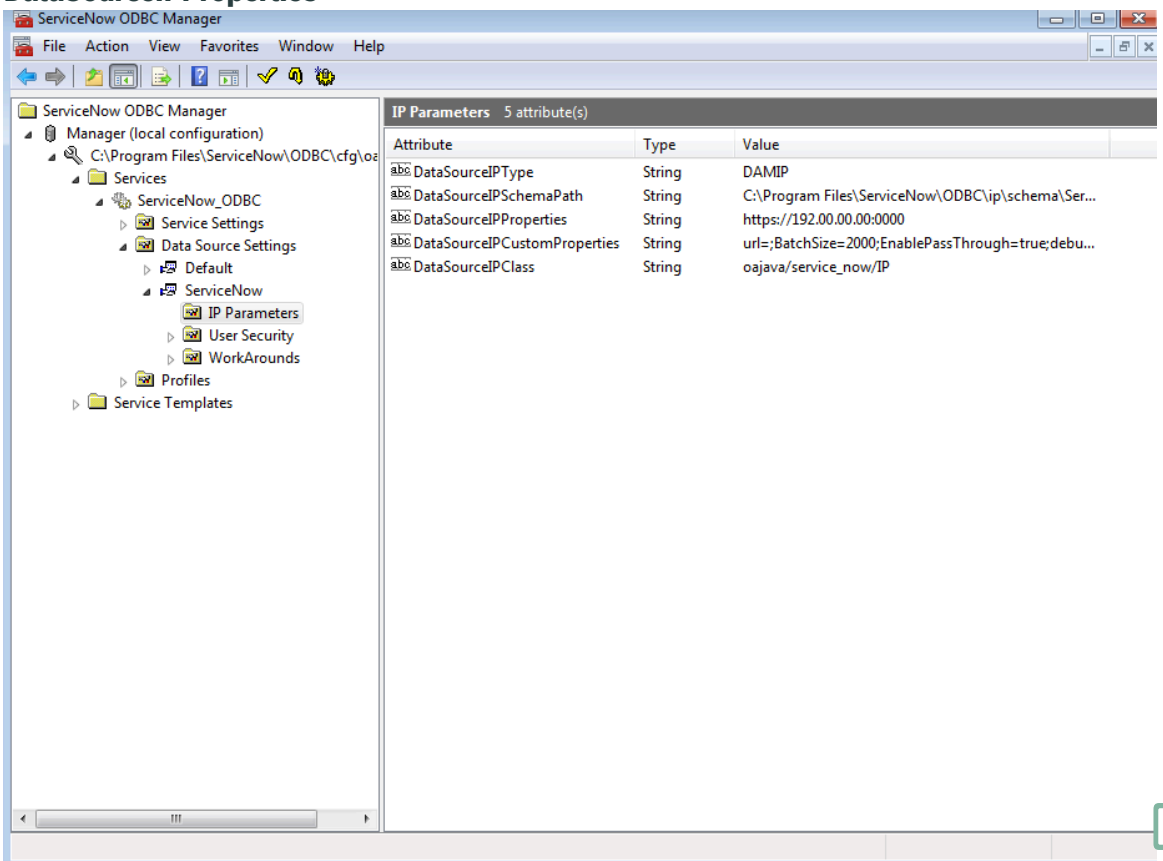
要求が エッジ暗号化 プロキシサーバー経由でルーティングされるように、ODBC ドライバーのプロパティを設定します。

始める前に
必要なロール：admin

手順

1. Windows で、次の場所に移動します: 開始 > プログラム > **ServiceNow ODBC** 管理コンソール.
2. コンソール ツリーのルートを ServiceNow ODBC Manager\Manager\<インストール場所>\Services\ServiceNow_ODBC\Data Source Settings\ServiceNow\IP Parameters まで展開します。
3. 属性をダブルクリックします。
4. [値] を、エッジ暗号化プロキシサーバーの URL (https://<IP アドレス>:<ポート> など) に変更します。

DataSourceIPProperties



5. **[OK]** をクリックします。

次のタスク
これで、要求が エッジ暗号化 プロキシサーバー経由でインスタンスにルーティングされるように、ODBC ドライバーが設定されます。

エッジ暗号化と **MID** サーバーの統合

MID サーバー プロキシサーバー経由でデータをルーティングするように エッジ暗号化 を設定します。

MID サーバー と統合する場合、エッジ暗号化 プロキシサーバーは MID サーバーのエンドポイントとして機能します。これで、エッジ暗号化 プロキシサーバーは、ServiceNow インスタンスと MID サーバー間で受け渡しされるデータの暗号化と復号化を行います。

MID サーバーと統合する際の制限事項

MID サーバー プロキシサーバー経由で渡されるように エッジ暗号化のデータを設定する場合、次の制限が適用されます。

- [ECC キュー] フィールド暗号化はサポートされていません。
- 暗号化されたデータをディスカバリーまたはサービスマッピングで使用することはできません。

MID サーバーからエッジ暗号化プロキシサーバーを指し示す

データを MID サーバーから エッジ暗号化 プロキシサーバー経由で渡すには、MID サーバーから エッジ暗号化 プロキシサーバーを指し示すように、MID サーバーの設定ファイルを更新します。

始める前に
必要なロール：admin

このタスクについて
MID サーバー プロキシサーバー経由でデータを渡すように エッジ暗号化を設定する場合、MID サーバーの設定ファイルで Web プロキシ プロパティを使用して、トラフィックを エッジ暗号化 プロキシサーバー経由でインスタンスにルーティングすることはできません。代わりに、エッジ暗号化 プロキシサーバーを MID サーバーのエンドポイントとして設定する必要があります。

手順

1. ローカルの MID サーバー ディレクトリーに移動し、config.xml ファイルを開きます。
2. 要素 `<parameter name="url" value="https://YOUR_INSTANCE.service-now.com" />` を探し、値プロパティを エッジ暗号化 プロキシサーバーの URL に変更します。
   たとえば、`http://hostname.mycompany.com:8081` などです。
   この手順により、MID サーバー に対して、トラフィックをインスタンスにではなく エッジ暗号化 プロキシサーバーに渡すよう指示します。トラフィックを受信した エッジ暗号化 プロキシサーバーは、必要なフィールドをすべて暗号化し、ペイロードをインスタンスに渡します。
3. ファイルを保存して閉じます。
4. MID サーバーが実行中の場合は再起動します。

エッジ暗号化の診断とパフォーマンス

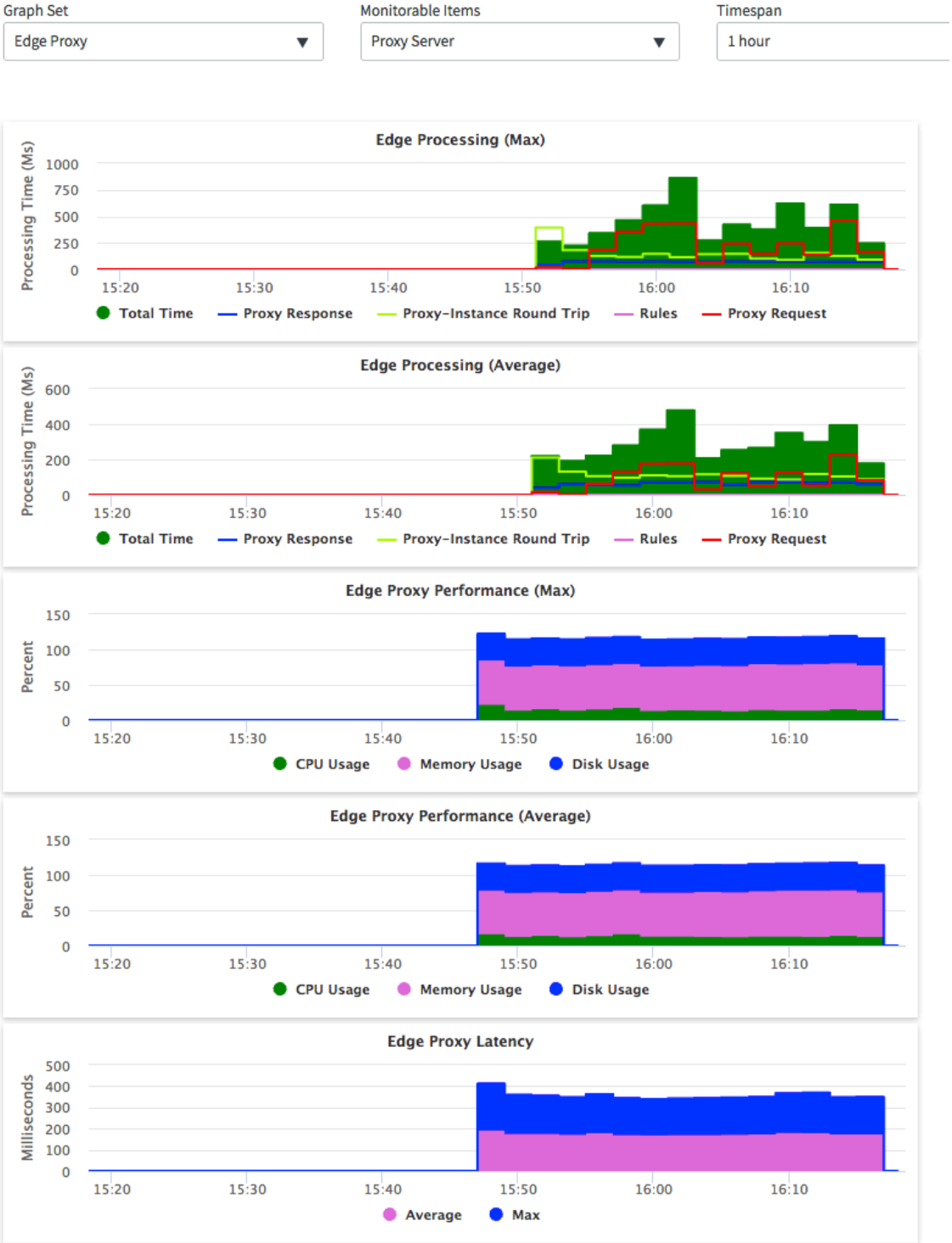
エッジ暗号化 プロキシサーバーのパフォーマンスの傾向を監視し、エッジ暗号化 プロキシサーバーにより生成されるエラーを詳しく調べます。

Edge プロキシのパフォーマンス

エッジ暗号化 パフォーマンスのホームページで設定されている Edge プロキシ グラフを使用して、ServiceNow プロキシサーバーの主要なパフォーマンスの傾向を表示します。次の傾向を監視します。

- クライアント、プロキシサーバー、およびインスタンスの間の最大および平均の応答時間。
- ホストマシンの CPU、ディスク領域、およびメモリの使用率。
- プロキシサーバーと ServiceNow インスタンス間の最大および平均のネットワークレイテンシー時間。

注: エッジ暗号化 プロキシサーバーのうち名前が重複しているものについては、パフォーマンスの傾向は報告されません。



Edge の処理 (最大および平均)

要求を処理するための最大および平均時間 (ミリ秒単位)。これらのデータ ポイントは、一般的な経時的傾向です。

- 合計時間：プロキシサーバーがクライアントから要求を受信して応答を送信するまでの時間。このデータ ポイントは、後続のデータ ポイントの合計です。
- プロキシの応答：プロキシサーバーがインスタンスからの応答を処理する時間。

- プロキシインスタンスの往復：プロキシサーバーがインスタンスに要求を送信して応答を受信するまでの時間。プロキシサーバーとインスタンス間のネットワーク レイテンシーと、インスタンスが要求を処理するために費やした時間が含まれます。
- ルール：プロキシサーバーが定義済みの暗号化ルールを使用して要求を評価する時間。
- プロキシの要求：プロキシサーバーがクライアント要求を処理してインスタンスに転送するまでの時間。

Edge プロキシのパフォーマンス (最大および平均)

ホスト マシンで使用されるリソースの最大および平均のパーセンテージ。

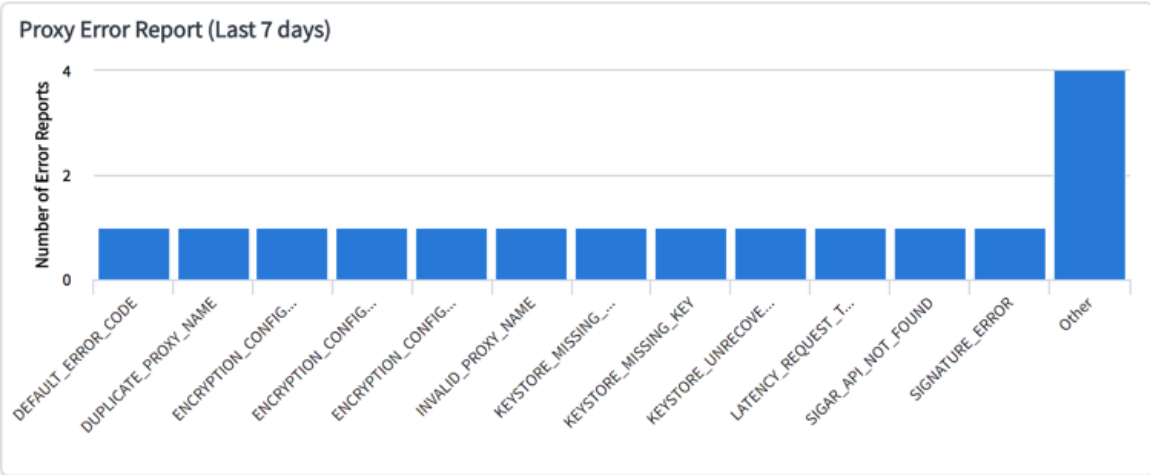
- CPU 使用率
- メモリ使用率
- ディスク使用率

Edge プロキシのレイテンシー

特定の時点における最大および平均のネットワーク レイテンシー (ミリ秒単位)。レイテンシーは、プロキシサーバーが単純な ping をインスタンスに送信して応答を受信するまでの往復時間によって決まります。

プロキシー エラー レポート

移動先 エッジ暗号化の設定 > 診断とトラブルシューティング > プロキシー エラー レポート 過去 7 日間に収集されたすべてのプロキシサーバーエラーを表示する。



エラーは 1 分間隔で収集されます。エラー レポートは毎分生成されます。縦軸には、過去 7 日間の各エラーを含むエラー レポートの数が表示されます。たとえば、1 分のレポート期間に DEFAULT_ERROR_CODE エラーが複数回スローされた場合でも、DEFAULT_ERROR_CODE のバーには [エラー レポートの数] 軸上で 1 とのみ表示されます。

この画面から次の操作が行えます。

- プロキシ エラー コードの各バーをクリックすると、プロキシサーバーごとの単一のエラーに関するレポートが表示されます。この画面からバーを再度クリックすると、エッジ暗号化 プロキシ統計情報テーブル [edge_encryption_stat] にエラー テキストを表示できます。エラー テキスト内のリンクをクリックすると、エラーの詳細と修正手順の候補を表示できます。
- [その他] をクリックすると、エラー レポートの 2 ページ目が表示されます。

注: 同じ名前のプロキシサーバーが複数ある場合、プロキシ エラー レポートには 1 つの DUPLICATE_PROXY_NAME エラーが表示されます。名前が重複するプロキシサーバーについては、他のエラーは報告されません。このエラーが発生した場合は、すべてのプロキシサーバーに一意の名前が付いていることを確認してください。

その他のモニタリング リソース

インスタンスは、すべての暗号化プロキシを追跡します。エッジ暗号化 プロキシサーバーはそれぞれ、起動時に登録されます。次の場合にインスタンスに通知が送られます。

- 新しい エッジ暗号化 プロキシサーバーが起動したとき。
- エッジ暗号化 プロキシサーバーが意図的にシャット ダウンされたとき。

エッジ暗号化 プロキシサーバーが、エッジ暗号化 がインストールされていないインスタンスを登録しようとした場合、プロキシは起動しません。

すべての暗号化設定ファイルが監査されます。削除済みレコードは、すべての暗号化設定ファイルで監査されます。監査レコードは sys_audit テーブルに格納されます。特定の設定レコードの履歴を表示するには、レコードを表示して、 履歴 > リスト メニューにあります。一括暗号化ジョブは監査されません。

さらに次のリソースを使用してプロキシサーバーを監視します。

テーブル	説明
無効な挿入試行 [sys_edge_encryption_invalid_insert_log]	次のデータを暗号化されたフィールドに保存しようとした試みのリスト。 • 暗号化されていないデータ。 • エッジ暗号化 プロキシから送られていないデータ。 この場合、インスタンスは、このデータを保存するすべての試行を拒否してログに記録します。security-admin ロールを持っている場合は、「無効な挿入試行」リストでログを表示できます。
ジョブの失敗 [sys_encryption_job_execution]	正常に実行されなかったジョブのリスト。
システムログ	インスタンスは、登録された各プロキシサーバーからのメッセージの有無を定期的にチェックします。プロキシサーバーから必要な期間内にメッセージが送信されなかった場合は、エラーがログに記録されます。ログ メッセージには、暗号化プロキシの情報と、プロキシが最後にインスタンスに ping を実行した時刻が記録されます。オンラインになっている暗号化プロキシがないとインスタンスが判断した場合は、メッセージがログに記録されます。これらのメッセージはシステムログに追加されます。

Edge プロキシの統計情報の収集を無効化または縮小する

エッジ暗号化 プロキシサーバーから Edge プロキシ グラフ セットの統計情報を ServiceNow パフォーマンスのホームページに送信しないようにするか、統計情報の収集頻度を減らします。

始める前に
必要なロール：admin または security_admin

このタスクについて
edgeencryption.properties 設定ファイルにプロパティを追加することで、次の操作が行えます。

- Edge プロキシ グラフ セットを無効にします。
- エッジ暗号化 プロキシサーバーによる統計情報の収集間隔を変更します。デフォルトでは、統計情報は 30 秒ごとに収集されます。

手順

1. プロキシサーバーのインストール ディレクトリーで、<インストール ディレクトリー>/conf/ フォルダーにある edgeencryption.properties 設定ファイルを開きます。
2. エッジ暗号化プロキシサーバーのプロパティ のいずれかを追加します。
3. プロキシサーバーを再起動します。

エッジ暗号化 プロキシのデバッグログ記録を増やす

ログ記録のレベルを上げて、ログを解釈し、プロキシの問題をデバッグします。

現在、エッジ暗号化 プロキシのデバッグログを増やすための、3 つのオプションがあります。問題をデバッグするためにログ記録のレベルを上げることで、問題を調査するためのより詳細なログステートメントでの情報をテクニカルサポートに提供できます。

デバッグしている問題に応じて、次の 3 つの方法のいずれかでデバッグログを設定します。

- SSL 接続以外の問題のデバッグ
- プロキシを介した要求のタイミングメトリクスのログ記録
- エッジ暗号化プロキシとインスタンス間の SSL 接続に関する問題のデバッグ

すべてのデバッグケースについて、自分でログを表示して解釈するか、インシデントを開いて ServiceNow テクニカルサポートから問題の説明とその再現方法について解釈を得ることができます。

SSL 接続以外の エッジ暗号化 アプリケーションに関する問題のデバッグ

この方法を使用して、プロキシを停止して再起動することなく、エッジ暗号化 アプリケーションの問題をデバッグします。これらの手順により、ログ記録レベルが向上し、詳細なログステートメントを使用して根本原因のトラブルシューティングを行うことができます。

始める前に
必要なロール：admin

> **注:** $proxy_installation_location/conf/log4j2.properties ファイルに加えられた変更は、変更を行ってから約 60 秒以内にプロキシによって取り込まれます。プロキシを再起動する必要はありません。

手順

1. $proxy_installation_location/conf/log4j2.properties ファイルで、次の行を見つけます。

```
logger.edge.level=info
```

2. 上記の行を次のように変更します。

```
logger.edge.level=debug
```

3. 変更を保存します。
   変更が有効になるまでに最大 60 秒かかる場合がありますが、プロキシを再起動する必要はありません。
4. 問題を再現します。
5. $proxy_installation_location/logs/edgeencryption.log ファイルでアプリケーションに関連するデバッグログステートメントを確認します。

結果
プロパティを変更した後に、$proxy_installation_location/logs/edgeencryption.log ファイルで追加の詳細を確認できます。デバッグが終了したら、$proxy_installation_location/conf/log4j2.properties ファイルに加えた変更を元に戻します。

プロキシを介した要求のタイミングメトリクスのログ記録

タイミングメトリクスのログ記録を有効にして、エッジ暗号化 プロキシによって処理される各要求のメトリクスステートメントを追加します。これらの各タイミングメトリクスログステートメントには、処理時間や使用された暗号化ルールなど、要求に関する有用な情報が含まれています。

始める前に
必要なロール：admin

注:

追加のログ記録設定が $proxy_installation_location/conf/log4j2.properties ファイルに追加されます。加えられた変更は、ファイルの変更が行われてから約 1 分以内に動的に取り込まれるため、プロキシを再起動する必要はありません。

手順

1. ファイルの末尾に次の行を追加して、$proxy_installation_location/conf/log4j2.properties ファイルを変更します。

```
appender.timinglog.type=RollingFile
appender.timinglog.name=TimingLog
appender.timinglog.fileName=../logs/edgenetwork.log
appender.timinglog.filePattern=../logs/$${date:yyyy-MM}/edgenetwork-%d{yyyy-MM-dd-HH}-%i.log.gz
appender.timinglog.layout.type=PatternLayout
appender.timinglog.layout.pattern=%d [%t] %-5p %m%n
appender.timinglog.policies.type=Policies
appender.timinglog.policies.size.type=SizeBasedTriggeringPolicy
appender.timinglog.policies.size.size=500MB
appender.timinglog.strategy.type=DefaultRolloverStrategy
appender.timinglog.strategy.max=4

logger.timing.name=com.snc.edgeencryption.metrics.EdgeEncryptionTimingMetricCache
logger.timing.level=debug
```

```
logger.timing.additivity=false
logger.timing.appenderRef.rolling.ref=TimingLog
```

2. ファイルを保存します。

結果
log4j.properties ファイルが保存されると、$proxy_installation_location/logs/edgenetwork.log ログファイルにネットワーク時間に関する次のタイプのメッセージが表示されます。

```
2022-07-21 12:56:15,783 [qtp1971991758-7700] DEBUG
 com.snc.edgeencryption.metrics.EdgeEncryptionTimingMetricCache -
 request_uri=/api/now/ui/presencesysparm_auto_request=true&cd=1658433375754
 request_method=POST client_request_received="2022-07-21 12:56:15,015"
 proxy_request_processing_time=6 all_rules_processing_time=0
 rule_executed="REST JSON" rule_execution_time=1 proxy_instance_round_trip=14
 proxy_response_processing_time=1 total_time_from_proxy=21 reponse_code=201
 glide_user=SCv3_1:BAz1ZK7ee9XoroG2nvMlixHpgTvsT4fY2bwQvnH2WdU=:y5HGsTTqo3Pjq6
G0xk4LoazCwCiWRJk4/6SpbXuBzqg=:6816f79cc0a8016401c5a33be04be441
 jsessionid_suffix=037A66
```

ログメッセージの値は次のとおりです。

```
request_uri: The URI being requested

request_method: The HTTP method being used, for example, GET, POST, PUT, PATCH,
 DELETE

client_request_received: The timestamp noting when the HTTP client request arrived at the
 Edge proxy

proxy_request_processing_time: How long the Edge proxy took to process the request in
 milliseconds

all_rules_processing_time: Total time it took to execute all of the Edge Encryption rules for the
 request in milliseconds

rule_executed: The name of the encryption rule that was executed

rule_execution_time: How long it took to execute listed rule_executed in milliseconds

proxy_instance_round_trip: The time from when the Edge proxy sent the request to the instance
 until the instance sent the response and was received by the edge proxy in milliseconds

proxy_response_processing_time: How long the Edge proxy took to process the response in
 milliseconds

total_time_from_proxy: The total time from when the Edge proxy received the request from the
 client and returned the response to the client in milliseconds

response_code: HTTP response code

glide_user: The glide_user cookie value

jsessionid_suffix: The JSession cookie suffix associated with the request
```

エッジ暗号化 プロキシとインスタンス間の **SSL** 接続に関する問題のデバッグ

この方法を使用して、エッジ暗号化 プロキシとインスタンスの間の SSL 接続に関する問題 (プロキシ経由のインスタンスへのアクセスが失敗するなど) をデバッグします。これらの手順により、ログ記録が増加し、詳細なログステートメントを見つけることができます。

始める前に
必要なロール：admin

注: SSL 接続のデバッグは、TLS 接続タイプの問題をトラブルシューティングする場合にのみ関連します。実際には、これは一般的ではなく、めったに必要ありません。

手順

1. プロキシサーバーを停止します。
2. $proxy_installation_location/conf/wrapper.confファイル (Java スタートアッププロパティ) に次の行を追加します。

   ```
   wrapper.java.additional.<next available number in sequence> = -Djavax.net.debug=all
   ```

   例：

   ```
   For example: wrapper.java.additional.4 = -Djavax.net.debug=all
   ```

3. 変更を保存し、プロキシサーバーを再起動します。
4. 接続の問題を再現します。

結果
問題を再現した後、SSL 交換に関連するデバッグログステートメントを $proxy_installation_location/logs/wrapper_<current date>.log ファイルで見つけることができます。デバッグが終了したとき、前の手順で作成された行を削除するかコメントアウトすることで、追加のログ記録をリモートで実行できます。

データベース暗号化

ServiceNow® は、すべてのデータが停止している時に保護が必要な可能性のあるデータ保護の法的義務を負う顧客向けに、データベース暗号化 (DBE) とフルディスク暗号化方法を提供します。

重要: Washington DC リリース以降、データベース暗号化が将来の廃止に備えて準備されます。クラウド暗号化は、静止データ暗号化の代替ソリューションです。詳細については、「キー管理を使用したクラウド暗号化」を参照してください。

探索	要求	参照
		
データベース暗号化の主な機能とビジネス価値について説明します。	データベースキーのローテーションを要求する方法の詳細をご覧ください。	カスタマー制御スイッチを使用したデータベースの暗号化 (DBE-CCS) は、すべての保存データをデータベースで使用されていないときに暗号化する暗号化ソリューションです。

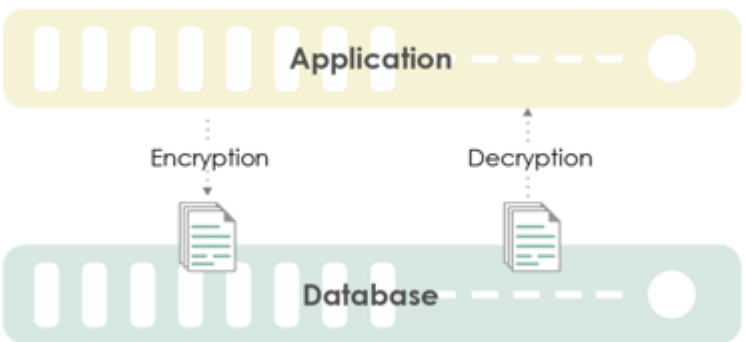
データベース暗号化の詳細

ServiceNow® は、すべてのデータが停止している時に保護が必要な可能性のあるデータ保護の法的義務を負う顧客向けに、データベース暗号化 (DBE) とフルディスク暗号化方法を提供します。

重要: Washington DC リリース以降、データベース暗号化が将来の廃止に備えて準備されます。クラウド暗号化は、静止データ暗号化の代替ソリューションです。詳細については、「キー管理を使用したクラウド暗号化」を参照してください。

データベース暗号化を使用すると、データベースがオンラインかオフラインかに関わらず、すべてのデータを対称 AES-256 暗号化で保護できます。ServiceNow AI Platform の観点から、すべてのデータが復号化されます。

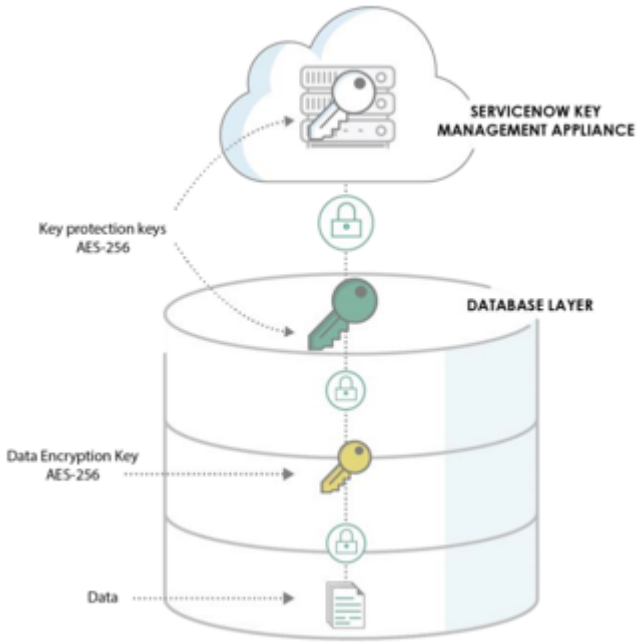
- データベース暗号化では、リアルタイムに保存データの暗号化をサポートし、機能を損なうことなくオンラインとオフラインのデータを保護します。
- フルディスク暗号化は、ディスクの紛失や盗難が発生した場合にオフラインデータを保護します。



データベース暗号化を使用すると、保存されているすべてのデータが暗号化され、個々のレコードまたはテーブルがアクセス時に復号化されます。新規または変更されたデータは、テーブルに入力されるときに暗号化され、関連するアクティビティログファイル (bin、redo、undo、error) も暗号化されます。

データベース暗号化はユーザーに対して透過的であり、機能が失われることはありません。この機能を使用すると、すべてのインスタンスがレプリケーショントラフィックとバックアップとともに暗号化されます。インスタンスのクローンは引き続き使用できますが、データベース暗号化を使用した場合、パフォーマンスに最大 5% の軽微な影響があります。サポートされているリリースの ServiceNow AI Platform の新規インスタンスと既存のインスタンスの両方で、データベース暗号化を利用できます。

図に示すように、ServiceNow は次の 3 つのレベルのキー階層を使用してキーを格納し、管理します。



1. 顧客固有の AES-256 キーがデータベースエンジンで作成され、データの暗号化に使用されます。
2. 2 つ目の顧客固有の AES-256 キーがデータベースエンジンで作成され、第 1 レベルのキーの保護に使用されます。
3. 3 つ目の AES-256 キーが作成され、ServiceNow データセンターの FIPS 140 検証済みキー管理アプライアンス内に保存されます。このキーは第 2 レベルのキーを保護し、顧客インスタンスごとに一意です。

ServiceNow AI Platform は、顧客指定のスイッチである DBE-CCS を使用したデータベース暗号化もサポートしています。これは、データベースで使用されていないすべてのデータを暗号化する暗号化ソリューションです。機能に影響することなく、業界標準の AES 暗号化を使用します。データはディスクに書き込まれるときに暗号化され、ディスクから読み取られるときに復号化されます。したがって、アプリケーションは常にデータを暗号化されていない状態で保持し、影響を受けることなく必要なロジックと機能を実行することができます。

注: データベース暗号化は、オンプレミスインスタンスではサポートされていません。

データベースの暗号化に独自のキーを使用する場合は、「カスタマー制御スイッチを使用したデータベースの暗号化」を参照してください。

データベースキーのローテーションの要求

年に 1 回または必要に応じて、サポートに要求を送信して、データベースキーをローテーションします。

始める前に
必要なロール：admin

重要: Washington DC リリース以降、データベース暗号化が将来の廃止に備えて準備されます。クラウド暗号化は、静止データ暗号化の代替ソリューションです。詳細については、「キー管理を使用したクラウド暗号化」を参照してください。

このタスクについて
キーローテーションは、有効期限前 24 時間以内の夜間に行われ、インスタンスのサービスは中断されません。

注: 現在、キーローテーションは ServiceNow Commercial、Government Community Cloud (GCC)、France、および Singapore の環境でのみ利用できます。

手順

カスタマーサービス & サポートに連絡して、次のキーローテーションアクションを要求してください。

- 登録すると、指定したすべてのインスタンスで年に 1 回キーローテーションが実行されます。
- 次の情報を含む、過去 3 回のキーローテーションの履歴レポートを取得できます。
  - インスタンス名
  - キー名とバージョン
  - ローテーションの日付と時刻
- 年に 1 回スケジュールされているローテーション以外に、早期のキーローテーションをスケジュールします。

カスタマー制御スイッチを使用したデータベースの暗号化

カスタマー制御スイッチを使用したデータベースの暗号化 (DBE-CCS) は、すべての保存データをデータベースで使用されていないときに暗号化する暗号化ソリューションです。

重要:

カスタマー制御スイッチを使用したデータベースの暗号化のサポート終了プロセスが開始され、Yokohama リリースの時点で販売終了と更新終了のマイルストーンに達しています。保存データの暗号化のサポートについては、「 キー管理を使用したクラウド暗号化」を参照してください。

概要

カスタマー制御スイッチを使用したデータベースの暗号化では、機能に影響を与えることなく、業界標準の AES 暗号化を使用します。データはディスクに書き込まれるときに暗号化され、ディスクから読み取られるときに復号化されます。アプリケーションは常にデータを暗号化されていない状態で保持し、必要なロジックと機能を実行することができます。

DBE-CCS は、テーブルスペース暗号化または透過的データ暗号化と呼ばれるデータベース固有の技術を使用します。この技術の詳細については、MariaDB Web サイト の「テーブルスペース暗号化」を参照してください。

DBE-CCS では、ServiceNow インスタンスに定期的に秘密キーを提供する HTTPS REST サービスエンドポイントを設定する必要があります。CCS エンドポイントは、データベースインスタンスの公開鍵で暗号化された秘密キーを顧客に返します。

顧客エンドポイント

重要: 組織は、CCS エンドポイントの設定と管理に対して単独で責任を負います。顧客エンドポイントの仕様は KB0789788 に記載されています。

ServiceNow テクノロジーパートナーの Fortanix が、顧客エンドポイントの実装をお手伝いします。統合の詳細については、テクノロジーパートナーにお問い合わせください。詳細については、「ServiceNow での Fortanix データセキュリティマネージャーの使用」を参照してください。

複数 **ServiceNow** バージョンのサポート

重要: データベースの暗号化は、リリースに依存しない有料のインフラストラクチャ製品です。サポートされているリリースと新しいまたは既存のインスタンスに適用することができます。

その他の参考資料

DBE-CCS の詳細については、次の参考資料を参照してください。

参考資料	説明
KB0993681	データベース暗号化のカスタマー制御スイッチの構造
KB0789788	DBE-CCS の実装ガイド

注: KB 記事にアクセスするには、まず Now Support への認証を行う必要があります。

アクセス管理

アクセス管理を使用して ServiceNow® インスタンスに安全にアクセスできます。

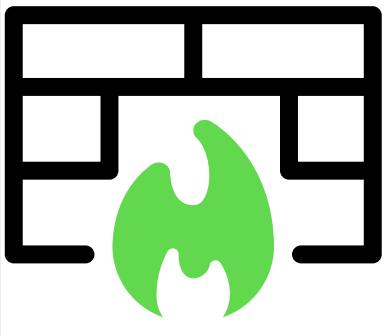
Zero Trust アクセス

Zero Trust アクセスでは、ユーザーの ID 検証とリスクアセスメントの後にのみ、アプリケーションとデータへのすべてのアクセスが最小限の権限で付与されます。

ドメインセパレーション

ServiceNow AI Platformにより、サービスプロバイダー (SP) は顧客に迅速なオンボーディングを提供し、コンプライアンスを満たし、ドメインセパレーションを使用してデータを保護できます。

認証

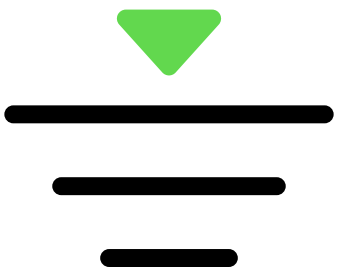


ServiceNow の認証は、インスタンスにアクセスするユーザーの ID を検証し、ユーザーのロールまたは職務に一致する機能に対してユーザーを許可します。

ACL

アクセス制御リスト (ACL) のルールが、ユーザーがデータと対話する前に一連の要件を満たすことを要求してデータへのアクセスを制限します。

データフィルタリング



読み込みクエリを実行するときに、データフィルタリングを使用し、対象の属性に基づいてテーブルとレコードへのアクセスを制御します。

セキュリティロール

自動翻訳

		セキュリティロールによりセキュリティを強化するためには、ユーザー全員に少なくとも 1 つのロールを割り当てて、インスタンスが内部ユーザーと外部ユーザーを区別できるようにする必要があります。
接続と資格情報  ディスカバリー、サービスマッピング、およびクラウド管理にコンピューターまたはネットワークデバイスへのアクセス権を得る、またはオーケストレーションを使用して作業を実行するには、認証情報と接続情報が必要です。	**ServiceNow** アクセス制御  SNC アクセス制御プラグインでは、どのカスタマーサービスとサポート従業員がインスタンスにいつアクセスできるかを制御できます。	

ゼロトラストアクセス (ZTA)

Zero Trust Access (ZTA) は、デフォルトで信頼できるユーザーまたはデバイスがないことを前提とするセキュリティモデルです。

探索

Zero Trust アクセスの機能とビジネス価値について説明します。

アクティブ化

Zero Trust アクセスを有効にする方法を把握します。

構成

Zero Trust アクセスを設定します。

参照 - プロパティ

自動翻訳

Zero Trust Access の詳細

Zero Trust Access (ZTA) は、デフォルトで信頼できるユーザーまたはデバイスがないことを前提とするセキュリティモデルです。

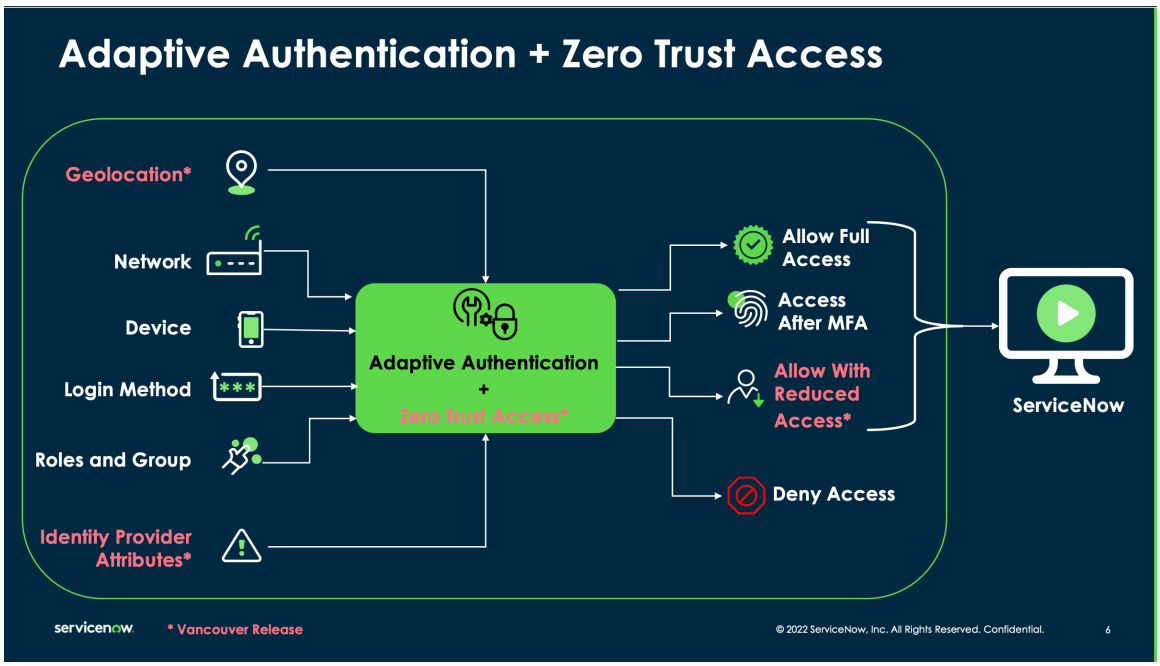
ZTA では、ユーザーの ID 検証とリスクアセスメントの後にのみ、アプリケーションとデータへのすべてのアクセスが最小限の権限で付与されます。

Zero Trust - Policy Based Session Access

ServiceNow Zero Trust - Policy Based Session Access (Session Access) を使用すると、IP アドレス、場所、認証方法、ユーザーのロール、グループ、MFA と ID プロバイダー (IDP) によって共有される属性を持つユーザーなど、さまざまな要素に基づいて Web セッションのユーザー権限を動的に削減できます。これにより、高権限ユーザーが信頼できないデバイスや場所からアプリケーションにアクセスする場合でも、無許可のアクセスやデータ侵害から組織を保護できます。

Zero Trust - Policy Based Session Access

セキュリティアドミニストレーターは、適応認証ポリシーを使用して、IP アドレス、場所、ID プロバイダー属性、およびユーザー属性に基づいて、セッション内のユーザーアクセスを削減または制限できます。



自動翻訳

注:

- セッションアクセスの構成は security_admin ロールでのみ実行できます。ロールを security_admin に昇格させる必要があります。
- セッションアクセスは統合をサポートしていません。
- 削減または制限されたロールがユーザーにアサインされていない場合、セッションアクセスは影響を及ぼしません。この場合、ログインしたセッションに変更はありません。ユーザーは引き続き、アサインされた権限でインスタンスにアクセスできます。
- ユーザーがインスタンスにログイン済みで、同時にアドミニストレーターがポリシーを設定する場合、セッションアクセスは影響を及ぼしません。ポリシーを有効にするには、ユーザーがセッションからログアウトする必要があります。
- ユーザーが信頼できるネットワークにあり、後でセッション内で VPN に切り替えた場合 (場所またはネットワークの変更)、セッションアクセスは影響を及ぼしません。
- セッションアクセスはログイン時に適用されます。セッション中にリスクパラメーターが変更されても、アクセスが制限されることはありません。たとえば、ユーザーがセッションの確立後に企業ネットワークから信頼できないネットワークに切り替えても、ユーザーがログアウトして再度ログインしない限り、アクセスが低下することはありません。
- セッションアクセス (Zero Trust Access - ZTA) 機能では、 snc_internal や snc_external などのロールは削除できません。
- セッションアクセス (Zero Trust アクセス - ZTA) 機能では、 sys_user_has_role またはユーザーグループメンバーシップテーブルからロールが削除されることはありません。ZTA ポリシーに基づいて、削減または制限されたロールでユーザーセッションを確立します。
- システムコンテキストで実行されるスクリプトは、ZTA セッションロールを考慮しません。

ユースケース

Zero Trust アクセスのユースケースの一部を次に示します。

- セッションに関連するリスクに基づいて権限を削減します。例：信頼できるネットワークの外部からログインする履行者ロールのユーザーが、セッションの要求者ロールのみを持つように構成できます。
- ユーザーが信頼できないデバイスを使用している場合は、ユーザーセッションの IDP 応答に基づいてアクセスを削減します。詳細については、「セッションアクセスの ID プロバイダー属性の設定」を参照してください。

このロールの降格により、ユーザーがセッションで他の既存の権限を持たないことが保証されます。ユーザーが信頼できるネットワークからログインすると、既存のすべての権限がセッションにアサインされます。

複数の IP 条件と複数のロールまたはグループのアサインをポリシーの一部として定義できます。

Zero Trust Access - Mobile

適応認証ポリシー内の Zero Trust Access - Session Access ポリシーを使用すると、モバイルにおける特定のセッションのロールまたは権限を減らすことができます。

Zero Trust Access - Session Access Mobile は、システムプロパティテーブルで **glide.authenticate.session_access.mobile.enabled** を有効にすると有効にできます。

IDP 属性で Zero Trust Access - Session Access Mobile を使用するには、**glide.authenticate.session_access.mobile.refresh_token_interval** フィールドを設定します。これにより、アドミニストレーターはリフレッシュトークンに基づいてセッションアクセスを効果的に制御できます。

詳細については、「モバイルの Zero Trust アクセスの構成」を参照してください。

Zero Trust Access のアクティブ化

Zero Trust - Policy Based Session Access com.snc.zero_trust_session_access プラグインをアクティブ化すると、セキュリティ管理者は、適応認証ポリシーを使用して、IP アドレス、場所、ID プロバイダー属性、およびユーザー属性に基づいて、セッション内のユーザーアクセスを削減または制限できます。

始める前に
必要なロール：admin

プラグインタイプ：有料でライセンスが必要です。

手順

1. 移動先 すべて > システムアプリケーション > 利用可能なすべてのアプリケーション > すべて.
2. フィルター基準と検索バーを使用して、**Zero Trust - Policy Based Session Access** (com.snc.zero_trust_session_access) プラグインを検索します。

   名前または ID でプラグインを検索できます。プラグインが見つからない場合は、ServiceNow 担当者から要求する必要があります。
3. [インストール] を選択して、インストールプロセスを開始します。

   注: ドメインセパレーションと代理アドミンがインスタンスで有効になっている場合、管理ユーザーはグローバルドメインに含まれている必要があります。それ以外の場合、次のエラーが表示されます： 「別の操作が実行されているため、アプリケーションのインストールは利用できません： <プラグイン名> のプラグインの有効化 (Application installation is unavailable because another operation is running: Plugin Activation for <plugin name>)」

   インストールが完了するとメッセージが表示されます。プラグインとともにインストールされるコンポーネントの詳細については、「 アプリケーションとともにインストールされているコンポーネントの検索 」を参照してください。

セッションアクセスロールの構成

適応認証ポリシーを使用して、IP、場所、ID プロバイダー属性、およびユーザー属性に基づいて、セッション内のユーザーアクセスを削減するようにセッションアクセスを構成します。

始める前に
必要なロール：security_admin

注:

- セッションアクセスの構成は security_admin ロールでのみ実行できます。ロールを security_admin に昇格させる必要があります。
- セッションアクセスは統合をサポートしていません。
- 削減または制限されたロールがユーザーにアサインされていない場合、セッションアクセスは影響を及ぼしません。この場合、ログインしたセッションに変更はありません。ユーザーは引き続き、アサインされた権限でインスタンスにアクセスできます。
- ユーザーがインスタンスにログイン済みで、同時にアドミニストレーターがポリシーを設定する場合、セッションアクセスは影響を及ぼしません。ポリシーを有効にするには、ユーザーがセッションからログアウトする必要があります。
- セッションアクセスはログイン時に適用されます。セッション中にリスクパラメーターが変更されても、アクセスが制限されることはありません。たとえば、ユーザーがセッションの確立後に企業ネットワークから信頼できないネットワークに切り替えても、ユーザーがログアウトして再度ログインしない限り、アクセスが低下することはありません。
- セッションアクセス (Zero Trust Access - ZTA) 機能では、 snc_internal や snc_external などのロールは削除できません。
- セッションアクセス (Zero Trust アクセス - ZTA) 機能では、 sys_user_has_role またはユーザーグループメンバーシップテーブルからロールが削除されることはありません。ZTA ポリシーに基づいて、削減または制限されたロールでユーザーセッションを確立します。
- システムコンテキストで実行されるスクリプトは、ZTA セッションロールを考慮しません。

手順

1. 移動先 すべて > **Zero Trust** アクセス > セッションアクセスロール構成.
2. セッションアクセスロールの構成を作成するには、[新規] を選択します。
3. フォームの各フィールドに入力します。

セッションアクセスのロール構成

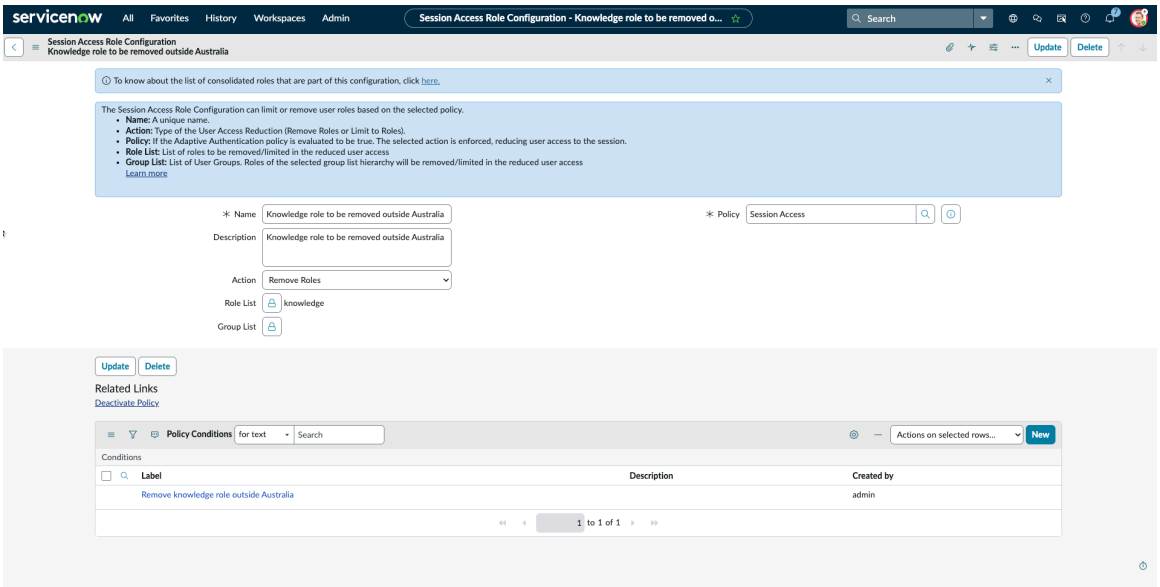
フィールド	説明
名前	構成の名前
説明	構成の簡単な説明
ポリシー	適応認証ポリシーを選択します。ルックアップアイコンを使用して、ポリシーのリストを表示します。
アクション	[ロールを削除] または [ロールに制限 **(Limit to Roles)**] ○ ロールを削除：構成されたユーザーがログインすると、ロールリストまたはグループリストで提供されたロールのリストがセッションに対して削除されます。 ○ ロールに制限 **(Limit to Roles)**：構成されたユーザーがログインすると、選択されたロールのみがユーザーに提供され、他のすべてのロールはセッションから削除されます。
ロールリスト	ロールリストからロールを選択します。

フィールド	説明
グループリスト	ユーザーを削除または制限するロールをグループリストから選択します。

4. [送信] を選択します。

   構成された国に基づいたユーザーのログインは次のとおりです。

   - [ロールを削除] で、選択したロールを持つ構成済みの国のユーザーは、セッションに対してそれらのロールを持たなくなります。
   - [ロールに制限 **(Limit to Roles)**] で、選択したロールを持つ構成済みの国のユーザーは、セッションに対してのみそれらのロールを持ちます。

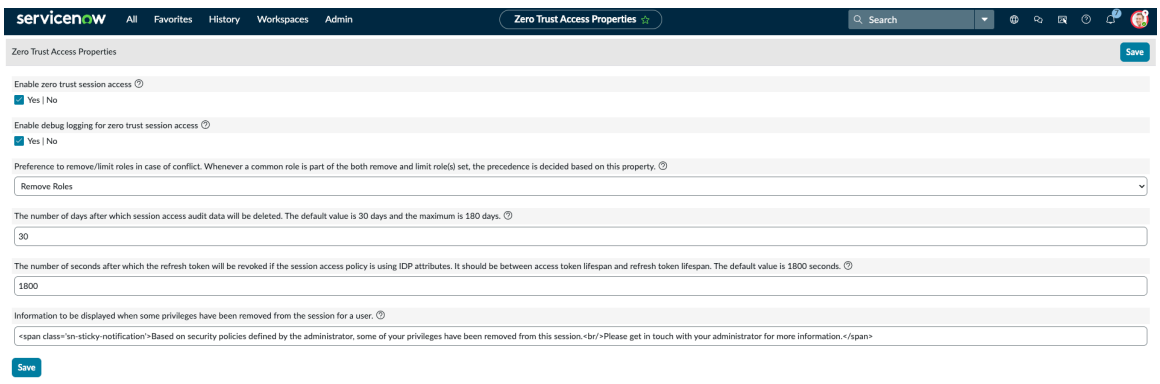


   サンプルユースケースで説明されているセッションに対してロールを削除または制限する方法の詳細については、「チュートリアル：Zero Trust アクセスを使用する」を参照してください。

Zero Trust Acess のシステムプロパティ

システムプロパティを使用すると、Zero Trust アクセスを有効化してカスタマイズし、セキュリティ要件を満たすことができます。

プロパティ



Zero Trust Acess のシステムプロパティ

プロパティ	説明
Zero Trust セッションアクセスを有効にする	アドミニストレーターが Zero Trust セッションアクセス機能を使用できるようにするオプションデフォルト値は false です。
Zero Trust セッションアクセスのデバッグのログ記録を有効にする	Zero Trust セッションアクセスのデバッグのログ記録を有効にするオプション。
競合が発生した場合にロールを削除/制限する設定。共通ロールが削除ロールと制限ロールの両方のセットに含まれる場合は常に、このプロパティに基づいて優先順位が決定されます。(Preference to remove/limit roles in case of conflict. Whenever a common role is part of both remove and limit role(s) set, the precedence is decided based on this property.)	[ロールを削除] または [ロールに制限 **(Limit to Roles)]**
セッションアクセス監査データが削除されてからの日数。デフォルト値は 30 日で、上限は 180 日です。(The number of days after which session access audit data will be deleted. The default value is 30 days and the maximum is 180 days.)	デフォルトは 30 日です。
セッションアクセスポリシーが IDP 属性を使用している場合に、リフレッシュトークンが取り消されるまでの秒数。アクセストークンの有効期間と更新トークンの有効期間の間である必要があります。デフォルト値は 1800 秒です。	デフォルトは 1800 秒です。
ユーザーのセッションから一部の権限が削除されたときに表示される情報 (Information to be displayed when some privileges have been removed from the session for a user.)	権限の制限または削除に関してユーザーに表示する説明。サンプルの説明は次のとおりです。 アドミニストレーターによって定義されたセキュリティポリシーに基づいて、一部のロールがこのセッションから削除されました。詳細については、アドミニストレーターにお問い合わせください。(Based on security policies defined by the administrator, some of your roles have been removed from this session. Please get in touch with your administrator for more information.)

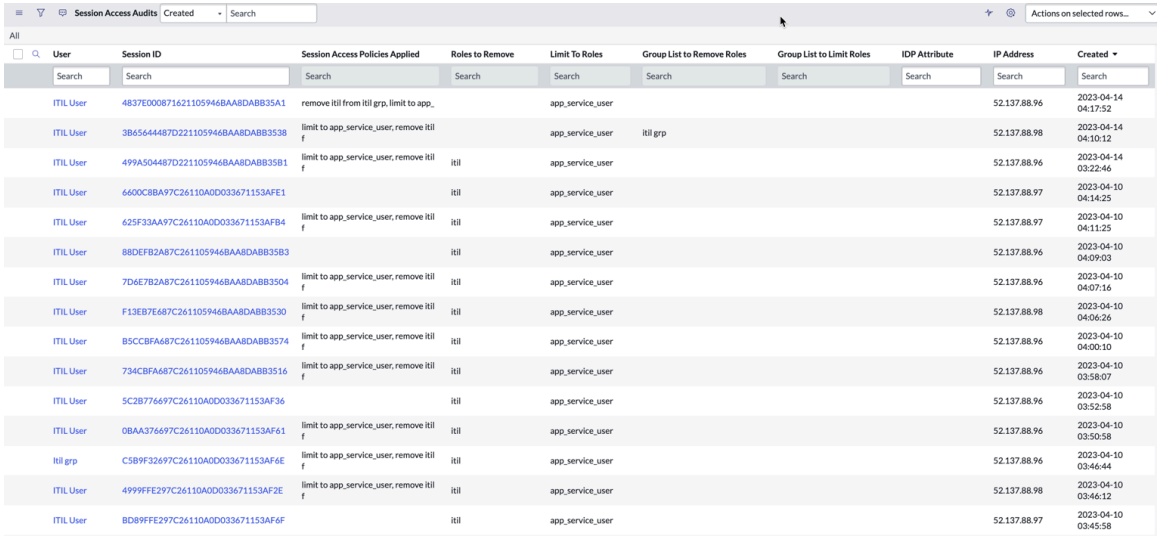
セッションアクセス監査

セッションアクセス監査には、ユーザーのセッションに関連するセッションアクセスログと情報が表示されます。

監査

セッションアクセス監査には、監査情報が次のように表示されます。

注: glide.authenticate.session_access.log_audit_event プロパティを使用して監査情報を入力します。



セッションアクセス監査

フィールド	説明
ユーザー	ユーザーの詳細
セッション ID	一意の ID として表示されるセッションの詳細
モバイルクライアント	モバイルクライアントの詳細。**ServiceNow Agent** (Now Agent) と **ServiceNow Request** (Now Mobile)。
セッションアクセスポリシーが適用されました	適用されたセッションアクセスポリシー。
削除するロール	ログイン中にユーザーから削除されたロール。
ロールに制限	ログイン中にユーザーに制限されたロール
ロールを削除するグループリスト	ユーザーのロールを削除したグループに関する情報。
ロールを制限するグループリスト	ユーザーのロールを制限したグループに関する情報。
モバイルクライアント	モバイルからログインしており、アクセス権が制限または削除されたユーザーの詳細。
IDP 属性	そのセッションに使用された IDP
IP アドレス	ユーザーがログインに使用する IP アドレスの詳細
作成済み	作成されたユーザーレコードの日時の詳細

チュートリアル：**Zero Trust** アクセスを使用する

エンドツーエンドのユースケースで Zero Trust アクセス機能を使用する手順。

始める前に
必要なロール：security_admin

[セッションアクセスの有効化 **(Enable Session Access property)**] を有効にします。

注:

- セッションアクセスの構成は security_admin ロールでのみ実行できます。ロールを security_admin に昇格させる必要があります。
- セッションアクセスは統合をサポートしていません。
- 削減または制限されたロールがユーザーにアサインされていない場合、セッションアクセスは影響を及ぼしません。この場合、ログインしたセッションに変更はありません。ユーザーは引き続き、アサインされた権限でインスタンスにアクセスできます。
- ユーザーがインスタンスにログイン済みで、同時にアドミニストレーターがポリシーを設定する場合、セッションアクセスは影響を及ぼしません。ポリシーを有効にするには、ユーザーがセッションからログアウトする必要があります。
- セッションアクセスはログイン時に適用されます。セッション中にリスクパラメーターが変更されても、アクセスが制限されることはありません。たとえば、ユーザーがセッションの確立後に企業ネットワークから信頼できないネットワークに切り替えても、ユーザーがログアウトして再度ログインしない限り、アクセスが低下することはありません。
- セッションアクセス (Zero Trust Access - ZTA) 機能では、 snc_internal や snc_external などのロールは削除できません。
- セッションアクセス (Zero Trust アクセス - ZTA) 機能では、 sys_user_has_role またはユーザーグループメンバーシップテーブルからロールが削除されることはありません。ZTA ポリシーに基づいて、削減または制限されたロールでユーザーセッションを確立します。
- システムコンテキストで実行されるスクリプトは、ZTA セッションロールを考慮しません。

セッションアクセスは、信頼できないネットワークからのログイン、別のデバイスからのログインなど、ユーザーがさまざまな環境からインスタンスにログインを試行するときに、アドミニストレーターがユーザーに対する一連のロールを動的に削減または制限できるようにする機能です。

セッションアクセスは、構成を実行するときに作成されたポリシーと選択したアクションによって制御できます。シナリオの一部は次のとおりです。

- ポリシーが true で、ロールアクションが [ロールを削除] に設定されている場合、ユーザーがインスタンスにログインしようとすると、選択したロールとそれに関連する子ロールがそのユーザーから削除されます。
- ポリシーが true で、ロールアクションが [ロールに制限 **(Limit To Roles)]** に設定されている場合、ユーザーがインスタンスにログインしようとすると、選択したロールとそれに関連付けられた子ロールのみがそのユーザーにアサインされます。

次の手順では、インスタンスにログインしているユーザーにロールが制限されるセッションアクセス構成のエンドツーエンド構成について説明します。同様に、構成中に [ロールを削除] オプションを選択して、ロールを削除することもできます。

手順

1. 移動先 すべて > セッションアクセス > セッションアクセスロール構成.
2. [セッションアクセスロールの構成 (Session Access Role Configurations)] ページで、[新規] を選択します。
3. ユーザーのロールを制限するには、フォームのフィールドに入力します。
   - 名前
   - 説明
   - ポリシー
   - アクション

- ロールリスト
- グループリスト

a. [ロールに制限 **(Limit to Roles)]** を選択して、ユーザーのロールを制限します。
例：**itil**

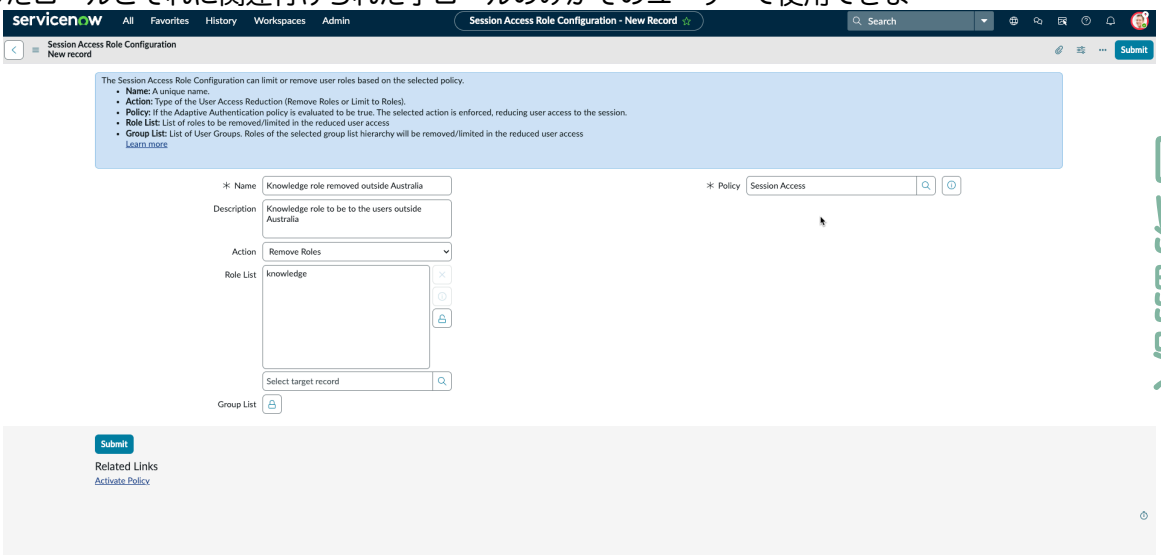
b. ロールリストから [ナレッジ] ロールを選択します。

c. [ポリシー] を選択します。

認証ポリシーとフィルター基準 (ロール、グループ、IP、場所) をポリシーの入力と条件とともに使用して、セッションアクセスポリシーを作成できます。

セッションアクセス構成でポリシーを使用します。たとえば、ロール (ナレッジ) を、場所 (オーストラリア) の外部でログインしているユーザーに制限するとします。

d. [アクション] で [ロールに制限 **(Limit to Roles)]** を選択します。
ポリシーが true の場合、ユーザーがインスタンスにログインしようとすると、選択したロールとそれに関連付けられた子ロールのみがそのユーザーで使用できます。

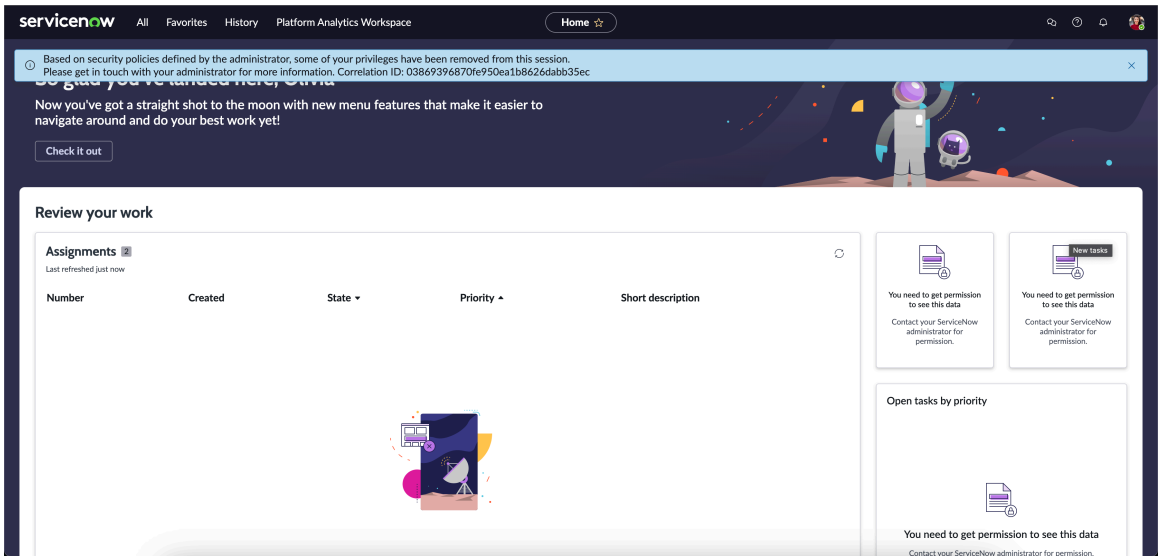


e. [送信] を選択します。

同様に、グループリストからグループを選択して、グループ内のユーザーのロールを制限または削除できます。

ユーザーがオーストラリア国外のインスタンスにログインすると、[ナレッジ] ロールとそれに関連する子ロールのみがログインセッションにアサインされ、ユーザーに対する他のロールは制限されます。

ログイン後、プラットフォームのプロファイルセクションに次のエラーメッセージが表示されます。



ユーザーはアドミニストレーターに連絡し、調査のために相関 ID を提供できます。

注: 相関 ID は、セッションアクセス監査テーブル内で対応する監査レコードの sys_id です。

セッションアクセスの **ID** プロバイダー属性の設定

セキュリティアサーションマークアップ言語 (SAML) 応答と OpenID Connect (OIDC) から作成された ID プロバイダー (IDP) 属性を使用して、インスタンスへのユーザーセッションアクセスを削除または制限します。

始める前に

必要なロール：security_admin

[セッションアクセスの有効化 **(Enable Session Access property)]** を有効にします。

注: セッションアクセスロール構成を使用するには、ロールを security_admin に昇格させる必要があります。

セッションアクセスは、構成を実行するときに作成されたポリシーと選択したアクションによって制御できます。シナリオの一部は次のとおりです。

- ポリシーが true で、ロールアクションが IDP 属性の入力と条件とともに [ロールを削除] に設定されている場合、ユーザーがインスタンスにログインしようとすると、選択したロールとそれに関連付けられた子ロールがそのユーザーから削除されます。
- ポリシーが true で、ロールアクションが IDP 属性の入力と条件とともに [ロールに制限 **(Limit To Roles)]** に設定されている場合、ユーザーがインスタンスにログインしようとすると、選択したロールとそれに関連付けられた子ロールのみがそのユーザーにアサインされます。

次の手順は、SAML 応答からポリシー入力として IDP 属性を設定してセッションアクセスを制御するステップを示しています。

手順

1. 移動先 すべて > セッションアクセス > セッションアクセスロール構成.
2. [セッションアクセスロールの構成 (Session Access Role Configurations)] ページで、[新規] を選択します。
3. ユーザーのロールを削除するには、フォームのフィールドに入力します。

- 名前
- 説明
- ポリシー
- アクション
- ロールリスト
- グループリスト

a. [ロールを削除] を選択して、ユーザーのロールを削除します。
例：**itil**

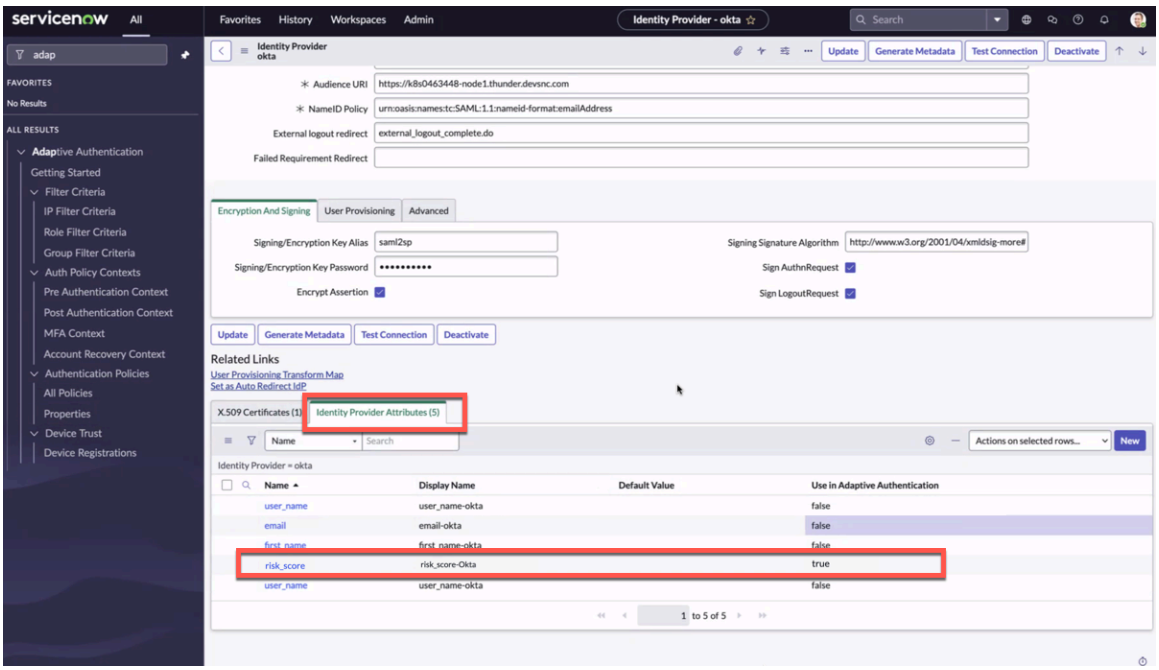
b. ロールリストから **[itil]** ロールを選択します。

c. [ポリシー] を選択します。
適応認証ポリシー作成を使用してさまざまなフィルター基準でポリシーを作成する方法の詳細については、「フィルター基準」を参照してください。

d. [アクション] で [ロールを削除] を選択します。
ポリシーが true で、ロールアクションが [ロールを削除] に設定されている場合、ユーザーがインスタンスにログインしようとすると、選択したロールがそのユーザーから削除されます。

e. ポリシー入力で、ポリシーの入力と条件を作成します。
例：

- ポリシー入力：Okta (IDP) からのリスクスコア属性
- ポリシー条件：条件としての 60 ～ 80 のリスクスコア



この構成に基づいて、Okta (IDP) からのリスクスコア属性値が 80 を超える場合、ユーザーはインスタンスから削除されたロール (従業員) およびその子ロールで認証されず、アサインされた他のロールでのみ認証されます。リスクスコアが 60 ～ 80 の場合、ユーザーはすべてのロールでインスタンスに認証されます。

ポリシーと条件で認証後コンテキストのポリシーを作成する方法の詳細については、「認証後コンテキスト」を参照してください。

> **注:** [セッションアクセスの有効化 **(Enable Session Access property)]** プロパティが非アクティブの場合、セッションアクセスの構成によってユーザーのロールが制限または削除されることはありません。

f. [送信] を選択します。

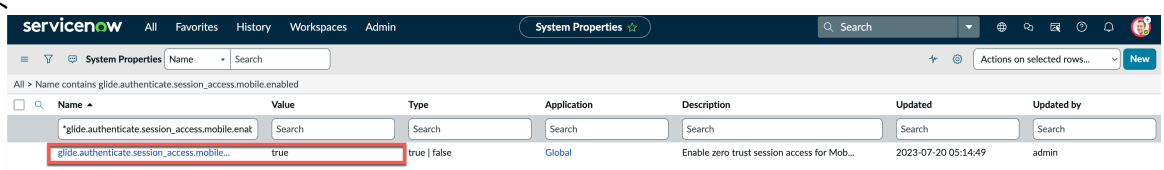
Zero Trust Access for Mobile

Zero Trust Access (ZTA) は、デフォルトで信頼できるユーザーまたはデバイスがないことを前提とするセキュリティモデルです。

適応認証ポリシー内の Zero Trust Access - Session Access ポリシーを使用すると、モバイルユーザー向けの特定のセッションのロールまたは権限を減らすことができます。

モバイルでゼロトラストアクセスを有効にするには、次のタスクを実行する必要があります。

- セッションアクセスの構成は security_admin ロールでのみ実行できます。ロールを security_admin に昇格させる必要があります。
- **Zero Trust - Policy Based Session Access** com.snc.zero_trust_session_access ポリシーを有効にします。
- システムプロパティテーブルで、**glide.authenticate.session_access.mobile.enabled** を有効にしま



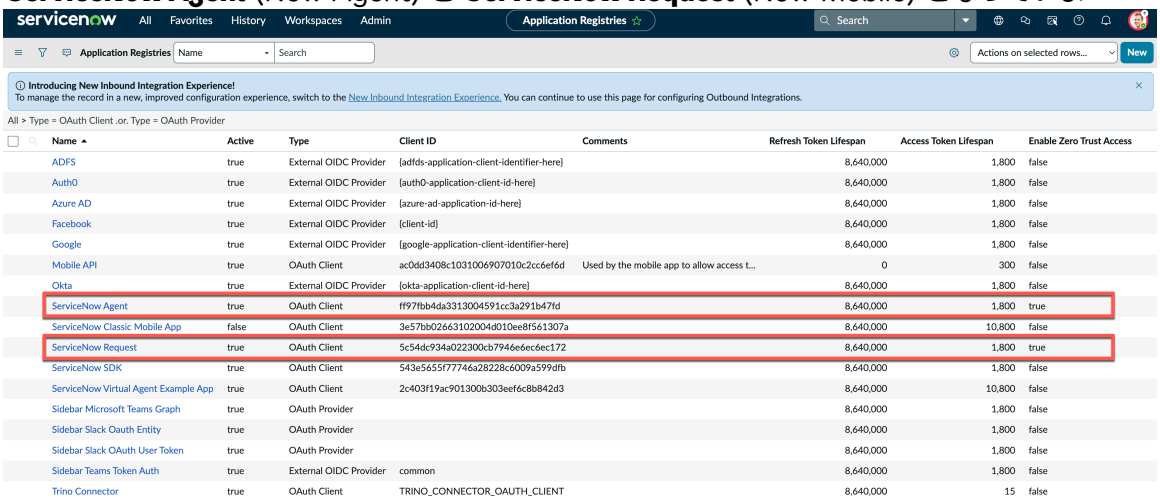
す。

- **glide.authenticate.session_access.mobile.refresh_token_interval** フィールドを設定し、リフレッシュトークンに基づいてモバイルでのセッションアクセスを制御しま

す。

> **注:** モバイルアプリのログインに IDP を使用する場合は、リフレッシュトークンの秒数を設定する必要があります。デフォルトでは、ユーザーは 1800 秒 (30 分) 後にモバイルアプリからログアウトされます。

- モバイルクライアントアプリケーション (OAuth クライアント) の [アプリケーションレジストリ] で [ゼロトラストアクセスを有効にする] を true に設定します。この場合、**ServiceNow Agent** (Now Agent) と **ServiceNow Request** (Now Mobile) となっています。



Name	Active	Type	Client ID	Comments	Refresh Token Lifespan	Access Token Lifespan	Enable Zero Trust Access
ADFS	true	External OIDC Provider	{adfds-application-client-identifier-here}		8,640,000	1,800	false
Auth0	true	External OIDC Provider	{auth0-application-client-id-here}		8,640,000	1,800	false
Azure AD	true	External OIDC Provider	{azure-ad-application-id-here}		8,640,000	1,800	false
Facebook	true	External OIDC Provider	{client-id}		8,640,000	1,800	false
Google	true	External OIDC Provider	{google-application-client-identifier-here}		8,640,000	1,800	false
Mobile API	true	OAuth Client	ac0dd3408c10310069070l0c2cc6ef6d	Used by the mobile app to allow access t...	0	300	false
Okta	true	External OIDC Provider	{okta-application-client-id-here}		8,640,000	1,800	false
ServiceNow Agent	true	OAuth Client	ff97fbb4da3313004591cc3a291b47fd		8,640,000	1,800	true
ServiceNow Classic Mobile App	false	OAuth Client	3e57bb02663102004d010ee8f561307a		8,640,000	10,800	false
ServiceNow Request	true	OAuth Client	5c54dc934a022300cb7946e6ec6ec172		8,640,000	1,800	true
ServiceNow SDK	true	OAuth Client	543e5655f77746a28228c6009a599dfb		8,640,000	1,800	false
ServiceNow Virtual Agent Example App	true	OAuth Client	2c403f19ac901300b303eef6c8b842d3		8,640,000	10,800	false
Sidebar Microsoft Teams Graph	true	OAuth Provider			8,640,000	1,800	false
Sidebar Slack Oauth Entity	true	OAuth Provider			8,640,000	1,800	false
Sidebar Slack OAuth User Token	true	OAuth Provider			8,640,000	1,800	false
Sidebar Teams Token Auth	true	External OIDC Provider	common		8,640,000	1,800	false
Trino Connector	true	OAuth Client	TRINO_CONNECTOR_OAUTH_CLIENT		8,640,000	15	false

す。

- セッションアクセスのロールを設定し、ポリシーの入力と条件に基づいてログ記録を行うユーザーのロールを減らすか削除します。設定の詳細については、「セッションアクセスロールの構成」を参照してください。

設定では、ログインを評価し、ポリシーのフィルターと条件に基づいて ServiceNow® インスタンスにアクセスするユーザーのロールを削減または削除します。詳細については、「モバイルの Zero Trust アクセスの構成」を参照してください。

継続認証 (CA)

ServiceNowの継続的な認証により、保護されているリソースにユーザーがアクセスした場合に、ユーザーを再検証して認証できます。

探索

継続的認証の機能とビジネス価値について説明します。

アクティブ化

継続的認証を有効にする方法を理解します。

構成

継続的認証を構成します。

参照 - プロパティ

継続的認証のプロパティについて説明します。

自動翻訳

継続的認証の詳細

ServiceNowの継続的認証 (CA) を使用すると、保護されているリソースにユーザーがアクセスした場合に、ユーザーを再検証して認証できます。

ServiceNowの継続的認証は、最初のログイン時だけでなく、ユーザーのセッション全体を通じてユーザーの ID を確認するように設計されたセキュリティ メカニズムです。CAは、 ServiceNowのゼロトラストアクセスセキュリティアーキテクチャに基づいて構築されており、最初の認証プロセス後もユーザーが主張する人物を維持できるようにすることでセキュリティを強化することを目的としています。

CA は、次のゼロトラストアクセス原則に基づいて作業します。

- 明示的に検証する:場所に関係なく、ネットワーク内のユーザー、デバイス、またはシステムに対する暗黙的な信頼はありません。すべてのユーザーとデバイスは、場所や過去のアクセスに関係なく、明示的に認証および承認される必要があります。
- 最小限の権限アクセスを使用する:特定のタスクを実行するために必要な最小限のアクセスまたは権限のみを付与し、侵害されたアカウントまたはシステムからの潜在的な損害を制限します。
- 侵害を想定する:予防だけに頼るのではなく、侵害を想定し、プロアクティブな検出、封じ込め、対応に重点を置きます。

CA は、ユーザーがアクセスしているデータとユーザーが実行しているアクティビティに基づいて、ステップアップ認証または再認証を適用する機能を提供します。アドミニストレーターは、テーブルまたはデータクラスレベルでセキュリティポリシーを作成するために選択できます。

ユーザーが個人識別可能情報 (PII) や機密データにアクセスしようとするたびに、ログインセッション内でステップアップ認証 (MFA) または再認証 (SSO - SAML または OIDC) を適用できます。

> **注:** ライセンスが必要な CA を選択するには、 **Zero Trust - Continuous Authentication** (com.snc.zero_trust_continuous_authentication) をインストールする必要があります。

福利厚生

CA を使用するメリットは次のとおりです。

- セキュリティの強化: ユーザーの ID を継続的に検証することで、システムは潜在的なセキュリティの脅威をより迅速に検出して対応できます。
- アカウント乗っ取りのリスクの低減:攻撃者がユーザーのセッションにアクセスした場合でも、継続的な認証により、機密データへのアクセスを防ぐことができます。

ユースケース

CA を使用するユースケースの一部を次に示します。

- さまざまなポリシーを使用して機密データへのアクセスを許可する前に再認証を強制します。
- さまざまなポリシーを使用して定期的な再認証またはステップアップ認証を強制します。
  - IdP の MFA、IdP の SSO を含むことができる再認証を使用します。
  - ServiceNowの MFA でステップアップ認証を使用します。

CA でのロール

CA には次のロールがあります。

- CA アドミン (ca_admin):CA ポリシーを作成、編集、および表示する機能。CA プロパティを設定し、CA のダッシュボード (メトリクス) を表示します。
- ポリシーアドミン (ca_policy_admin):CA ポリシーを作成、編集、および表示する権限。
- 監査人 (ca_auditor):CA のダッシュボード (メトリクス) を表示する機能。ポリシーや CA のログも同様です

CA を設定するには、ロールを **ca_admin** に昇格させ、ポリシー構成を実行する必要があります。

注: これら 3 つのロールはすべて昇格したロールです。

CA のモジュール

CA 内のさまざまなモジュールは次のとおりです。

- ポリシー:作成されたさまざまな継続認証ポリシーを表示します。
- メトリクス:KPI の目的で継続的認証のさまざまなメトリクスを表示し、組織内での CA の使用状況を把握します。
- システムプロパティ:システムプロパティを使用して、ゼロトラストアクセスのセキュリティ要件を満たすために継続的認証 (CA) を有効にしてカスタマイズします。

関連トピック

- 方針
- メトリクス
- システムプロパティ
- 継続認証の事前作業
- 継続認証のアクティブ化
- 継続認証の構成
- 継続認証を使用する高保証セッション

方針

作成されたさまざまな継続認証ポリシーを表示します。

CA ポリシーフォームには、テーブルまたはデータクラスに対して作成された CA ポリシーの詳細が含まれています。カスタムポリシーからポリシーレコードを管理できます。

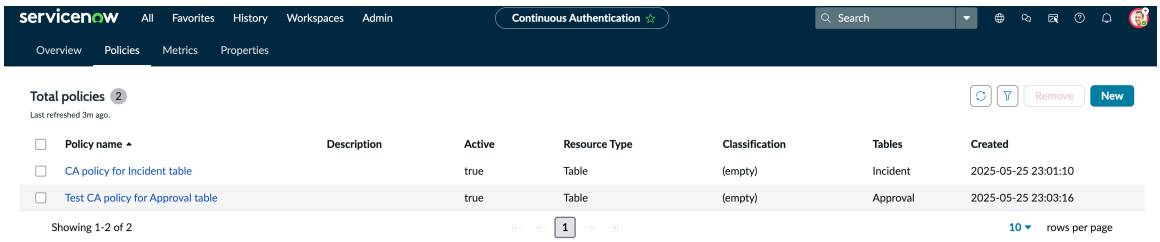
ポリシーページにアクセスするには、 すべて > 継続認証をクリックし、[ポリシー] タブを選択します。ページに表示されるポリシーの詳細は次のとおりです。

ポリシーページ

フィールド	説明
ポリシー名	ポリシー名
説明	ポリシーの説明
アクティブ	ポリシーステータス
リソースタイプ	ポリシーに使用する選択されたリソースタイプ
分類	ポリシーに対して選択されたデータ分類
テーブル	ポリシーに対して選択されたテーブル

ポリシーページ (続く)

フィールド	説明
作成日時	ポリシー作成の詳細



関連トピック

- メトリクス
- システムプロパティ
- 継続的認証の詳細

メトリクス

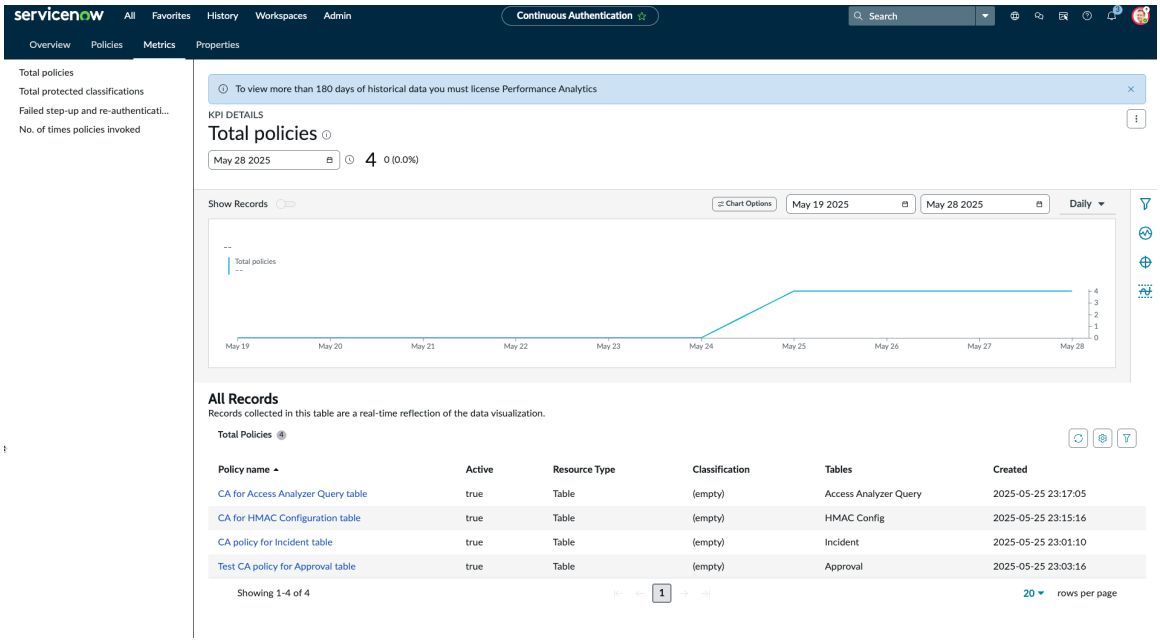
継続的認証のさまざまなメトリクスを表示します。

メトリクスページにアクセスするには、 すべて > 継続認証をクリックし、[メトリクス] タブを選択します。

継続的認証で表示できるさまざまな KPI 詳細は次のとおりです。

メトリクス

メトリクス	説明
合計ポリシー	ユーザーに対して作成された継続認証ポリシーの数の KPI。
保護された分類の合計	CA ポリシーの作成によって保護される分類の数の KPI。
ステップアップまたは再認証の失敗	ステップアップ (MFA) または再認証 (SSO) の失敗数の KPI。
ポリシーが呼び出された回数	CA ポリシーが呼び出された合計回数の KPI。



注: KPI 詳細が 180 日間表示されます。180 日を超える履歴データを表示するには、 パフォーマンスアナリティクスのライセンスを取得する必要があります。

関連トピック

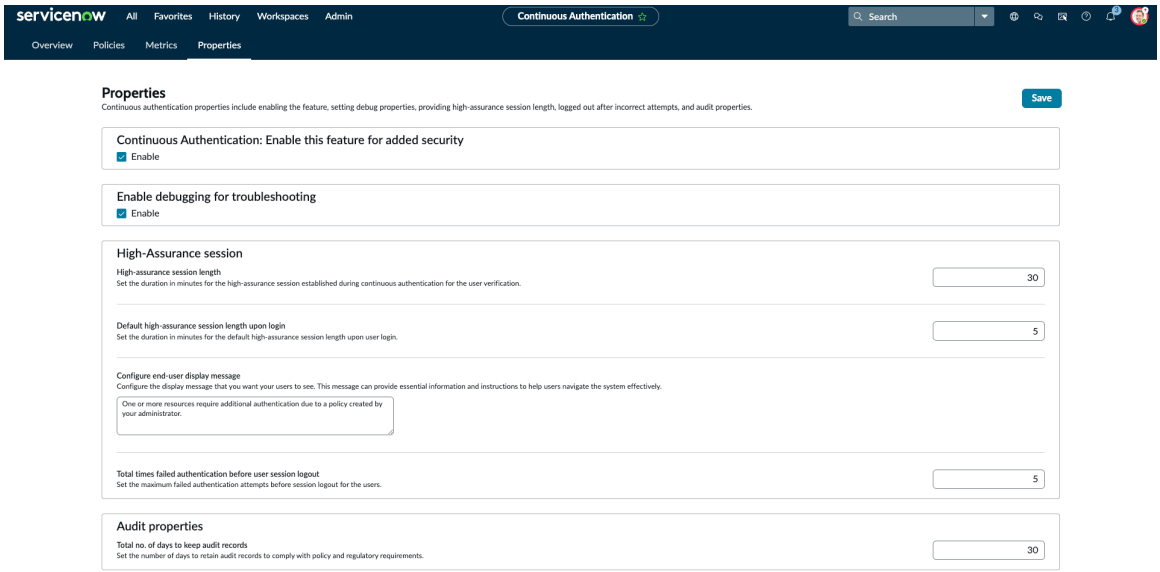
- 方針
- メトリクス
- システムプロパティ
- 継続的認証の詳細

システムプロパティ

システムプロパティを使用して継続的認証 (CA) を有効にしてカスタマイズし、ゼロトラストアクセスのセキュリティ要件を満たします。

プロパティ

プロパティページにアクセスするには、次に移動します: すべて > 継続認証をクリックし、[プロパティ] タブを選択します。



CA のさまざまなシステムプロパティは次のとおりです。

継続認証システムプロパティ

プロパティ	説明
一般的なプロパティ	
継続認証 (*glide.zta.continuous_authentication.enabled*)	継続認証機能の使用を有効にする
デバッグを有効にする (*glide.zta.continuous_authentication.debug.enabled*)	継続認証に関連するデバッグ情報を表示できるようにします。
高保証	
ハイアシュアランスセッションの長さ (*glide.zta.high_assurance.session.timeout*)	エンドユーザーが再認証する必要があるハイアシュアランスセッションの長さを指定します。デフォルト:10 分。 **注:** 値は 1 ～ 480 の範囲でなければなりません。
ログイン時のデフォルトのハイアシュアランスセッション長 (*glide.zta.default.high_assurance.session.timeout*)	ユーザーログイン時のデフォルトのハイアシュアランスセッション長の期間を分単位で指定します。デフォルト値: **5** 分。 **注:** このプロパティは、ローカルログインにのみ適用されます。
エンドユーザー表示メッセージの構成 (*glide.zta.high_assurance.session.message*)	再認証のためにエンドユーザーに表示するメッセージを指定します。デフォルトメッセージ: アドミニストレーターが作成したポリシーにより、1 つ以上のリソースに追加認証が必要です。

継続認証システムプロパティ (続く)

プロパティ	説明
ユーザーアカウントのロックアウト前の認証失敗の合計時間 (*glide.zta.high_assurance.session.max.login.failed.attempts*)	ユーザーがログアウトされるまでの認証試行失敗の最大回数を設定します。 注: 値は 3 から 10 の範囲でなければなりません。
監査プロパティ	
監査レコードの合計保持日数 (*glide.zta.continuous_authentication.audit.lifespan*)	CA の監査レコードを保存する日数を指定します。 注: 値は 1 ～ 180 の範囲でなければなりません。
合計数非アクティブ化後にポリシーが削除されるまでの日数 (*glide.zta.continuous_authentication.policy.lifespan*)	CA ポリシーが削除されるまでの日数を指定します。

重要:

- デフォルトでは、ソースで継続的認証ポリシーがアクティブになっている場合でも、モバイルアプリセッションにハイアシュアランスセッションは必要ありません。この動作を変更してモバイルアプリセッションからのアクセスをブロックするには、**glide.zta.high_assurance.mobile.session.allowed** プロパティの値を false に更新します。
- **sys_properties**、**sys_continuous_auth_policy**、**sys_user**テーブルは CA から除外され、CA ポリシー構成に追加することはできません。

関連トピック

継続的認証の詳細

方針

メトリクス

継続認証の事前作業

継続認証の事前作業

継続認証 (CA) を使用する前に、次の事前作業を実行してください。

CAは、 ServiceNowのゼロトラストアクセスセキュリティアーキテクチャに基づいて構築されており、最初の認証プロセス後もユーザーが主張する人物を維持できるようにすることでセキュリティを強化することを目的としています。

注: ライセンスが必要な CA を選択するには、 **Zero Trust - Continuous Authentication** (com.snc.zero_trust_continuous_authentication) をインストールする必要があります。

CA 構成は、以下に基づいて実行できます。

- LDAP またはローカルログイン (ユーザー名とパスワード)
- SSO ログイン (SAML または OIDC)。

ローカルログインの CA

ユーザーがローカルログインを実行し、ユーザーの ID を確認する場合、MFA が認証メカニズムとして使用されます。ユーザーが MFA をセットアップしていない場合は、セットアップを完了する必要があります。

注:

- Yokohama 以降、ローカルログインを実行する ServiceNow 、ログインするたびに MFA が適用されます。
- MFA プロパティがアクティブで、要件に基づいて構成されていることを確認します。MFA プロパティの詳細については、「 マルチファクター認証システムプロパティ」を参照してください。

詳細については、「非 SSO ログインのハイアシュアランスセッション」を参照してください。

SSO ログインの CA

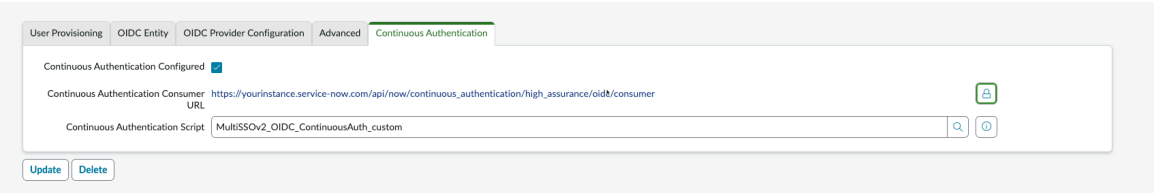
ユーザーが SSO ベースのログイン (SAML または OIDC) を実行している場合。ユーザーの ID を確認するために、初回ログイン時に使用されたのと同じ SSO が、再認証または IdP の MFA による再検証として表示されます。

注:

- ライセンスが必要な CA を選択するには、 **Zero Trust - Continuous Authentication** (com.snc.zero_trust_continuous_authentication) をインストールする必要があります。
- Integration - Multiple Provider Single Sign-On Installer (*com.snc.integration.sso.multi.installer*) プラグインをアクティブ化します。

CA の IDP を次のように設定する必要があります。

- 指定されたマルチ SSO レコードに設定する必要があるチェックボックスをオンにして、そのセットが CA を使用できることを検証します。



- OIDC の場合、CA は api/now/continuous_authentication/high_assurance/oidc/consumer エンドポイントへのリダイレクトに依存しており、IdP で設定する必要があります。再認証と IdP の MFA オプションの両方を使用できます。
- SAML の場合、SSO レコードはすべての ID プロバイダー (IdP) のデフォルトの再認証スクリプトを使用して再認証をサポートします。

  注:

  - **Okta** のステップアップを設定するには、IdP レコードの **ContinuousAuth_Okta_StepUp_Script** を使用します。詳細については、 このドキュメント を参照してください。
  - **Entra ID** または **Azure** のステップアップを構成するには、**ContinuousAuth_Azure_StepUp_Script**を使用して必要な要求を追加します。詳細については、 このドキュメント を参照してください。

詳細については、「SSO ログインのハイアシュアランス」を参照してください。

関連トピック

- 継続的認証の詳細
- 継続認証のアクティブ化
- 継続認証の構成

継続認証のアクティブ化

インスタンスで継続認証機能をアクティブ化するには、 **Zero Trust - Continuous Authentication** (com.snc.zero_trust_continuous_authentication) プラグインをインストールします。

始める前に

必要なロール：admin

このプラグインを使用すると、セキュリティアドミニストレーターは、ユーザーがアクセスしているデータとユーザーが実行しているアクティビティに基づいて、ログインセッション内でステップアップ認証 (MFA) または再認証 (SSO) を必要とするセキュリティポリシーを定義できます。

ライセンスが必要な CA を選択するには、 **Zero Trust - Continuous Authentication** (com.snc.zero_trust_continuous_authentication) をインストールする必要があります。

手順

1. 移動先 すべて > システムアプリケーション > 利用可能なすべてのアプリケーション > すべて.
2. フィルター基準と検索バーを使用して、 **Zero Trust - Continuous Authentication** (com.snc.zero_trust_continuous_authentication) プラグインを検索します。

   名前または ID でプラグインを検索できます。プラグインが見つからない場合は、ServiceNow 担当者から要求する必要があります。
3. [インストール] を選択して、インストールプロセスを開始します。

   注: ドメインセパレーションと代理アドミンがインスタンスで有効になっている場合、管理ユーザーはグローバルドメインに含まれている必要があります。それ以外の場合、次のエラーが表示されます： 「別の操作が実行されているため、アプリケーションのインストールは利用できません： <プラグイン名> のプラグインの有効化 (Application installation is unavailable because another operation is running: Plugin Activation for <plugin name>)」

   インストールが完了するとメッセージが表示されます。プラグインとともにインストールされるコンポーネントの詳細については、「 アプリケーションとともにインストールされているコンポーネントの検索 」を参照してください。

関連トピック

- 継続認証の事前作業
- 継続認証の構成
- 継続認証を使用する高保証セッション

継続認証の構成

継続的認証 (CA) ポリシーを構成して、保護されているリソースへのアクセスが試行された場合にユーザーを再認証します。

始める前に

- 必要なロール:admin (ca_admin)

  注: ロールを **ca_admin** に昇格させる必要があります。

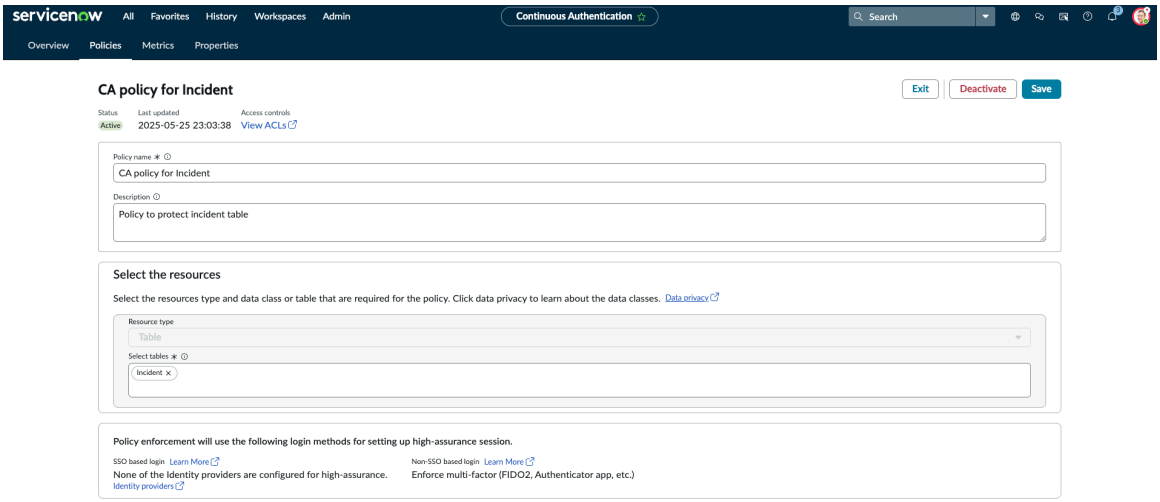
- ライセンスが必要な CA を選択するには、 **Zero Trust - Continuous Authentication** (com.snc.zero_trust_continuous_authentication) をインストールする必要があります。
- 継続的認証 (*glide.zta.continuous_authentication.enabled*) システムプロパティを有効にします。詳細については、「システムプロパティ」を参照してください。
- Integration - Multiple Provider Single Sign-On Installer (*com.snc.integration.sso.multi.installer*) プラグインをアクティブ化します。
- インスタンスの CA を構成する前に必要な事前作業を理解します。詳細については、「継続認証の事前作業」を参照してください。
- CA ポリシーは、データクラスまたはテーブルに対して構成できます。

手順

1. 移動先 すべて > 継続認証.
2. [ポリシー] タブを選択します。
3. **[New** (新規)] を選択します。
4. フォームの各フィールドに入力します。

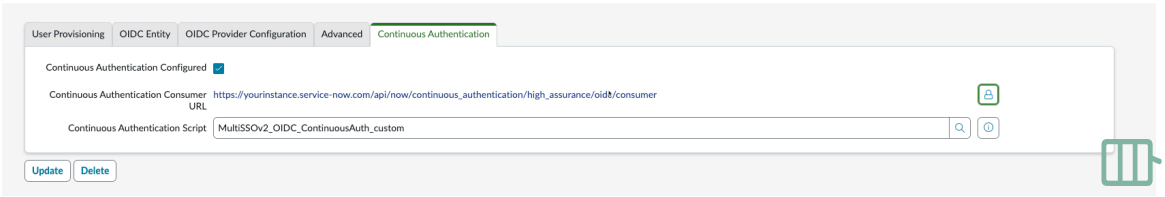
継続認証

フィールド	説明
ポリシー名	ポリシーの名前
説明	ポリシーの一般的な説明
リソースを選択	オプション： ○ データクラス。データクラスを作成し、CA ポリシー構成に使用できます。 注: データクラスの作成方法の詳細については、「 データ分類」を参照してください。 ○ テーブル 注: ○ メタデータとともに選択されたテーブルにエラーが表示されます。 ○ ユーザーの構成アクセスに影響を与える可能性があるため、実際にメタデータテーブルへのアクセスを制限するかどうかを確認する必要があります。 ○ **sys_properties**、**sys_continuous_auth_policy**、**sys_us** ブルは CA から除外され、CA ポリシー構成に追加することはできません。



注: CA ポリシーには、次のいずれかのログイン方法を使用できます。

- **SSO** ベースのログイン:ID プロバイダーレコード内の [継続的認証] タブで フィールドを指定し、ID プロバイダーレコードを アクティブとして設定しま す。

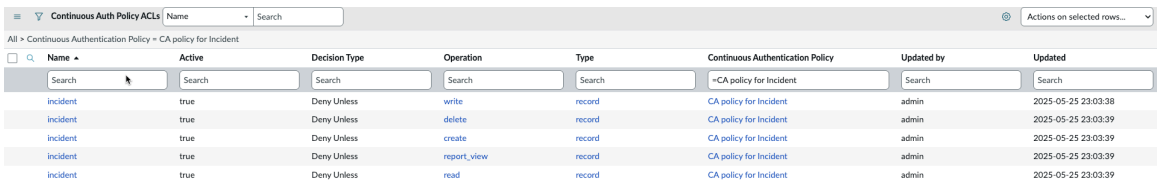


ID プロバイダーの構成の詳細については、「 OIDC と SAML」を参照してください。

- 非 **SSO** ベースのログイン:デフォルトでは、継続的認証構成の ID プロバイダーがない場合は、ログイン方法としてマルチファクター認証 (MFA) が使用されます。MFA プロパティがアクティブで、要件に基づいて構成されていることを確認します。MFA プロパティの詳細については、「 マルチファクター認証システムプロパティ」を参照してください。

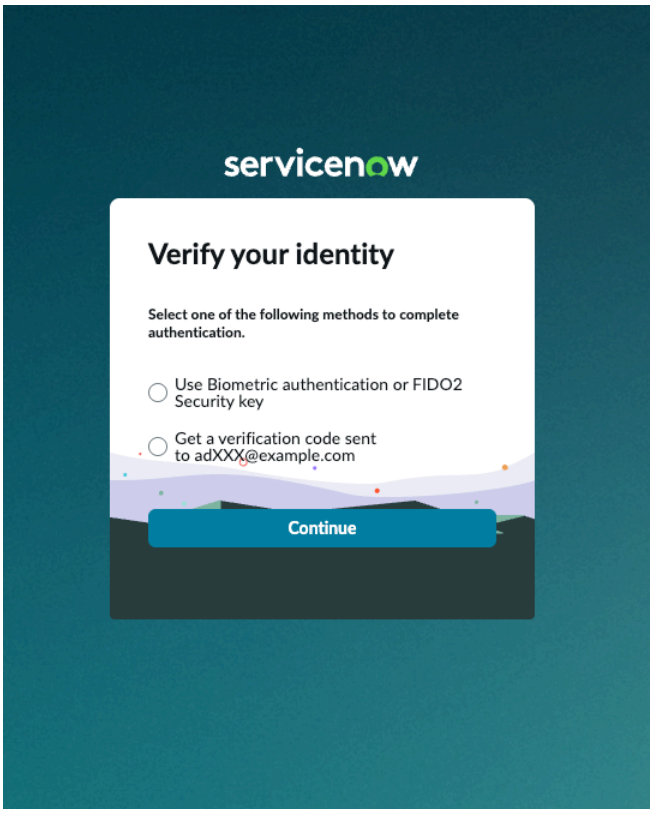
5. [保存してアクティブ化] を選択します。

結果
構成に指定された詳細に基づいて、選択したテーブルまたはデータクラスのアクセス制御リスト (ACL) を使用して CA ポリシーが作成されます。作成された ACL の詳細を表示するには、ポリシーページで **[ACL** の表示] を選択します。



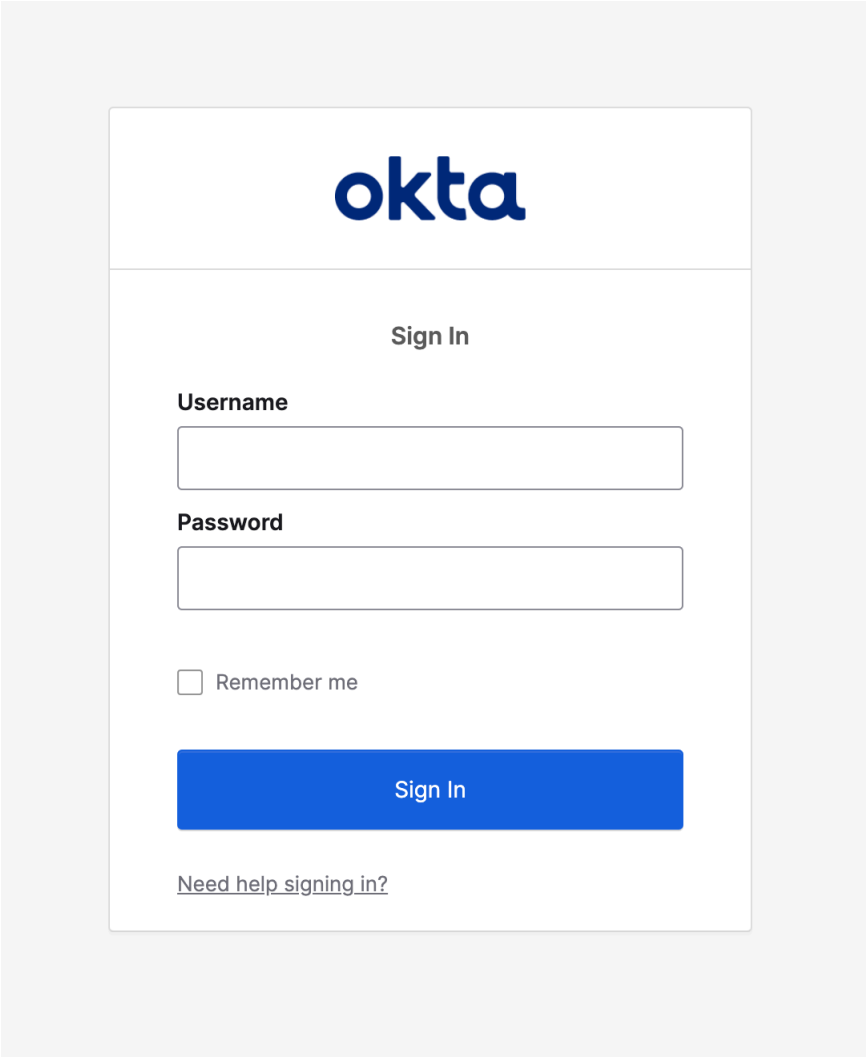
作成された CA ポリシーは、次のシナリオに基づいて、ポリシーを使用して保護したテーブルまたはデータクラスにアクセスするための認証をユーザーに求めます。

- インスタンスにログインするためにローカルログインを実行したユーザーには、ステップアップ認証用のプラットフォーム MFA が表示されます。



注: 認証に最近使用した MFA 要素が表示されます。

- インスタンスにログインするために SSO ログイン (OIDC または SAML) を実行したユーザーは、再認証のために SSO とともに表示されます。



これで、ユーザーに対してハイアシュアランスセッションが確立されました。高保証セッションは、高保証セッション長 (*glide.zta.high_assurance.session.timeout*) システムプロパティに制限されます。ハイアシュアランスセッション時間がプロパティ長を超えると、ユーザーは再認証またはステップアップ認証を求められます。

テーブルまたはデータの継続的認証のエンドツーエンド構成の詳細については、以下を参照してください。

- チュートリアル:テーブルの継続認証の構成。
- チュートリアル:データクラスの継続的認証の構成。

関連トピック

継続的認証の詳細

継続認証の事前作業

継続認証のアクティブ化

チュートリアル:テーブルの継続認証の構成

テーブルの継続的認証ポリシーのエンドツーエンド構成と、構成変更によるユーザーへの影響について説明する手順。

始める前に

- 必要なロール:admin (ca_admin)

  注: ロールを **ca_admin** に昇格させる必要があります。

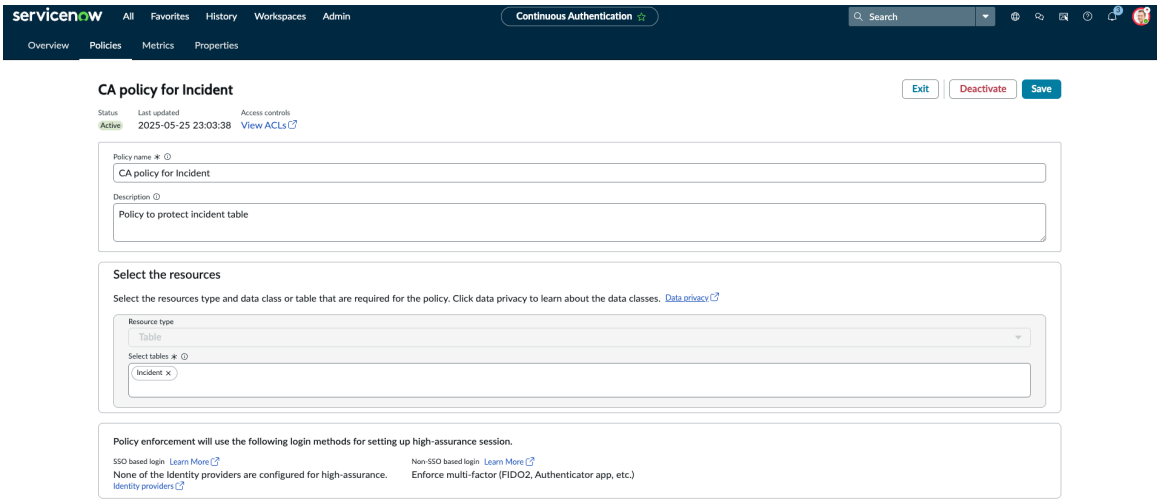
- ライセンスが必要な CA を選択するには、 **Zero Trust - Continuous Authentication** (com.snc.zero_trust_continuous_authentication) をインストールする必要があります。
- 継続的認証 (*glide.zta.continuous_authentication.enabled*) システムプロパティを有効にします。詳細については、「システムプロパティ」を参照してください。
- Integration - Multiple Provider Single Sign-On Installer (*com.snc.integration.sso.multi.installer*) プラグインをアクティブ化します。
- インスタンスの CA を構成する前に必要な事前作業を理解します。詳細については、「継続認証の事前作業」を参照してください。

手順

1. 移動先 すべて > 継続認証.
2. [ポリシー] タブを選択します。
3. **[New** (新規)] を選択します。
4. フォームの各フィールドに入力します。

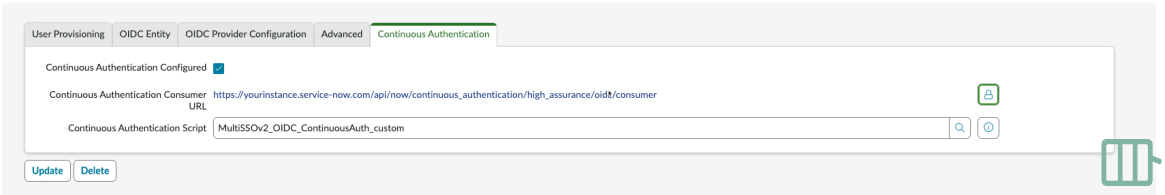
継続認証

フィールド	説明
ポリシー名	ポリシーの名前
説明	ポリシーの一般的な説明
リソースを選択	テーブルを選択します。 注: ○ この例では、[インシデント] テーブルが選択されています。要件に基づいてテーブルをいくつでも選択できます。 ○ メタデータとともに選択されたテーブルにエラーが表示されます。ユーザーの構成アクセスに影響を与える可能性があるため、実際にメタデータテーブルへのアクセスを制限するかどうかを確認する必要があります。 ○ **sys_properties**、**sys_continuous_auth_policy**、**sys_us** ブルは CA から除外され、CA ポリシー構成に追加することはできません。



注: CA ポリシーには、次のいずれかのログイン方法を使用できます。

- **SSO** ベースのログイン:ID プロバイダーレコード内の [継続的認証] タブで フィールドを指定し、ID プロバイダーレコードを アクティブとして設定しま す。

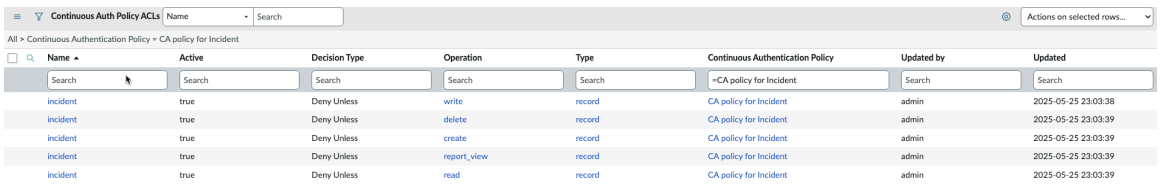


ID プロバイダーの構成の詳細については、「 OIDC と SAML」を参照してください。

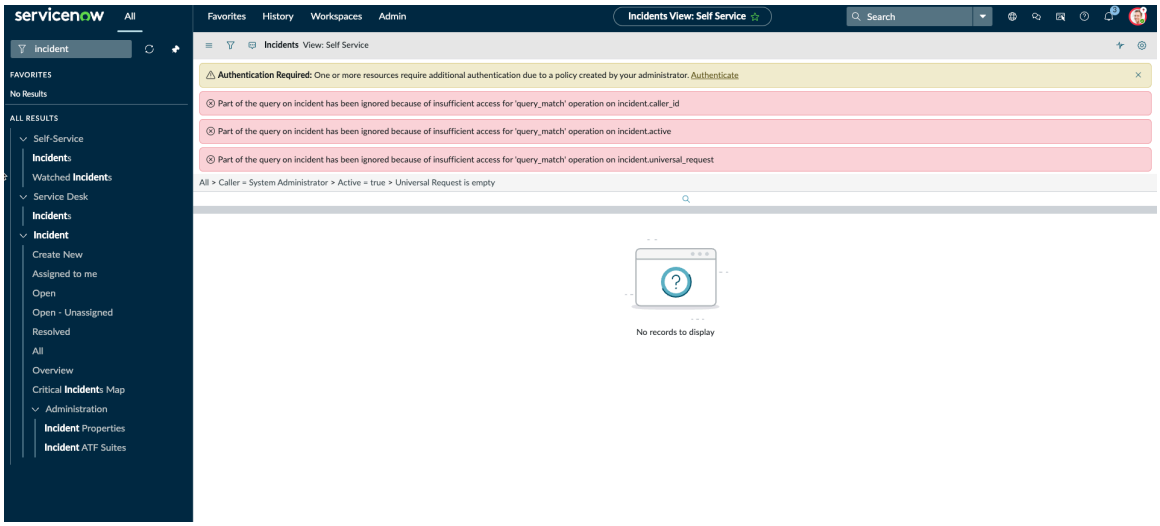
- 非 **SSO** ベースのログイン:デフォルトでは、継続的認証構成の ID プロバイダーがない場合は、ログイン方法としてマルチファクター認証 (MFA) が使用されます。MFA プロパティがアクティブで、要件に基づいて構成されていることを確認します。MFA プロパティの詳細については、「 マルチファクター認証システムプロパティ」を参照してください。

5. [保存してアクティブ化] を選択します。

結果
構成に指定された詳細に基づいて、選択したテーブルまたはデータクラスのアクセス制御リスト (ACL) を使用して CA ポリシーが作成されます。作成された ACL の詳細を表示するには、ポリシーページで **[ACL** の表示] を選択します。

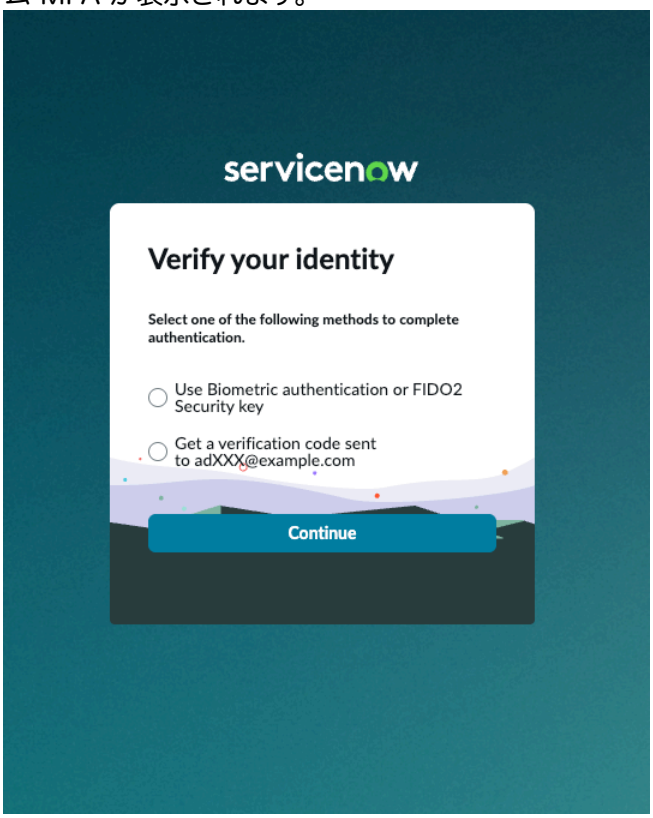


作成された CA ポリシーは、ポリシーを使用して保護したテーブル (この場合 はインシデント テーブル) にアクセスするための認証をユーザーに求めます。ユーザーは [認証] オプションを選択できます。

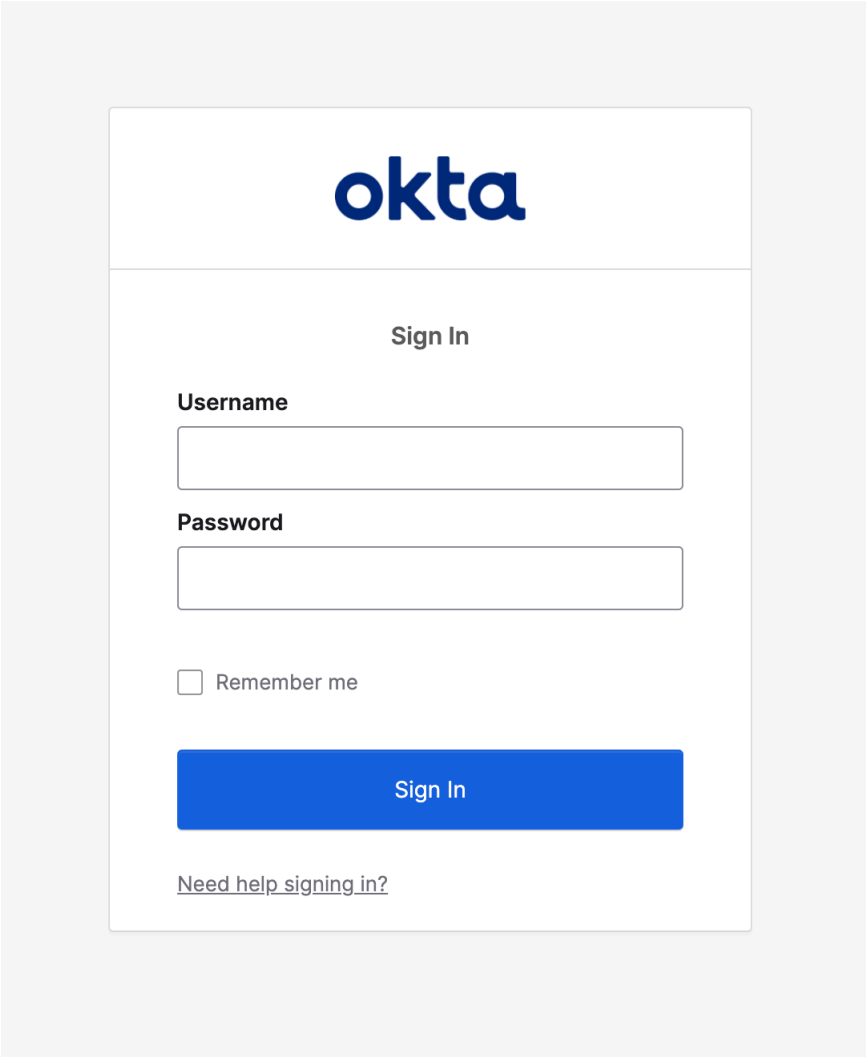


以下に基づいて認証を実行します。

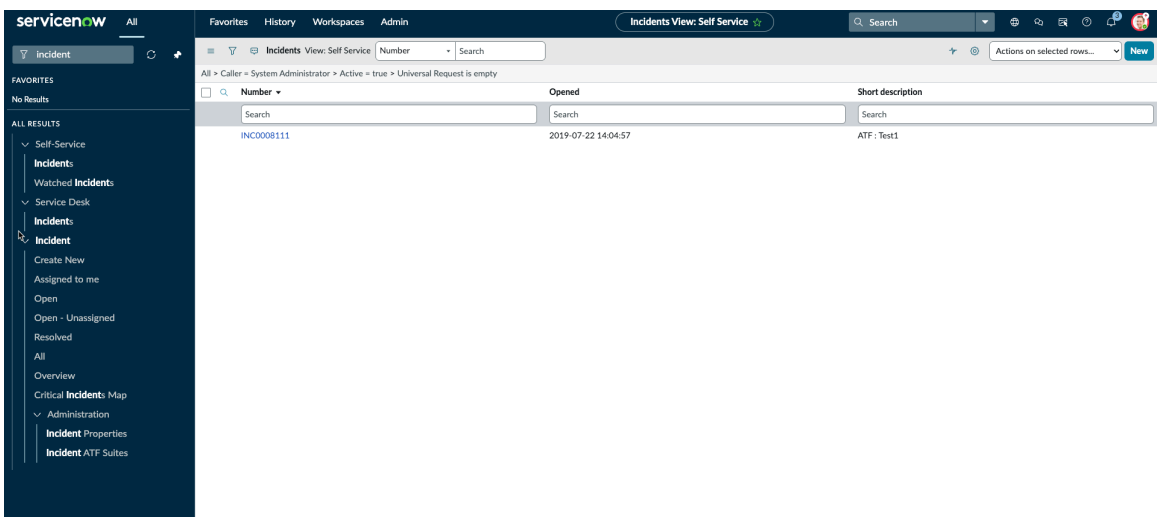
- インスタンスにログインするためにローカルログインを実行したユーザーには、ステップアップ認証用のプラットフォーム MFA が表示されます。



- インスタンスにログインするために SSO ログイン (OIDC または SAML) を実行したユーザーは、再認証のために SSO とともに表示されます。



認証に成功すると、テーブルが表示されます。



これで、ユーザーに対してハイアシュアランスセッションが確立されました。高保証セッションは、高保証セッション長 (*glide.zta.high_assurance.session.timeout*) システムプロパティに

制限されます。ハイアシュアランスセッション時間がプロパティ長を超えると、ユーザーは再認証またはステップアップ認証を求められます。

関連トピック

継続認証の構成

継続認証を使用する高保証セッション

継続的認証の詳細

チュートリアル:データクラスの継続的認証の構成

データクラスの継続的認証ポリシーのエンドツーエンド構成と、構成変更によるユーザーへの影響について説明する手順。

始める前に

- 必要なロール:admin (ca_admin)

  注: ロールを **ca_admin** に昇格させる必要があります。

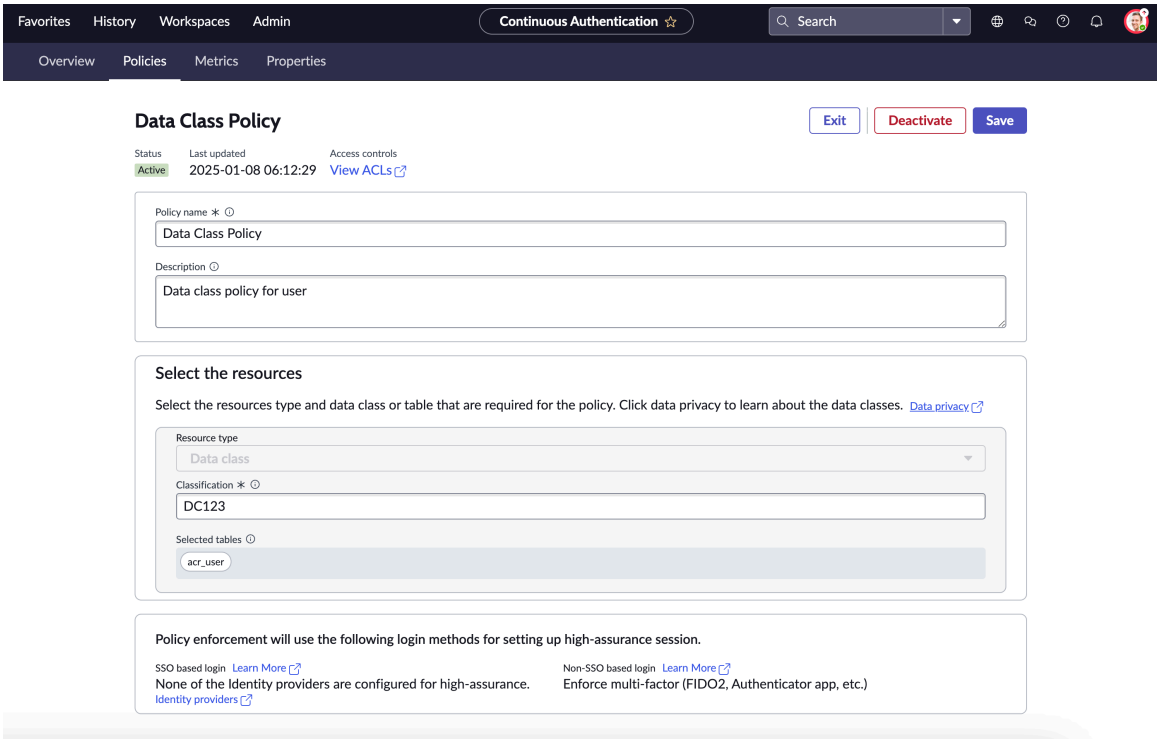
- ライセンスが必要な CA を選択するには、 **Zero Trust - Continuous Authentication** (com.snc.zero_trust_continuous_authentication) をインストールする必要があります。
- 継続的認証 (*glide.zta.continuous_authentication.enabled*) システムプロパティを有効にします。詳細については、「システムプロパティ」を参照してください。
- Integration - Multiple Provider Single Sign-On Installer (*com.snc.integration.sso.multi.installer*) プラグインをアクティブ化します。
- インスタンスの CA を構成する前に必要な事前作業を理解します。詳細については、「継続認証の事前作業」を参照してください。

手順

1. 移動先 すべて > 継続認証.
2. [ポリシー] タブを選択します。
3. **[New** (新規)] を選択します。
4. フォームの各フィールドに入力します。

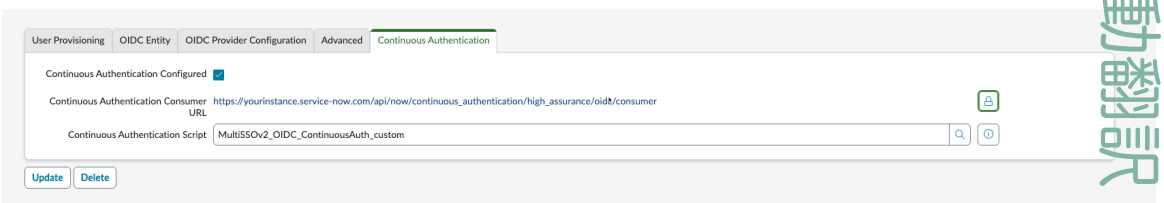
継続認証

フィールド	説明
ポリシー名	ポリシーの名前
説明	ポリシーの一般的な説明
リソースを選択	データクラスを選択します。データクラスを作成し、CA ポリシー構成に使用できます。 注: データクラスの作成方法の詳細については、「 データ分類」を参照してください。



注: CA ポリシーには、次のいずれかのログイン方法を使用できます。

- **SSO** ベースのログイン:ID プロバイダーレコード内の [継続的認証] タブで フィールドを指定し、ID プロバイダーレコードを アクティブとして設定しま す。

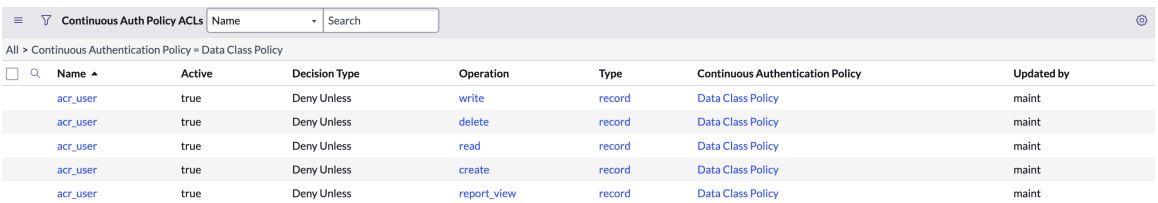


ID プロバイダーの構成の詳細については、「 OIDC と SAML」を参照してください。

- 非 **SSO** ベースのログイン:デフォルトでは、継続的認証構成の ID プロバイダーがない場合は、ログイン方法としてマルチファクター認証 (MFA) が使用されます。MFA プロパティがアクティブで、要件に基づいて構成されていることを確認します。MFA プロパティの詳細については、「 マルチファクター認証システムプロパティ」を参照してください。

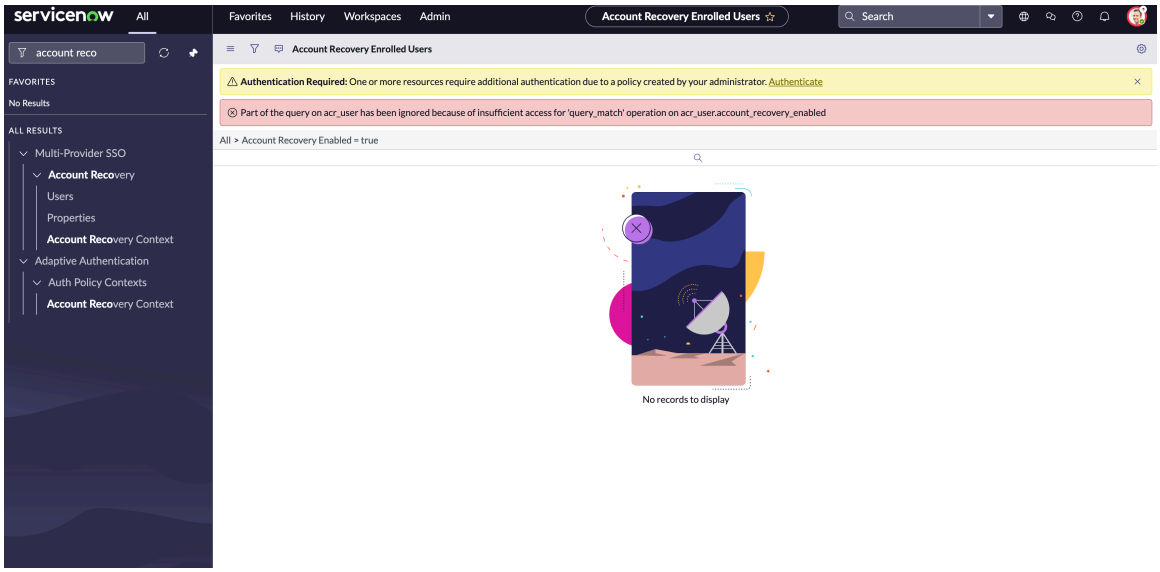
5. [保存してアクティブ化] を選択します。

結果
構成に指定された詳細に基づいて、選択したテーブルまたはデータクラスのアクセス制御リスト (ACL) を使用して CA ポリシーが作成されます。作成された ACL の詳細を表示するには、ポリシーページで **[ACL** の表示] を選択します。



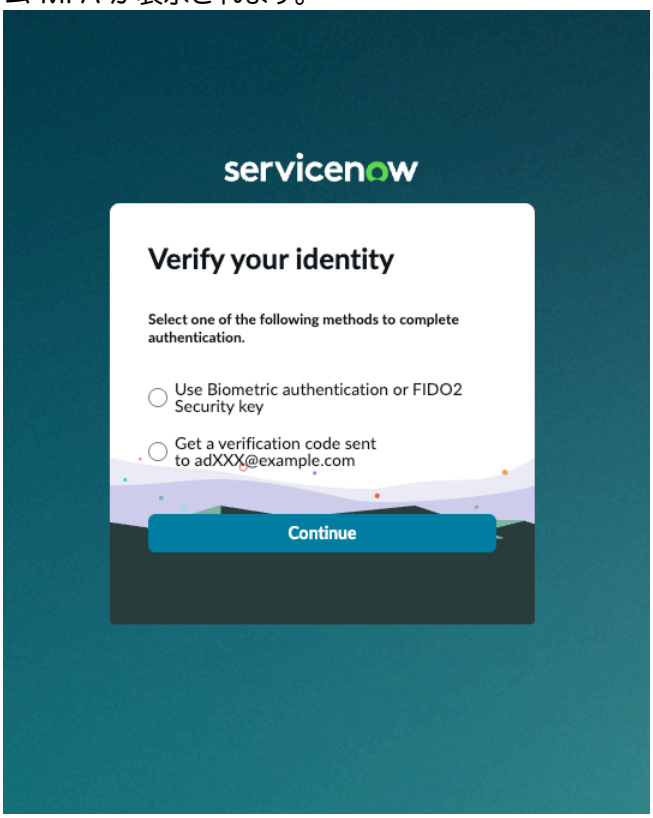
Name	Active	Decision Type	Operation	Type	Continuous Authentication Policy	Updated by
acr_user	true	Deny Unless	write	record	Data Class Policy	maint
acr_user	true	Deny Unless	delete	record	Data Class Policy	maint
acr_user	true	Deny Unless	read	record	Data Class Policy	maint
acr_user	true	Deny Unless	create	record	Data Class Policy	maint
acr_user	true	Deny Unless	report_view	record	Data Class Policy	maint

作成された CA ポリシーは、ポリシーを使用して保護したデータクラス (この場合は アカウント復旧テーブルに設定されたデータクラス) への認証をユーザーに求めます。ユーザーは [認証] オプションを選択できます。



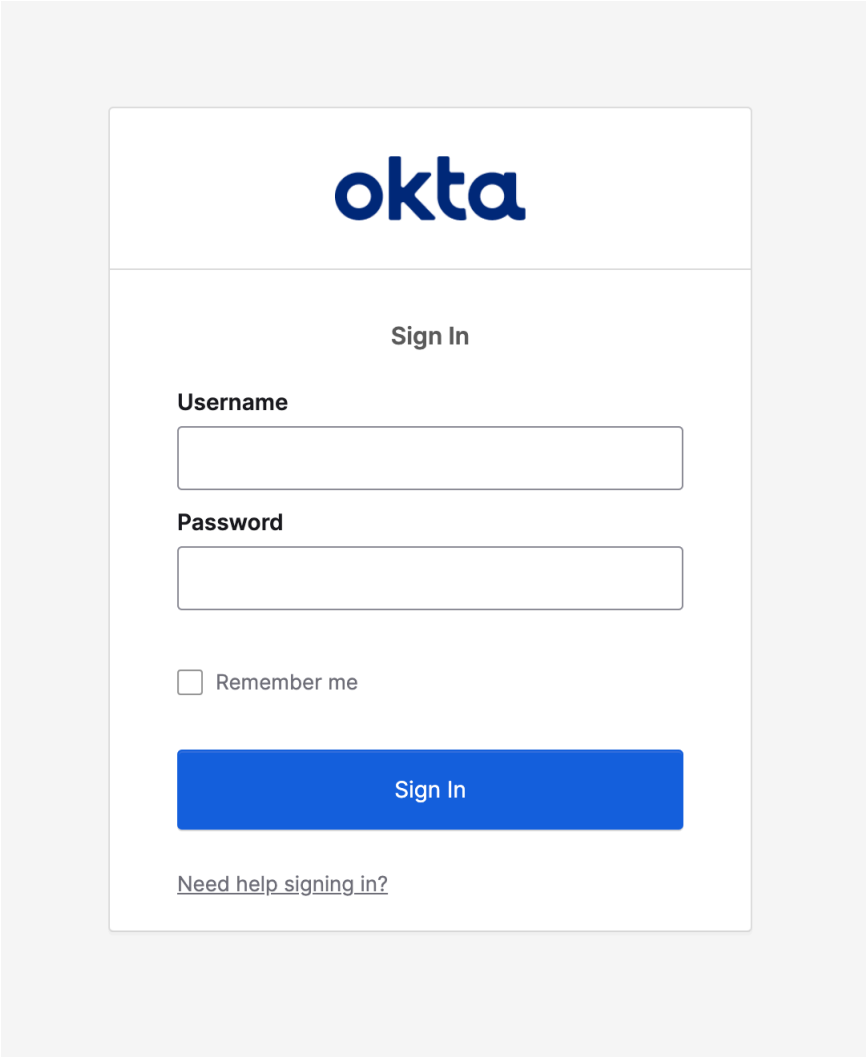
以下に基づいて認証を実行します。

- インスタンスにログインするためにローカルログインを実行したユーザーには、ステップアップ認証用のプラットフォーム MFA が表示されます。

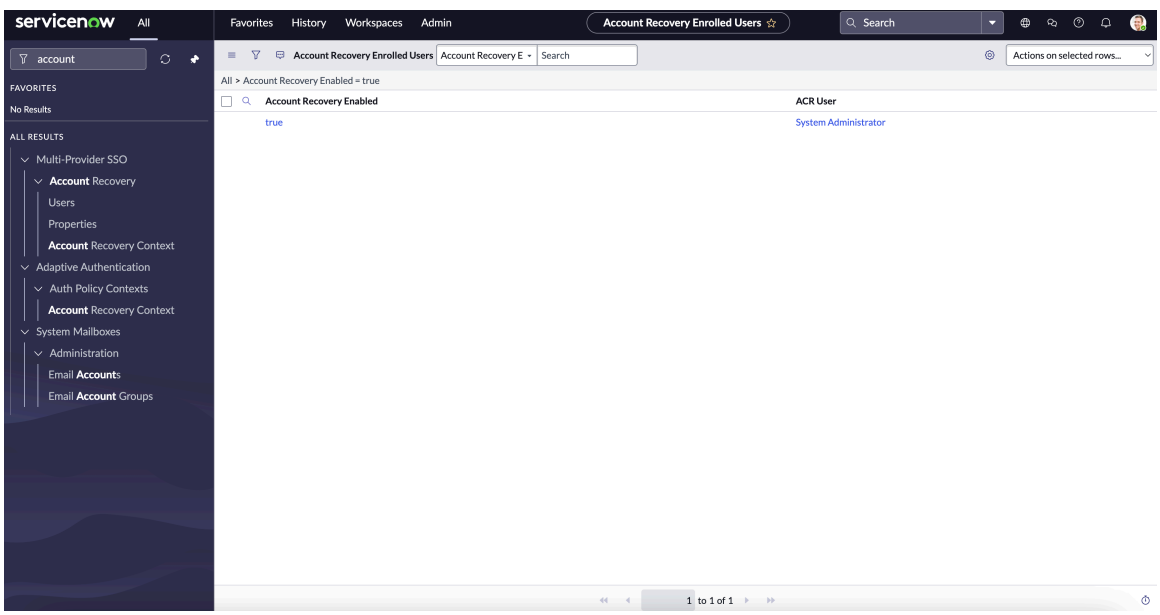


自動翻訳

- インスタンスにログインするために SSO ログイン (OIDC または SAML) を実行したユーザーは、再認証のために SSO とともに表示されます。



認証が成功すると、データクラスを持つテーブルが表示されます。



これで、ユーザーに対してハイアシュアランスセッションが確立されました。高保証セッションは、高保証セッション長 (*glide.zta.high_assurance.session.timeout*) システムプロパティに制限されます。ハイアシュアランスセッション時間がプロパティ長を超えると、ユーザーは再認証またはステップアップ認証を求められます。

関連トピック

- 継続認証の構成
- 継続認証を使用する高保証セッション
- 継続的認証の詳細

継続認証を使用する高保証セッション

ServiceNowの継続的な認証を使用して、ハイアシュアランスセッションを確立します。

高保証セッションは、データにアクセスし、高い信頼性で検証された ID (ユーザー) との安全で信頼できる接続を確立するためのセキュリティ対策です。

ServiceNowの高保証は、ユーザーが機密性の高いデータにアクセスしようとしている間に、マルチファクター認証 (MFA) やシングルサインオン (SSO) などの方法を使用して再認証を強制する堅牢な認証方法によって実現されます。

ユーザーが再認証またはステップアップ認証 (MFA) を実行すると、ハイアシュアランスセッションが確立され、CA ポリシー構成に基づいて CA アドミニストレーターによって保護されているデータにユーザーがアクセスできるようになります。

ログインのタイプに基づいてハイアシュアランスを確立するために使用される再認証方法は次のとおりです。

- SSO ログインのハイアシュアランス
- 非 SSO ログインのハイアシュアランス

ユーザーによって作成された高保証セッションは、CA アドミニストレーターによって決定された高保証セッション長 (*glide.zta.high_assurance.session.timeout*) に基づいて有効です。

高保証セッションは、 高保証 システムプロパティを設定することで、要件に基づいてカスタマイズできます。

高保証システムプロパティ

フィールド	説明
ハイアシュアランスセッションの長さ (*glide.zta.high_assurance.session.timeout*)	エンドユーザーが再認証する必要があるハイアシュアランスセッションの長さを指定します。デフォルト:30 分。 注: 値は 1 ～ 480 の範囲でなければなりません。
ログイン時のデフォルトのハイアシュアランスセッション長	ユーザーログイン時のデフォルトのハイアシュアランスセッション長の期間を分単位で指定します。デフォルト値: **5** 分。 注: このプロパティは、SSO 以外のログインにのみ適用されます。

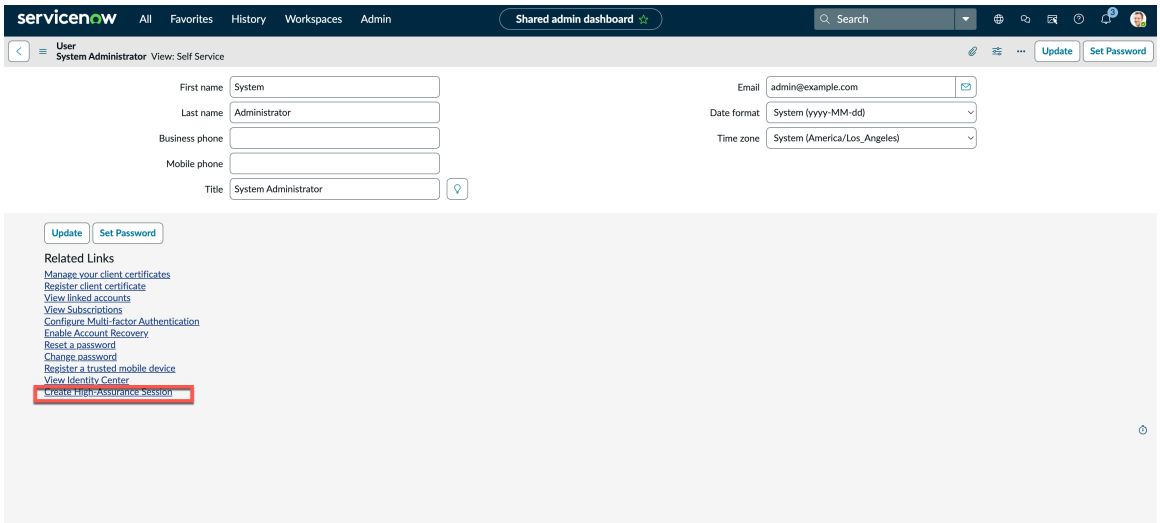
高保証システムプロパティ (続く)

フィールド	説明
エンドユーザー表示メッセージの構成 (*glide.zta.high_assurance.session.message*)	再認証のためにエンドユーザーに表示するメッセージを指定します。デフォルトメッセージ: アドミニストレーターが作成したポリシーにより、1 つ以上のリソースに追加認証が必要です。
ユーザーアカウントのロックアウト前の認証失敗の合計時間 (*glide.zta.high_assurance.session.max.login.failed.attempts*)	ユーザーがログアウトされるまでの認証試行失敗の最大回数を設定します。 **注:** 値は 3 から 10 の範囲でなければなりません。

先制措置としての高保証セッション

金融取引、政府情報、PII などの高権限データを扱うユーザーは、ログインセッション中に頻繁に認証通知が行われないようにするための予防策として、高保証セッションを確立できます。

高保証セッションは単独で作成できます。ハイアシュアランスセッションを作成するには、次を選択します ユーザープロファイル > プロファイル. [関連リンク] セクションで、[ハイアシュアランスセッションの作成] を選択します。本人確認を行い、ハイアシュアランスセッションを作成します。



関連トピック

- 継続的認証の詳細
- 継続認証の事前作業
- 継続認証の構成

SSO ログインのハイアシュアランス

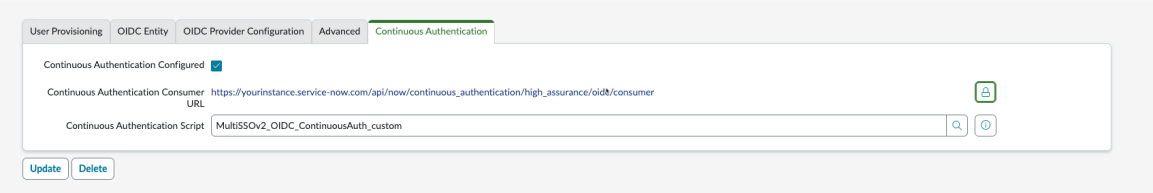
ServiceNow の継続的認証を使用して、SSO ログインのハイアシュアランスセッションを確立します。

ハイアシュアランスセッションは、ユーザーが ID を検証し、特定の期間、特定の ID または ID プロバイダーで認証することを要求するセッションです。

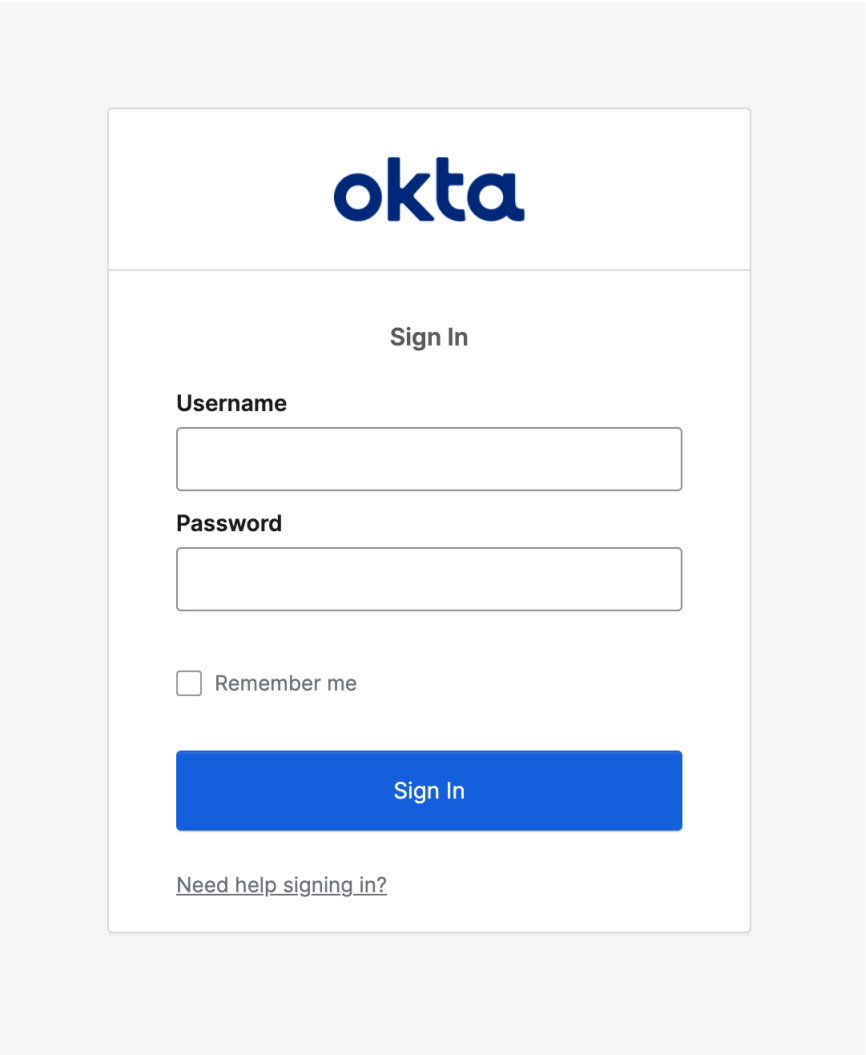
ServiceNowの継続的認証 (CA) 機能を使用すると、個人識別可能情報 (PII) や機密データにアクセスするユーザーに対してハイアシュアランスセッションを作成したり、保護する明示的なデータへのアクセスを制限したりするポリシーを作成できます。

ユーザーが再認証されると、ハイアシュアランスセッションが確立され、CA ポリシー構成に基づいて CA アドミニストレーターによって保護されたデータにユーザーがアクセスできるようになります。

CA ポリシーを作成して、ユーザーの ID を確認し、保護したデータにアクセスするためのユーザーを認証できます。CA ポリシーを設定し、ID プロバイダーの詳細を ID プロバイダーレコードに指定して、ハイアシュアランスセッションを確立できます。



SSO ベースのログイン (SAML または OIDC) を実行しているユーザーは、保護されたデータへのアクセスが試行されるたびに、最初のログイン時にユーザーが使用したものと同じ SSO で再認証画面が表示されます。



自動翻訳

SSO 認証が成功すると、保護されたデータが特定の期間ユーザーに表示されます。要件に基づいて時間制限を変更するようにプロパティを設定できます。詳細については、「継続認証を使用する高保証セッション」を参照してください。

SSO ログイン (SAML または OIDC) で再認証を実行すると、保護されたデータにアクセスしている ID (ユーザー) との安全で信頼できる接続を確立する高保証セッションが作成されます。

ユーザーに対して確立されるハイアシュアランスセッションは、ハイアシュアランスセッション長 (*glide.zta.high_assurance.session.timeout*) システムプロパティに制限されます。ハイアシュアランスセッション時間がプロパティ長を超えると、ユーザーは再認証またはステップアップ認証を求められます。

関連トピック

- 継続認証を使用する高保証セッション
- 継続認証の事前作業
- 継続認証の構成

非 SSO ログインのハイアシュアランスセッション

ServiceNow の継続的な認証を使用して、非 SSO ログイン (ローカルまたは LDAP) のハイアシュアランスセッションを確立します。

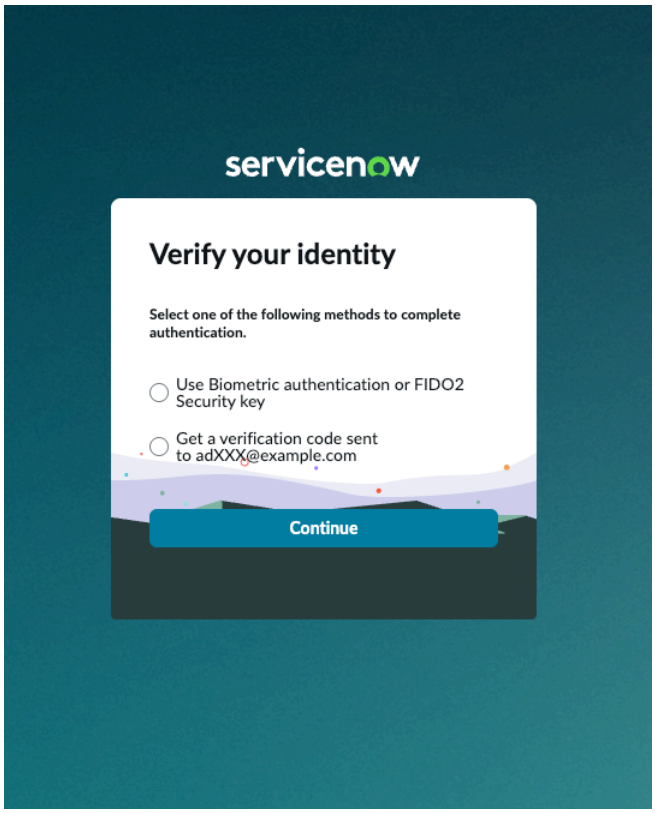
ハイアシュアランスセッションは、ユーザーが ID を検証し、特定の期間、特定の ID または ID プロバイダーで認証することを要求するセッションです。

ServiceNowの継続的認証 (CA) 機能を使用すると、個人識別可能情報 (PII) や機密データにアクセスするユーザーに対してハイアシュアランスセッションを作成したり、保護する明示的なデータへのアクセスを制限したりするポリシーを作成できます。

ユーザーがステップアップ認証 (MFA) を実行すると、ハイアシュアランスセッションが確立され、CA ポリシー構成に基づいて CA アドミニストレーターによって保護されたデータにユーザーがアクセスできるようになります。

CA ポリシーを作成して、ユーザーの ID を確認し、保護したデータにアクセスするためのユーザーを認証できます。非 SSO ベースのログイン (ローカルまたは LDAP) を実行しているユーザーで、保護されたデータへのアクセスが試行されるたびに、ステップアップ認証 (MFA) 画面がユーザーに表示されます。

自動翻訳



認証が成功すると、保護されたデータが特定の期間ユーザーに表示されます。要件に基づいて時間制限を変更するようにプロパティを設定できます。詳細については、「継続認証を使用する高保証セッション」を参照してください。

注: ユーザーが MFA をセットアップしていない場合は、セットアップを完了する必要があります。

ステップアップ認証 (MFA) を実行すると、保護されたデータにアクセスしている ID (ユーザー) との安全で信頼できる接続を確立するハイアシュアランスセッションが作成されます。

ユーザーに対して確立されるハイアシュアランスセッションは、ハイアシュアランスセッション長 (*glide.zta.high_assurance.session.timeout*) システムプロパティに制限されます。ハイアシュアランスセッション時間がプロパティ長を超えると、ユーザーは再認証またはステップアップ認証を求められます。

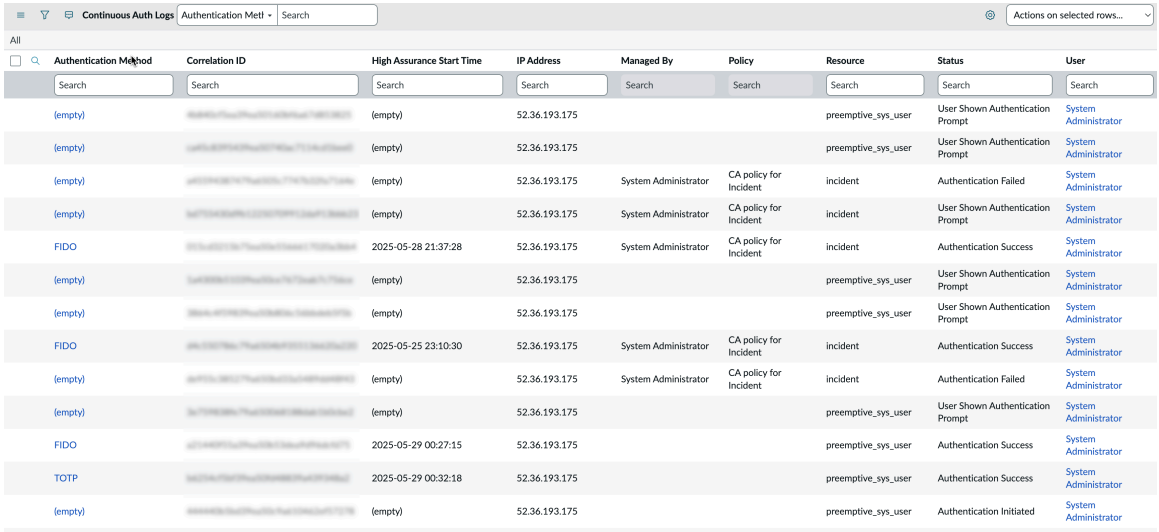
関連トピック

- 継続認証を使用する高保証セッション
- 継続認証の事前作業
- 継続認証の構成

継続認証監査ログ

継続的認証 (CA) ログの詳細について説明します。CA ログには、ユーザーが実行したすべての認証試行が表示されます。

CA 関連のログ監査情報は、 **continuous_auth_log** で入手できます。LIST ページ。「continuous_auth_log」と入力できます。CA ログを表示するためのナビゲーションの LIST。



ログページには、認証方法、相関 ID、およびユーザーのセッションに関連するその他の詳細に関する情報が表示されます。

サービスプロバイダーのドメインセパレーション

ServiceNow プラットフォームを使用すると、サービスプロバイダー (SP) は、顧客に迅速なオンボーディングを提供し、コンプライアンスを満たし、ドメインセパレーションを使用してデータを保護できます。クライアントデータ、プロセス、およびレポートをドメインと呼ばれる論理グループに分けることができます。SP は、誰がどのコンテンツを表示してアクセスするかをコントロールします。

自動翻訳

探索

ドメインセパレーションについて学習します。

構成

ドメインセパレーションを構成します。

参照

ドメインセパレーションの詳細を取得します。

分析

ドメインセパレーションの分析方法の詳細を表示

ドメインセパレーションの概要

ドメインセパレーションを使用すると、データ、プロセス、管理タスクを論理的に定義されるドメインに分けることができます。

ドメインセパレーションは、以下の顧客にとって最適です。

- ビジネスエンティティ間で絶対的なデータ分離を適用する必要がある (データ分離)。
- 各ドメインのビジネスプロセス定義とユーザーインターフェイスをカスタマイズする (委任管理)。
- 単一のインスタンスでグローバルなプロセスとグローバルなレポートを維持する。
- サービスプロバイダー、顧客、パートナー、またはサブ組織間でデータを分離する。
- 顧客間で軽微なまたは中程度のプロセスの違いがある。

個別のインスタンスとドメインセパレーションの比較

ドメインセパレーションがマルチテナントのサポートを提供している間は、マルチテナントはまだ単一のインスタンス内に含まれています。一部のグローバルなプロパティ、データ、およびプロセスは、すべてのドメインで共有されます。たとえば、システムのログインページの [記憶する] はグローバルであり、ドメインごとに指定できません。

すべてのシステムのプロパティを完全に分離する必要があり、グローバルレポートまたはグローバルプロセスを必要としない場合、インスタンスの分離が最適なオプションです。

データ分離

ドメインのメンバーは、そのドメインまたはドメイン階層の下位にある子ドメインに含まれるデータのみを表示できます。アドミニストレーターが特定のドメインにアサインした場合を除き、デフォルトで、すべてのユーザーとすべてのレコードがグローバルドメインのメンバーになります。ユーザーまたはレコードをドメインにアサインすると、そのインスタンスはユーザーのドメインとレコードのドメインを比較して、そのユーザーがそのレコードを表示できるかどうかを判断します。

ServiceNow アプリケーションは、次の増分サポートレベルで定義されています。これらのレベルは、実際のユースケースとペルソナの観点に基づいています。

データ分離：テナントには、表示権限のあるデータのみが表示されます。テナントには、他のテナントデータへのアクセス権を付与できますが、アクセス権がない場合、テナントデータを照会することはできません。

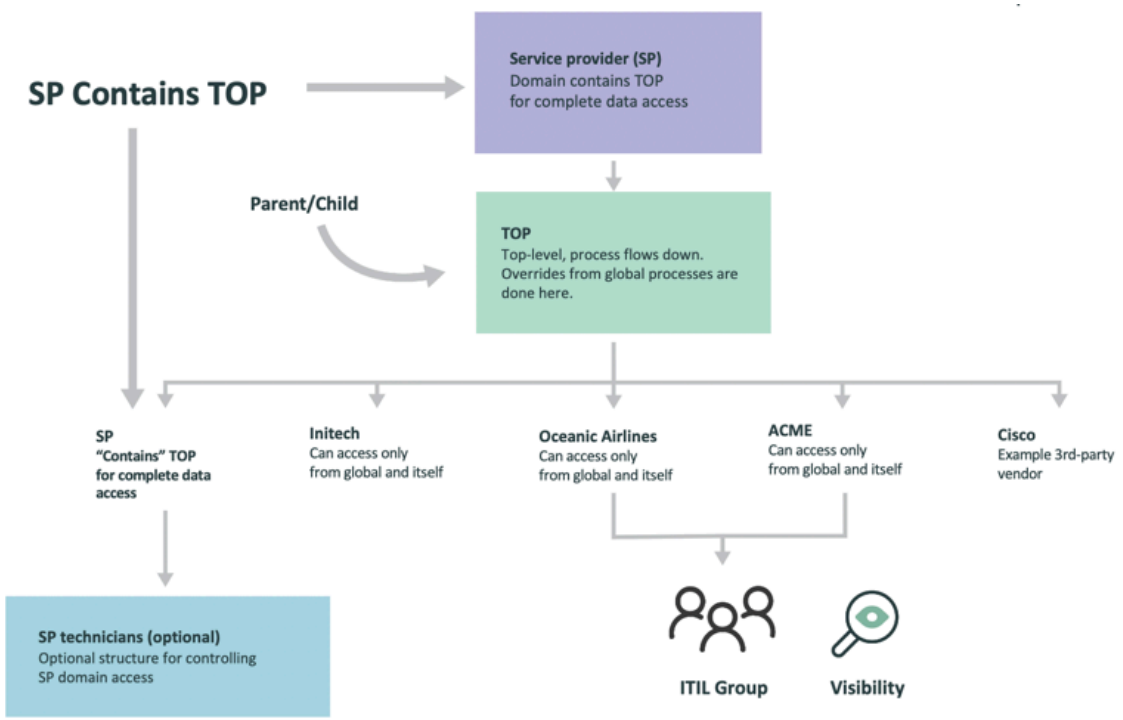
UI 分離：ビュー、リスト、ラベルといった UI 要素のテナント固有のエクスペリエンスをサポートします。

ビジネスロジック分離：メール通知、ビジネスルール、クライアントスクリプト、UI ポリシー、UI アクションなど、テナント固有のシステムポリシーを作成できます。

階層モデル：ネストされたマルチテナントであるため、親テナントが子テナントのリソースにアクセスできます。親テナントのビジネスロジックは子テナントに対して自動的に実行され、任意のレベルで上書きできます。

クロステナントインテリジェンス (ドメインスコープ)：追加のテナントデータにアクセスできるテナントのデータ、メタデータ、ビジネスロジック、および処理コンテキストを自動的に処理します。

一般的に、ドメイン階層の上位レベルで定義されたデータは、階層の下位レベルでは表示されません。

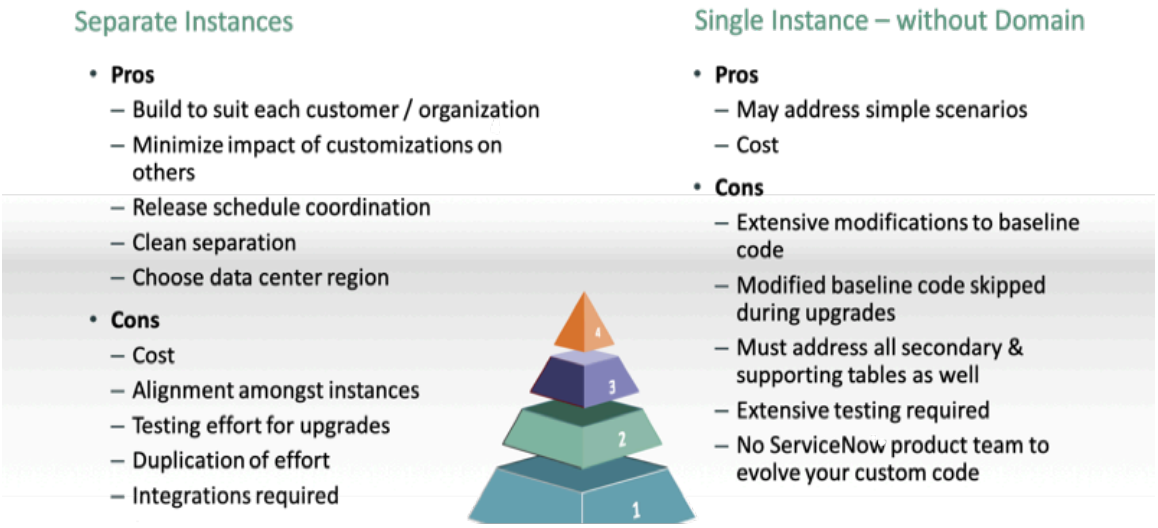


ドメインパスの移行

ドメインパスはすべての顧客に使用されます。ドメインの番号付けは使用されません。カスタマーサービス & サポートがアップグレードのサポートを行います。

ドメインセパレーションの代替手段

個別のインスタンスは、ドメインセパレーションの一般的な代替手段です。これにより、他の人にほとんど影響を与えることなく、顧客やステークホルダーの要件を満たすための柔軟性が高まります。



⚠ **警告:** ドメインセパレーションをアクティブ化する前に、担当者に問い合わせて、お使いの環境に適していることを確認してください。ドメインセパレーションは、一定レベルの管理オーバーヘッドを追加します。無効にすることはできますが、インスタンスから削除することはできません。

関連トピック

- サービスプロバイダー向けのドメインセパレーションの推奨プラクティス
- ドメインセパレーションプラグイン

内部や外部の顧客に委任できる構成

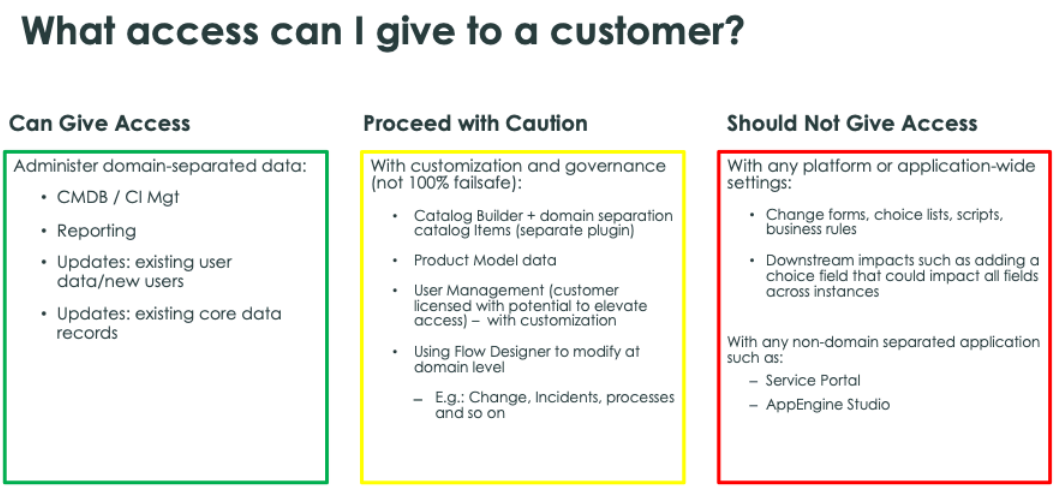
ドメインセパレーションは、ServiceNow® サービスプロバイダー (SP) が顧客に提供するサービスを設定できるように設計されています。このトピックで詳しく説明されているいくつかのエリアを除き、顧客がこれらのサービスを自分で管理できるように設計されていません。

概要

SP の顧客は、ライセンスや他の顧客に影響を与えないドメイン内のデータを自分で安全に管理することができます。たとえば、顧客が新しいレポートを作成したり構成アイテムを管理したりすることは安全ですが、フィールド、選択肢、ビジネスルール、および同じインスタンス上の他の顧客に影響を与える可能性のあるその他のプロセスをカスタマイズすることは安全ではありません。

ServiceNow プラットフォームの ServiceNow ベースシステム管理ロールとそのアクセス制御は、インスタンスごとに 1 人のアドミンチーム用に設計されています。たとえば、インスタンスのすべてのドメイン設定を管理し、新しいドメインを作成するために、domain_admin ロールが SP のリソースの 1 つに付与されます。ドメイン固有の管理タスクで SP は、顧客に特定のアクセス権を付与するために、必要に応じて新しい「顧客 admin」ロールとアクセス制御を作成する必要があります。

次の画像は、顧客が安全に実行できるさまざまなカテゴリの一般的なアドミン機能の概要で



す。

🟢 アクセス権を付与可能

例：

- CMDB での CI データ管理
- レポートの作成
- 既存のユーザーデータの更新、またはロールのない新規ユーザー
- 部門、グループ、場所、コストセンター、ロールのない新しいグループ、新しい部門/コストセンター/場所などの既存のコアデータレコードを更新します。

🟡 注意して続行

例：

- カタログアイテム：顧客が更新できる顧客固有のカタログアイテムを作成するために、次の 2 つの機能を併用できます。カタログアイテムのドメインセパレーション (ドメインセパレーションとサービスカタログ) を使用することで、インスタンスオーナーは顧客のドメイン内にアイテムを作成することができます。インスタンスオーナーは、顧客が価格、説明、画像などの安全なフィールドを更新できるようにするロールを作成できます。カタログビルダー (Quebec リリースの新機能) を使用すると、SP アドミンチームは、規範的な UI エクスペリエンス内から顧客のドメインで新しいアイテムを作成するために、顧客に安全に配布できるアイテムテンプレートを作成できます。
- ユーザー/グループ管理：ユーザーレコードの作成と変更ができる「顧客 admin」ロールを安全に作成できますが、ロールの追加や削除はセキュリティとライセンスに影響を与える可能性があります。ベースシステムでは、顧客が安全に付与できるロールを細分化することはできません。グループの作成と変更についても同様です。グループ自体を変更することはできますが、ロールの追加や削除はコントロールする必要があります。
- フローデザイナー ：ServiceNow フローデザイナー は、テーブルのプロセス (ワークフロー) を作成するために使用するビルドツールです。flow_designer ロールで、顧客はスクリプトなしにフローをビルドできます。階層内で上位のドメイン内のすべてのフローを読み取ってクローンを作成できます。ドメイン内でフローの作成と変更ができます。ただし、これはサイロでは行えません。プロセスが互いにキャンセルし合ったり、他の競合が発生したりしないように、プロセスに影響を与える可能性のあるユーザーをグローバルアドミンチームに追加する必要があります。

アクセス権を付与しない

選択フィールドの仕組みを理解することは、SP アドミンチームのみが選択フィールドを管理する必要がある理由を理解するのに役立ちます。

- 選択フィールドの構造：選択フィールドの値は sys_choice テーブルに格納され、テーブル、ドメイン、および言語に基づいてグループ化されます。

  たとえば、タスクの [ステータス] フィールドは、タスクを拡張するすべてのテーブルで使用できます。したがって、各テーブルが独自の値を持ち、それらの値にドメイン数を乗じ、ドメイン値に言語数を乗じることができます。

  すべてのテーブル、ドメイン、および言語の [ステータス] のすべての値が [ステータス] フィールドの値と見なされます。

- 選択フィールドの変更の仕組み： 選択フィールドが更新されると、そのフィールドのすべての値 (テーブル、ドメイン、言語) を含むペイロードが作成されます。このペイロードをインスタンスにインストールすると、フィールドの既存の値がすべて削除され、新しい値がロードされます。これにより、重複するエントリや残っていた無効な値がなくなります。

  そのため、ドメインセパレーションされたインスタンスで選択フィールドを直接更新する機能を顧客に提供することはできません。インスタンス全体に影響してしまうためです。また、選択フィールドに影響するインポートされた更新セットにより既存の選択肢が上書きされるため、本番インスタンスで選択肢を直接更新することもできません。場合により、選択フィールドでプロセス自体が実行される可能性があります。このプロセスは、顧客が選択フィールドを変更しようとすると中断されます。

詳細については、以下を参照してください。

- ユーザー管理の詳細
- ACL ルールを作成する
- ServiceNow University のサービスプロバイダーラーニングパス
- サービスプロバイダーのドメインセパレーション
- サービスプロバイダーの概念
- ドメインセパレーションのアプリケーションサポート
- ドメインセパレーションのリリースノート

ドメインアサイン

デフォルトでは、ドメインセパレーションは、テーブルとその拡張にドメインフィールドを追加します。

sys_domain フィールドをテーブルの辞書定義に追加することで、作成する任意の新しいテーブルにドメインセパレーションを拡張することもできます。デフォルトでは、システム専用ドメインは、適切な場合に、プラットフォームとベースラインアプリケーションテーブルを分離します。

⚠ 警告: ServiceNow では、予期しない結果が生じる可能性があるため、ドメインセパレーションプラットフォームテーブル (辞書エントリ [sys_dictionary] や辞書エントリの上書き [sys_dictionary_override] テーブルなどの sys_prefix を持つテーブル) は推奨されません。

各レコードには単一のドメインが割り当てられます。 そのドメインは **sys_domain**フィールドに格納されます。 いくつかのテーブルには、デフォルトで **sys_domain** 列があり、既にドメインセパレーションされています。

sys_domain フィールドの値には、次のいずれかでレコードに割り当てられたドメインが含まれます。

- ユーザーが所属する会社
- レコード作成時のビジネスルール
- レコードの作成時に使用されるモジュール
- レコードの作成時に使用されるフォームテンプレート
- 親レコードのドメイン
- ユーザーレコードに割り当てられたドメイン
- 作成したユーザーのドメイン

次のテーブルはドメインセパレーションされません。

- アクセス制御 [sys_security_acl]
- スクリプトインクルード [sys_script_include]
- システムのプロパティ [sys_properties]
- セキュリティ除外/包含リストエンティティ [sys_security_restricted_list]
- 辞書エントリ [sys_dictionary]
- 辞書エントリの上書き [sys_dictionary_override]

会社へのユーザー割り当て

アドミニストレーターは、ユーザーを会社に割り当てることで、ユーザーをドメインにすばやく割り当てることができます。ユーザーがドメインに割り当てられると、レコードはユーザーのドメインを自動的に継承します。

たとえば、Bow Ruggeri を ACME 社に割り当てると、そのユーザーは自動的に ACME ドメインに割り当てられます。Don Goodliffe を Initech 社に割り当てると、そのユーザーは自動的に Initech ドメインに割り当てられます。作成したレコードは自動的に適切なドメインに追加されます。

ビジネスルールを使用したドメインの割り当て

アドミニストレーターは、ビジネスルールを使用して、レコードの作成時にドメイン値を自動的に設定できます。ビジネスルールは **[sys_domain]** フィールドに値を設定する必要があります。アドミニストレーターは、レコードのテーブルで **[sys_domain]** 列が使用可能であることを確認する必要があります。詳細については、「サービスプロバイダー向けのドメインセパレーションの推奨プラクティス」を参照してください。

モジュールを使用してドメインを割り当てる

アドミニストレーターは、**sysparm_domain** URL パラメーターを使用して、モジュールから特定のドメインに新しいレコードを自動的に割り当てることができます。アドミニストレーターは、**Argument** の値が sysparm_domain=ドメインのsys_ID であるモジュールを作成する必要があります。

フォームテンプレートを使用してドメインを割り当てる

アドミニストレーターは、フォームテンプレートを使用して、特定のドメインに新しいレコードを自動的に割り当てることができます。アドミニストレーターは、フォームに **sys_domain** フィールドを追加し、ドメイン値を選択する必要があります。たとえば、**sys_domain** フィールドを **TOP/**

ACME ドメイン に設定すると、このテンプレートのすべてのレコードが TOP/ACME ドメインに自動的に割り当てられます。

テーブルのドメイン継承

デフォルトでは、関連レコードは親レコードのドメインを継承します。例：

- 変更タスクレコードは、親変更要求レコードのドメインを継承します。
- 問題レコードは、親インシデントレコードのドメインを継承します。

ユーザードメインに基づく自動ドメインアサイン

他のドメイン条件が適用されない場合、レコードはそれを作成したユーザーのドメインを自動的に継承します。

ヴィジビリティドメインと包含ドメイン

ヴィジビリティドメインは、特定のユーザーまたはユーザーのグループが表示できる内容をコントロールします。「包含」ドメインは、ユーザーのドメイン全体で表示可能な内容をコントロールします。

ヴィジビリティドメイン

「ヴィジビリティドメイン」要素は、あるドメインのユーザーが別のドメインのレコードにアクセスできるかどうかを決定します。アクセスする別のドメインのレコードの関連リストにあるユーザー [sys_user] とグループ [sys_user_group] レコードにヴィジビリティドメイン要素を関連付けます。グループは、グループのヴィジビリティドメインをメンバーに付与します。ユーザーがグループを離れると、グループのヴィジビリティドメインは失われます。ユーザーにヴィジビリティドメインを付与すると、ACL (アクセス制御リスト) ルールに基づいて、そのドメイン内のレコードにすべての権限が付与されます。

ヴィジビリティドメイン：

- ユーザーとドメインの関係であり、明示的に付与されます。
- 子ドメインではありません。
- ドメインピッカーでの選択でコントロールされません。ヴィジビリティドメインへのアクセス権を持つユーザーには、常にそのドメインとその子ドメイン内のデータが表示されます。

注: ヴィジビリティドメインを過度に使用することはお勧めしません。可視化はユーザーがレコードにアクセスできるようにする 1 つの方法ですが、より強力にコントロールするには包含ドメインをお勧めします。

包含ドメイン

通常、親子関係はドメイン階層を定義します。包含ドメインでは、親子関係に関係なく、必要に応じてドメインを関連付けることができます。ただし、包含ドメインはドメインデータにのみ可視化を付与します。プロセスは包含関係の影響を受けません。

包含ドメイン：

- ドメイン間の多対多の関係です。
- 子ドメインがある場合があります。ドメインを選択すると、そのドメインとその子のデータを表示できます。
- ドメインピッカーでの選択でコントロールされます。

注: ドメインレコードを開くと、スコープがそのレコードのドメインに設定されるため、子ドメインのみが表示されます。メニューから [ドメインスコープの切り替え] を選択して、関連リストに入力します。

ドメインの例を含める

ユーザーのホームドメインが A で、A ドメインにドメイン B と C が含まれている場合、それらはすべてピアドメインになります。これは、ユーザーがホームドメイン A にいるときにドメイン A、B、および C のデータが表示されることを意味します。ユーザーがドメインピッカーを使用してドメイン B に変更すると、ドメイン B のデータのみが表示されます。ユーザーがドメイン B またはドメイン C のレコードを直接操作すると、そのドメインのデータのみが表示されます。

ヴィジビリティドメインの例

ドメインの可視化を使用しているときに、Don Goodliffe がデータベースドメインに、Bow Ruggeri がネットワークドメインにいて、グローバルドメインにインシデントがない場合、データ分離により Don は Bow のインシデントにアクセスできません。

グループメンバーシップに基づくヴィジビリティドメインの継承

ドメインテーブルをグループ [sys_user_group] テーブルに設定すると、ユーザーはグループメンバーシップに基づいてヴィジビリティドメインを継承できます。

関連トピック

- 包含クエリーとドメインアクセス
- サービスプロバイダー向けのドメインセパレーションの推奨プラクティス

ドメインスコープ

ドメインスコープは、ユーザーがアクセスできるものとできないものを定義します。

ドメインセパレーションされたインスタンスでセッションを確立する場合、すべてのユーザーに 2 つのドメインスコープがあります。

- セッションスコープは、セッションの確立時にユーザーのユーザーレコードにリストされているドメインに設定されます。ユーザーは、ドメインピッカーからセッションのドメインスコープを手動で変更できます。
- レコードスコープはレコードのドメインを使用し、任意のレコードのフォームを表示すると有効になります。

デフォルトでは、レコードスコープがセッションスコープよりも優先されるため、上位レベルのドメインのユーザーは各レコードのデータとプロセスの制約に従います。ただし、これらのユーザーは、ドメインスコープを展開するか折りたたんで、他のドメインからのデータを表示または非表示にすることができます。たとえば、サービスプロバイダー (SP) ドメイン内のユーザーは、ACME ドメインなどの子ドメインへの可視化も持っています。ACME ドメインからのインシデントレコードを調べる場合、ユーザーはドメインスコープを展開して SP ドメインの値を表示するか、ドメインスコープを折りたたんでレコードの ACME ドメインに一致するレコード値のみを表示するかを選択できます。

注: ユーザーは、ドメインの可視化によって明示的に付与されたドメインのデータに常にアクセスできます。

domain_expand_scope ユーザーロールを持つユーザーは、フォームの [ドメインスコープの切り替え] UI アクションからドメインスコープを選択できます。レコードスコープが有効になると、UI アクションをクリックしてセッションスコープを拡大し、ユーザーのドメインと子ドメインに基づいて

使用可能なすべてのデータを表示できます。セッションスコープが有効な場合は、UI アクションをクリックしてレコードスコープを折りたたみ、現在のレコードのドメインと一致するデータのみを表示できます。

> **注:** レコードがグローバルドメイン内にある場合、またはユーザーのドメインがレコードのドメインと一致する場合、レコードにはドメインスコープを切り替えるための UI アクションは表示されません。

他のドメインからのレコード値の選択

複数のドメインを表示できるユーザーには、レコードのドメインとは異なるドメインからレコード値を選択するオプションがあります。

たとえば、サービスプロバイダーのサービスデスク担当者は、顧客に代わって問題を解決するために、特定のインシデントを自分自身に割り当てることができます。これを行うと、インシデントレコード自体が ACME などの子ドメインに関連付けられていても、インシデントの [アサイン先] フィールドに SP ドメインのユーザーが含まれる可能性があります。

別のドメインからレコード値を選択しても、レコードのドメインは変更されません。レコードは元のドメインを保持します。ユーザーが複数のドメインの値が含まれるレコードを表示する場合、表示される内容はユーザーのドメインの可視化によって決まります。

レコードの値の選択

これらの条件が満たされている場合	ユーザーはこれらの **UI** 要素にアクセスできます
ユーザーは、フィールドで参照されている現在のレコードのドメインにアクセスできます。	ユーザーは次のことができます。 • 参照フィールドには、表示値が表示されます。たとえば、[アサイン先] フィールドにユーザー名が表示されます。 • 参照アイコンから関連レコードが表示されます。たとえば、[アサイン先] フィールドでユーザーのユーザーレコードが表示されます。 • 表示されているドメインから値を選択します。たとえば、SP ドメインと ACME ドメインのいずれかからユーザーを選択できます。
ユーザーは、フィールドで参照されている現在のレコードのドメインにアクセスできません。	ユーザーは次のことができます。 • 参照フィールドには、表示値が表示されません (これは、Madrid 以降のリリースでドメインセパレーションがアクティブにされており、ユーザーがそのレコードのドメインにアクセスできない場合に該当します)。 • レコードのドメインからのみ値を選択します。たとえば、ACME ドメインからのみユーザーを選択できます。

ドメインと関連会社

ドメインセパレーションを使用すると、会社レコードに加えた変更を、会社に関連付けられたドメインと他のレコードにカスケードできます。

デフォルトでは、ユーザーは会社と同じドメインに自動的に割り当てられます。たとえば、ACME 社のすべてのユーザーは自動的に TOP/ACME ドメインのメンバーになります。

注: admin ロールを持つユーザーは、自分のユーザーレコードを変更できるため、ドメインを変更できます。サービスプロバイダーは、委任管理を無効にするか、ユーザーに admin ロールが必要であることを確認するための承認プロセスを設定することができます。

会社のドメインを変更すると、次の関連レコードのドメインが会社の新しいドメインに合わせて自動的に変更されます。

- 所在地
- 部門
- グループ
- ユーザー

注: インスタンスは、[管理対象ドメイン] チェックボックスをオンにしたレコードのドメインを自動的に変更しません。

ドメインの非アクティブ化および関連会社

ドメインを非アクティブ化すると、インスタンスでは次のアクションも自動的に完了します。

- ドメイン内のすべての会社を非アクティブにします。
- 非アクティブな会社に割り当てられたすべてのユーザーがログインできないようにします。

注: 非アクティブな会社のユーザーがログインしようとすると、「会社が非アクティブ - このインスタンスへのアクセスは許可されていません(Company inactive - your access to this instance is not authorized)」のようなエラーメッセージが表示されます。

たとえば、サンプルデータから ACME ドメインを非アクティブにすると、インスタンスによって ACME 社も非アクティブにされ、3 人のサンプルユーザーがロックアウトされます。

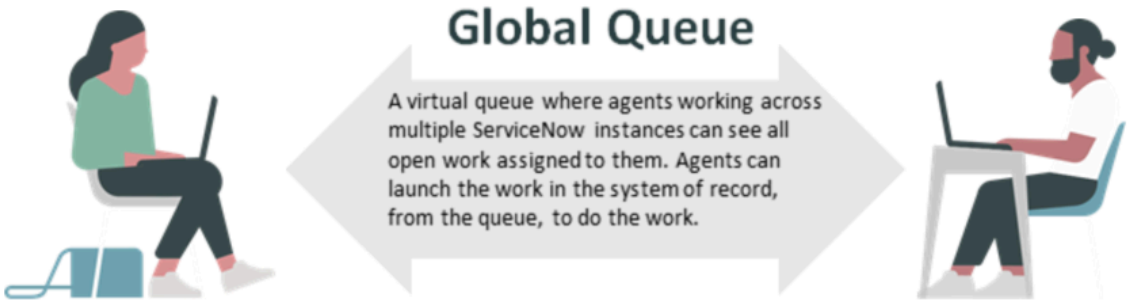
サービスプロバイダーの概念

サービスプロバイダーの概念は、一般的なユースケースの解決に役立つ既存の ServiceNow プラットフォーム機能を扱います。

グローバルキュー **v.2**

グローバルキューの概念は、複数のインスタンスに存在するタスクの単一の仮想ビューを提供します。このコンセプトでは、タスクやデータを複製することなく、複数のインスタンスに存在する作業の履行者ビューを提供するカスタムアプリケーションを作成します。

概要



自動翻訳

複数のシステムのタスクを担当するエージェントを持つサービスプロバイダーは、データを中央のインスタンス (またはインスタンス間の「回転椅子」) に統合する傾向があります。この方法が適切な場合もありますが、ビルドと維持にコストと時間がかかります。また、この方法では、データが存在するすべてのインスタンスで、一般データ保護規則 (GDPR) などの潜在的な監査やデータ要件を考慮する必要があります。

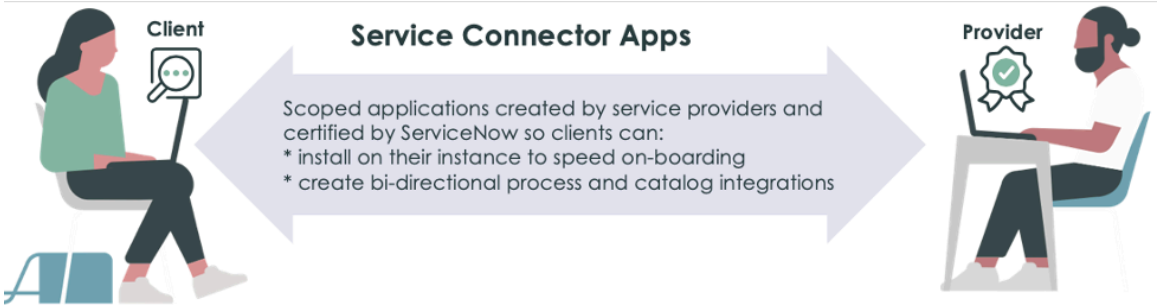
グローバルキュー v.2 は代替手段です。この方法では、エージェントは、ログインしているインスタンスに主権データを保持することなく、単一のインスタンスから自分に割り当てられたデータを表示できます。たとえば、クライアントにデータの所在要件があり、他の国のエージェントによるアクセスが許可されている場合、プロバイダーはグローバルキュー v.2 を使用する「フォローザサン」ヘルプデスクを使用できます。

グローバルキュー v.2 の概念実証 の詳細については、ServiceNow ナレッジサイトを参照してください。

注: Quebec リリース以降では、グローバルキューの概念実証がグローバルキュー v. 2 にアップグレードされています。

サービスプロバイダーコネクタ

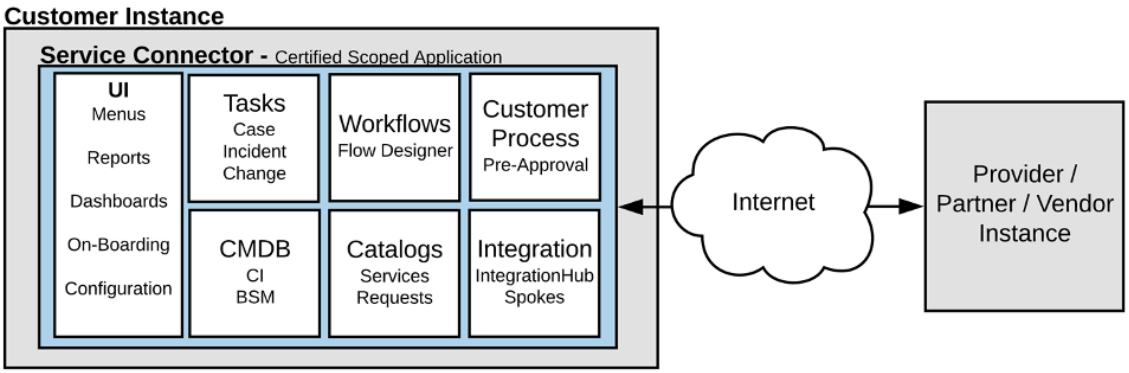
サービスプロバイダーコネクタアプリケーションは、顧客がシステムとの統合に使用する ServiceNow ストアアプリケーションを作成するためのリファレンス設計です。サービスプロバイダーアプリケーションは、オンボーディングを迅速化し、標準化された統合を実現するのに役立ちます。



サービスコネクタのメリット

サービスプロバイダー (ベンダー、サプライヤー、パートナー) がコネクタを公開すると、顧客のオンボーディングが迅速化されるため、請求が速くなります。統合されたインスタンスにより、生産性、ServiceNow エコシステムの可視化、およびパートナープログラムのメリットが向上します。具体的なメリットは次のとおりです。

- カスタム統合 (提供および維持に必要なサービスコストを含む) が不要になります。
- サービスは ServiceNow インスタンス内でプロバイダーによって定義されるため、カスタム統合の複雑さとコストが削減されます。
- ワークフローとカタログ要求は、プロバイダーのプロセスに先行する顧客のプロセスと承認により同期できるため、顧客は独自のプロセスに従うことができます。
- プロバイダーのインスタンスで顧客に対して作成または変更されたデータ (CI など) は、可視化を提供したり、プロセスで使用したりするために、顧客のインスタンスに戻すことができます。



主な機能

機能	説明
ユーザーインターフェイス	サービスコネクタには、少なくとも次の UI コンポーネントが含まれます。 • レポート/ダッシュボード – ペルソナによって可視化が高められた事前定義のレポートとダッシュボード。 • メニューとモジュール – 顧客は、わかりやすいメニュープロバイダー **XYZ** サービスを使用してアプリケーションを見つけることができます。 • ロール – 顧客によってアクセスをコントロールできるようにするため、すべてのコネクタにユーザーロールと admin ロールが必要です。例：「x_snc_xyz_user」または「x_snc_xyz_admin」。 • オンボーディング – 迅速なオンボーディングは、優れたカスタマーエクスペリエンスに不可欠です。また、顧客が必要なプロフェッショナルサービスを必要とせずに本番稼働できるように、迅速なオンボーディングをプレイブック、ガイド付きセットアップ、またはカタログアイテムによって組み込む必要があります (プロフェッショナルサービスを利用できますが、このサービスを利用せずにオンボーディングを行うことができます)。 • 構成 – データポイントをプロセスに組み込み、顧客が達成する必要がある通常のプロセス構成のほとんどを占めるようにする必要があります。そうすることで、プロフェッショナルサービスの必要性が可能な限り排除されます。 • ドキュメント – 詳細なドキュメントを提供することで、顧客満足度と使いやすさが向上します。

機能	説明
	• サポート統合または連絡先情報 – 顧客は、問題、質問、または要求がある場合にサービスプロバイダーと連絡を取り合うための簡単な方法を必要としています。
タスク	インシデントやケース、変更、問題などのタスクをコネクタ内で事前定義する必要がある場合は、 アプリケーションを使用します。フローデザイナー と 統合ハブ を使用してビルドされた統合は、高レベルの復元力とパフォーマンスを確保するのに役立ちます。
CMDB	適切な ITIL プロセスに必要な CI などのコアデータ同期は、プロバイダーと顧客のインスタンス間で常に同期する必要があります。
ワークフロー	対障害弾力性とパフォーマンスを確保するには、すべてのワークフローを フローデザイナー で設計する必要があります。
カタログ	顧客が要求するプロバイダーのカタログは、レコード生成カタログとしてアプリケーションに含める必要があります。顧客のインスタンス内のアイテムで生成された要求は、プロバイダーのインスタンスと eBonding する必要があります。プロバイダーのワークフローは、要求を最新の状態に保ち、顧客のインスタンスに同期します。
カスタマープロセス	要求をプロバイダーのインスタンスと同期するメカニズムでは、要求がプロバイダーのインスタンスに送信される前に、顧客のプロセスがその要求とやり取りできるようにする必要があります。プロバイダーの処理中に、必要に応じて顧客のインスタンスに承認を送信できます。
統合	高レベルの復元力とパフォーマンスを確保するために、統合は フローデザイナー と 統合ハブ を使用してビルドする必要があります。

サービスコネクタに含めることができるコンポーネント

コンポーネント	説明
インスタンスデータレプリケーション (IDR)	レプリケーションが目的の場合： • プロセス統合に使用できるものの、状況移行に基づく複雑性統合ロジックによっては厳格すぎる場合があります。 • 詳細については、「インスタンスデータレプリケーション (IDR)」を参照してください。
統合ハブ	プロセス統合が目的の場合：

コンポーネント	説明
	• フローの途中の複雑なステップまたは条件付きステップの一部として、プロセスの途中で簡単に挿入できます。 • 詳細については、「IntegrationHub」を参照してください。
グローバル作業キュー (仮想)	タスクフェデレーションが目的で、データを外部に保存できない場合： • エージェントが複数の ServiceNow インスタンスにまたがって作業していて、割り当てられたすべてのオープン作業を確認する必要がある場合に使用されます。 • 返される行数は 1000 未満に制限する必要があります。 • 詳細については、「グローバル作業キュー」を参照してください。
リモートテーブル	ストレージなしに外部データを使用することが目的である場合： • インスタンスをサードパーティソースまたは別のインスタンスに接続し、外部データを取得して必要に応じてメモリにキャッシュできるようにするために使用します。データは、グループ、ソート、アグリゲート、フィルターなどの読み取り専用の目的で、インスタンス内のテーブルとして扱われます。 • 詳細については、「リモートテーブルとスクリプトを使用した外部データの取得」を参照してください。
フローデザイナー	プロセス設計が目標である場合 • 単一の設計環境でプロセスを自動化するために使用されます。プロセスオーナーは自然言語を使用して、承認、タスク、通知、およびレコード操作をコーディングなしで自動化できます。 • 詳細については、「フローデザイナー」を参照してください。

サービスコネクタの概念実証 の詳細については、ServiceNow ナレッジサイトを参照してください。

ドメインセパレーションと一緒にインストールされるコンポーネント

いくつかのプラットフォームコンポーネントが、ドメインセパレーションによって追加または変更されます。

ロール

ロール	説明
domain_admin	ドメインの作成、編集、削除ができます。

[sys_domain] フィールドへの追加

次のテーブルに [sys_domain] フィールドが追加されます。

[sys_domain] フィールドを含むテーブル

テーブル
sys_attachment
sys_user_has_role
sys_group_has_role
sys_email
sys_user_group
core_company
cmn_location
cmn_department
sys_gauge
sys_report
kb_feedback
sysapproval_approver
sys_user_grmember

タスクテーブルのフィールド

MSP 拡張は [task_for] フィールドをタスクテーブルに追加します。この参照フィールドはユーザーテーブルを参照します。

グループタイプのオプション

MSP 拡張は、グループテーブルのタイプフィールドにいくつかの新しいデフォルトオプションを追加します。ドメインをサポートするために、必要に応じてこれらのタイプを追加または更新します。

テーブル
セキュリティ
サポート
可視化

ビジネスルール

名前	テーブル	説明
ドメイン - アクティブ化/非アクティブ化	core_company	少なくとも 1 つの会社がアクティブな場合、関連ドメインをアクティブにします。関連するすべての会社が非アクティブな場合は、関連ドメインを非アクティブにします。
ドメイン - カスケード会社	core_company	会社のドメインとユーザー、グループ、部門、および場所との同期を維持します。
ドメイン - カスケードドメイン - メール	sys_email	メールのドメインと添付ファイルとの同期を維持します。
ドメイン - カスケードドメイン - グループ	sys_user_group	グループのドメインと継承されたロール (sys_group_has_role レコード) との同期を維持します。
ドメイン - カスケードドメイン -ナレッジ	kb_knowledge	ナレッジ記事のドメインと関連するフィードバックとの同期を維持します。
ドメイン - カスケードドメイン - タスク	タスク	関連タスク と、wf_context、wf_executing、wf_history、添付ファイル、メール、task_sla とそのワークフロー、sysapproval_approver とそのワークフロー、および sysapproval_group とそのワークフローとの同期を維持します。
ドメイン - カスケードドメイン - ユーザー	sys_user	ユーザーのドメインとそのグループメンバーシップ (sys_user_grmember) およびロール (sys_user_has_role) レコードとの同期を維持します。
ドメイン - カスケードドメイン - バージョン	wf_workflow_version	ドメインと wf_activity および wf_transition の関連するワークフローバージョンとの同期を維持します。
ドメイン - 会社の非アクティブ化	domain	ドメインが非アクティブ化されている場合、関連する会社を非アクティブ化します。
ドメイン - デフォルト - タスク	タスク	ユーザーのドメインのタスクに基づいてタスクドメインを設定します。このドメインがグローバルである場合、代わりにドメインをデフォルトに設定します。
ドメイン - デフォル	sys_user	ドメインがグローバルである場合、ユーザーのドメインをデフォルトに設定します。

名前	テーブル	説明
ト - ユーザー		
ドメイン - グローバルドメインレコードの禁止	domain	global という名前のドメインが作成されないようにします。
ドメイン - 上書きコピー	sys_app_application	ドメインでアプリケーションが上書きされると、新しいアプリケーションにそのモジュールのコピーが作成されます。
ドメイン - 上書きコピー	sys_data_policy2	ドメインでデータポリシーが上書きされると、新しいデータポリシーのデータポリシールールのコピーが作成されます。
ドメイン - 上書きコピー	sys_gauge	ドメインでゲージが上書きされると、新しいゲージのゲージ数のコピーが作成されます。
ドメイン - 上書きコピー	sys_ui_action	ドメインで UI アクションが上書きされると、新しい UI アクションの UI アクションビューのコピーが作成されます。
ドメイン - 上書きコピー	sys_ui_list_control_embedded	ドメインで埋め込みリストコントロールが上書きされると、新しい埋め込みリストコントロールのクライアントおよびサーバースクリプトのコピーが作成されます。
ドメイン - 上書きコピー	sys_ui_policy	ドメインで UI ポリシードメインが上書きされると、新しい UI ポリシーの UI ポリシーアクションのコピーが作成されます。
ドメイン - ドメインを設定 - 承認	sysapproval_approver	承認されるレコードのドメインに基づいてドメインを設定します。
ドメイン - ドメインを設定 - 添付ファイル	sys_attachment	親レコードのドメインに基づいてドメインを設定します。
ドメイン - ドメインの設定 - CMDB_CI	cmdb_ci	CI のドメインを会社のドメインに設定します。
ドメイン - ドメインを設定 - 部門	cmn_department	部門のドメインを会社のドメインに設定します。
ドメイン - ドメインを設定 - ドメイン	domain	ドメインのドメインをそれ自体に設定します。

名前	テーブル	説明
ドメイン - ドメインを設定 - メール	sys_email	親レコードのドメインに基づいてドメインを設定します。メールの親レコードは、インスタンスフィールドで指定されたレコードです。
ドメイン - ドメインを設定 -フィードバック	kb_feedback	ナレッジフィードバックのドメインをナレッジ記事のドメインに設定します。
ドメイン - ドメインを設定 - グループ	sys_user_group	グループのドメインを会社のドメインに設定します。
ドメイン - ドメインを設定 -グループ承認	sysapproval_group	承認されるレコードのドメインに基づいてドメインを設定します。
ドメイン - ドメインを設定 - グループロール	sys_group_has_role	グループロールのドメインをグループのドメインに設定します。
ドメイン - ドメインを設定 - 場所	cmn_location	場所のドメインを会社のドメインに設定します。
ドメイン - ドメインを設定 - タスク SLA	task_sla	タスク SLA のドメインをそのタスクのドメインに設定します。
ドメイン - ドメインを設定 - ユーザー	sys_user	ユーザーのドメインを会社のドメインに設定します。
ドメイン - ドメインを設定 - ユーザーロール	sys_user_has_role	ユーザーロールのドメインをユーザーのドメインに設定します。
ドメイン - ドメインを設定 - WF アクティビティ履歴	wf_history	親ワークフローコンテキストのドメインに基づいてワークフローアクティビティの履歴ドメインを設定します。

名前	テーブル	説明
ドメイン - ドメインを設定 - WF コンテキスト	wf_context	参照されるレコードのドメインに基づいてワークフローコンテキストドメインを設定します (存在する場合)。
ドメイン - ドメインを設定 - 実行アクティビティ WF	wf_executing	親ワークフローコンテキストのドメインに基づいてワークフロー実行アクティビティのドメインを設定します。
ドメイン - タスクの設定 - 変更	change-request	チケットを変更要求に変換するときに、[要求元] フィールドの値をチケットのタスクに設定します。
ドメイン - タスクの設定 - インシデント	incident	チケットをインシデントに変換するときに、[問い合わせユーザー] フィールドの値をチケットのタスクに設定します。
ドメイン：デフォルトを検証	domain	1 つのドメインのみに [デフォルト] チェックボックスをオンにします。
ドメイン - プライマリを検証	domain	1 つのドメインのみに [プライマリ] チェックボックスをオンにします。
ドメインサポートプラグインと一緒にインストールされるビジネスルール		
ドメインセットを変更	sys_dictionary	現在のドメインに設定されたドメインを設定します。
ドメインサポートプロパティ	sys_properties	ドメインクエリー方法 (ドメインパスまたはドメイン番号付け) に一致するようにシステムプロパティを設定します。

クライアントスクリプト

クライアントスクリプト	説明
ドメイン - 会社と場所を設定 (sys_script)	[インシデントの発信者] フィールドの変更を監視します。[会社] と [場所] フィールドにまだ値が含まれていない場合、スクリプトはこの情報を発信者レコードから追加します。[会社] と [場所] フィールドに既に値が含まれている場合、スクリプトは既存の値を保持します。
非アクティブ化されたスクリプト	

クライアントスクリプト	説明
(BP) 場所をユーザーに設定	[インシデントの場所] フィールドを監視し、[場所] フィールドを発信者の場所に設定します。

関連トピック

サービスプロバイダー向けのドメインセパレーションの推奨プラクティス

アプリケーションでのドメインセパレーションのサポート

多くの ServiceNow アプリケーションはベースシステムでドメインセパレーションをサポートしていますが、すべてのアプリケーションではありません。一部のサポートされているアプリケーションには、ドメインを分離できるデータ設定と管理設定に関する制限があります。これらの定義は、実際のユースケースとそれを使用するユーザーの観点から、ドメインセパレーションのサポートレベルを示しています。

ドメインセパレーションのサポートレベル

ドメインセパレーションをサポートする ServiceNow アプリケーションは、データとデータルーティングの分離のみをサポートするか、高度なビジネスロジック分離機能があるか、アプリケーションのテナント (顧客) レベルの管理をサポートする可能性があります。ServiceNow アプリケーションは、次の増分サポートレベルで定義されます。

サポートはありません

- ドメインフィールドがデータテーブルに存在している可能性がありますが、データを管理するロジックはありません。
- このレベルでは、ドメイン分割は考慮されません。

基本

- アプリケーションサービスプロバイダー (SP) のユースケースに合わせてデータが適切なドメインに送られるようにするビジネスロジックが存在します。
- アプリケーションでは、ユーザーインターフェイス、キャッシュキー、レポート、ロールアップ、集計など、すべて本番環境の実行時にドメインを使用します。
- インスタンスのオーナーは、複数のテナント間で機能するようにアプリケーションをセットアップできる必要があります。

サンプルユースケース：サービスプロバイダーがチャットを使用してテナント顧客のメッセージに応答する場合、クライアントがサービスプロバイダーの応答を確認できるようにする必要があります。

標準

- ベーシックレベルサポートを含みます。
- ビジネスロジック：サービスプロバイダー (SP) によって顧客ごとにプロセスを作成または変更できます。ユースケースには、単一のインスタンスでの複数のサービスプロバイダー顧客によるアプリケーションの正しい使用が反映されています。
- インスタンスのオーナーは、特定のアプリケーションで想定されているテナントあたりの最小実行可能製品 (MVP) ビジネスロジックとデータパラメーターを設定できる必要があります。

サンプルユースケース：admin は、レコードを他のテナントに対してはクローズしないが、1 つのテナントに対してクローズする場合、コメントを必須にできる必要があります。

拡張機能

- ベーシックレベルと標準レベルを含みます。
- データドリブンプロセスにより、サービスプロバイダーの顧客は定義されたユースケースに基づくビジネスロジックを変更できます。これらの構成は UI ベースでフェイルセーフであるため、1 人の顧客による構成が別のユーザーに影響を与えることはありません。
- インスタンスのテナントは、それ自体、MVP (minimum viable product) ビジネスロジックとデータパラメーターを設定できる必要があります。アプリケーションの通常の関数では、このロジックとパラメーターが想定されます。

サンプルユースケース：共有環境のテナント顧客は、影響度、緊急性、または優先度のマトリクスに変更を加えて、ドメイン内で優先順位を設定できる必要があります。

注: 有効なドメイン **(*)**

ドメインフレームワークが使用されていなくても、プラットフォーム機能またはアプリケーションが SP のユースケースを効果的にサポートできる場合があります。その場合、ユースケースにはドメインセパレーションのサポートが詳述されている必要があります。サポートレベルの後のアスタリスク (*) は、この種類の構成を示します。

サポートされている機能	基本	標準	拡張機能
ベースシステムのアプリケーションテーブルにドメイン列が存在します。	●	●	●
ドメイン固有の構成は、インスタンスオーナーによって管理されます。	●	●	●
テナントドメインは、独自のアプリケーションデータを管理できます。		●	●
アプリケーションプロパティは、必要に応じてドメイン対応です。		●	●
インスタンスオーナーは、ビジネスロジックとプロセスをドメインセパレーションすることができます。		●	●
ビジネスロジックとプロセスはテナントドメインによって管理できます。			●

アプリケーションによるサポートレベル

製品スイート	アプリケーション	サポートレベル
アプリ開発とローコード	App Engine Studio	サポートはありません
	オートメーションセンター	基本
	ロボティックプロセスオートメーション (RPA) ハブ	基本
	ServiceNow スタジオ	サポートはありません
	テーブルビルダー	基本
	App Engine 管理センター	サポートはありません
	ディシジョンテーブルの探索	標準
	エンタープライズリソースプランニング統合	サポートはありません
	Enterprise Resource Planning Customization Mining	サポートはありません
	Next Experience UI ビルダー	基本
カスタマーサービス管理 (CSM)	Communities	サポートはありません
	カスタマーサービス管理 (CSM)	基本
	リリース管理	基本*
	カスタマーサービス管理 (CSM) 用注文管理	基本
	Post-Sales Support	基本
	カスタマーサービスのワークフォース最適化のドメインセパレーション	基本
DevOps	ドメインセパレーションと DevOps 変更速度	サポートはありません
	ドメインセパレーションと DevOps コンフィグ	
従業員サービス管理 (ESM)	HR サービスデリバリ (HRSD)	基本*
	衛生安全	サポートはありません
	契約管理プロ	基本
	法務サービスデリバリ	基本
	調達サービス管理 (PSM)	サポートはありません
	セーフワークプレイススイート	個々のアプリケーションサポートレベルについては、アプリケーションサイトを参照してください
	SharePoint Online 検索コネクター	基本
	ユニバーサル要求	基本

製品スイート	アプリケーション	サポートレベル
	ユニバーサルタスク	基本
	HR のワークフォース最適化	基本
環境、社会、ガバナンス管理	環境、社会、ガバナンス管理	基本
フィールドサービス管理 (FSM)	フィールドサービス管理 (FSM)	基本
ガバナンス、リスク、コンプライアンス	事業継続性管理	基本
	ガバナンス、リスク、コンプライアンス (GRC)	基本
	Operational Resilience	基本
業種別製品		
・ファイナンシャルサービス	Financial Services Card Operations	基本
	Financial Services Deposit Operations	
	Financial Services Payment Operations	
	Intelligent Servicing for Fraud	
	Property and Casualty Insurance Servicing	
	Life Insurance Servicing	
	Insurance Claims	
	Financial Services Know Your Customer	
	Financial Services Credit Operation	
	Financial Services Document Processor	
・医療および生命科学	EMR Help	基本
	Healthcare and Life Sciences Service Management Core	基本
	Pre-Visit Management	基本
	Patient Support Services	基本
	ワクチン接種アドミニストレーション管理	基本
製造業営業オペレーション	製造業営業オペレーション	標準
	小売業コア	基本
	小売業タスク管理	基本
・製造業	産業用プロセスマネージャー	標準

自動翻訳

製品スイート	アプリケーション	サポートレベル
	オペレーショナルテクノロジーマネージャー	標準
	オペレーショナルテクノロジー脆弱性対応	標準
	オペレーショナルテクノロジーマネージャー	標準
電気通信、メディア、およびテクノロジー (TMT)	カスタマーサクセス管理	基本
	カスタマーサービス問題管理	基本
	通信事業、メディア、テクノロジー (TMT) 向け Now Assist	基本 (Now Assist アドミンコンソールでのドメインセパレーション から継承)。
	Proactive Service Experience Workflows	標準
	Service Bridge (サービスブリッジ)	標準
	Technology Product Support Case アプリケーション	基本 (カスタマーサービス管理 (CSM) から継承)。
	通信ネットワークインベントリ	基本
IT Asset Management	クラウドインサイト	サポートはありません
	ハードウェア資産管理	拡張機能
	ソフトウェア資産管理	拡張機能
	Enterprise Asset Management (エンタープライズ資産管理)	標準
戦略的ポートフォリオ管理 (SPM)	Agile Development	基本*
	アラインメントプランナーワークスペース	基本
	アプリケーションポートフォリオ管理	基本
	コスト管理	サポートはありません
	デマンド管理	基本
	ファイナンシャルマネジメント	サポートはありません
	投資枠管理	基本
	プロジェクトポートフォリオ管理	基本*
	リリース管理	基本*
	Scaled Agile Framework (SAFe)	基本*
	テスト管理	基本*
	ゴールフレームワーク	基本

自動翻訳

製品スイート	アプリケーション	サポートレベル
IT Operations Management (ITOM)	Cloud Provisioning and Governance	基本
	エージェントクライアントコレクター	基本
	ディスカバリー	標準
	イベント管理	基本
	ITOM のサービスオペレーションワークスペース	基本
	ヘルスログアナリティクス	基本
	メトリックインテリジェンス	基本
	サービスマッピング	基本
	クラウド移行アセスメント	基本
	アクションライブラリ	サポートはありません
	クラウドコンフィグレーションガバナンス	サポートはありません
	Tag Governance	基本
	クラウドインサイト請求	サポートはありません
	Cloud Provisioning and Governance：Google Cloud	基本
	Cloud Provisioning and Governance Terraform	基本
	クラウドオペレーションワークスペース	基本
	クラウドディスカバリー	標準
IT Service Management	Benchmarks	サポートはありません
	変更管理	基本
	コーチング	基本
	継続的改善管理	基本
	契約管理	サポートはありません
	ドメインセパレーションとデジタルプロダクトリリース	基本
	経費ライン	サポートはありません
	インシデントコミュニケーション管理	標準
	インシデント管理	標準
	設備サービス管理	標準
	インシデント管理	標準
	オンコールスケジューリング	標準

自動翻訳

製品スイート	アプリケーション	サポートレベル
	資産管理	基本
	問題管理	標準
	調達	標準*
	製品カタログ	標準
	要求管理	標準
	サービスカタログ	標準
	サービスレベル管理	基本
	サービスポートフォリオ管理	基本*
	サイトリライアビリティオペレーション	基本*
	タスクの機能停止	基本
	ドメインセパレーションと ベンダー管理ワークスペース	サポートはありません
	ウォークアップエクスペリエンス	基本
モバイル構成およびナビゲーション	モバイル	基本
Now Intelligence	Dashboards	基本
	パフォーマンスアナリティクス	拡張機能
	プロセス最適化	基本
	Reporting	基本
	ユーザーエクスペリエンスアナリティクス	基本
[変更管理 - リスク アセスメント] プラグインが有効化されているときは、「 ServiceNow AI Platform	管理	
	ドメインセパレーションと エージェントチャット	標準
	ServiceNow AI Platform 機能	
	ユーザーインターフェイス	
	Advanced Work Assignment (高度な作業アサイン)	標準
	AI 検索	インデックス付きレコードからのドメイン制限を考慮した検索
	App Engine Studio	サポートはありません
	アプリケーション管理	サポートはありません
	アセスメント	標準
	自動テストフレームワーク (ATF)	標準*
	ServiceNow 音声アシスト機能	

自動翻訳

製品スイート	アプリケーション	サポートレベル
	『Code Signing』	基本サポート
	コンテキスト検索	標準
	Configuration Management (CMDB)	標準
	コンテンツ管理システム	サポートはありません
	認証情報と接続	標準
	データ認定	基本*
	データ分類	拡張機能
	データプライバシー	サポートはありません
	データ管理	基本*
	委託開発	サポートはありません
	依存関係ビュー	基本
	ドキュメントサービス	サポートはありません
	動的翻訳	基本
	エッジ暗号化	基本サポート
	外部コンテンツコネクタ	サポートはありません*
	フィールド暗号化	サポートはありません
	暗号化	サポートはありません
	鍵管理によるクラウド暗号化	基本サポート
	フィールド正規化	サポートはありません
	フローデザイナー	標準*
	ガイド付きセットアップ	サポートはありません
	ドメインセパレーションと 統合ハブ	標準*
	サードパーティアプリケーションおよびデータソースとの統合	基本 + 標準
	ナレッジ管理	標準
	Hermes Messaging Service	サポートはありません
	ドキュメント管理	サポートはありません
	メトリックベース	基本
	Natural Laungage Understanding	基本 + 標準
	通知	標準
	ODBC Driver	基本*
	オーケストレーション	標準*
	パスワードリセット	標準

自動翻訳

製品スイート	アプリケーション	サポートレベル
	プラットフォームセキュリティ	ドメインセパレーションランディングページ
	データプライバシー	サポートはありません
	予測インテリジェンス	標準
	プロアクティブトリガー	基本
	プロセスオートメーションデザイナー	基本
	リモートテーブル	サポートはありません
	スケジュール	基本
	スクリプトデバッガー	基本
	検索提案	サポートはありません
	サービスポータル	サポートはありません
	サービスグラフコネクタ	サポートはありません
	ドメインセパレーションと サイドバー	標準
	状況フロー	サポートはありません
	サブスクリプション管理	基本*
	サーベイ管理	基本*
	タスクインテリジェンス	サポートはありません
	ドメインセパレーションとタイムカード	基本*
	UI ビルダー	標準
	仮想エージェント	基本
	ビジュアルタスクボード	基本
	Web サービス	標準*
	Workflow	標準*
	ワークスペース	標準
セキュリティオペレーション	コンフィグレーションコンプライアンス	標準
	構成データ管理	基本
	IBM QRadar 違反の取り込み	基本
	Microsoft Graph Security API アラート取り込み統合	基本
	セキュリティインシデントレスポンス	標準
	脅威インテリジェンス	標準
	脆弱性対応	標準
サービス管理	設備サービス管理	標準

製品スイート	アプリケーション	サポートレベル
	計画済みメンテナンス	標準*
	プロアクティブトリガー	基本
	Now コードエディター	サポートはありません
	ITSM のワークフォース最適化	基本
	Vendor Management Workspace	基本

サービスプロバイダー向けのドメインセパレーションの推奨プラクティス

アプリケーションとサービスのドメインセパレーションを作成、実装、および維持できます。

ドメインの基本

ドメインセパレーション (ServiceNow マルチテナントプラットフォームアーキテクチャとも呼ばれる) を使用すると、クロステナントインテリジェンスを使用した階層モデルをサポートする、アプリケーションデータ、ユーザーインターフェイス、およびビジネスロジックを単一の顧客インスタンスに分離できます。ビジネスロジックは、ドメインセパレーションの構成方法と、構成に影響するルールを記述します。

ドメインセパレーションのジャーニーを開始する前に、従うべき推奨事項がいくつかあります。必要に応じてトピックを選択するか、画像の下のリンクをクリックして順番にフォローしてください。

ドメインセパレーションの説明

ドメインセパレーションを使用すると、アプリケーションデータ、UI、および ビジネスロジック (ルールやワークフローなど) を単一の顧客インスタンスに分離できます。これらの要素を論理的に定義されたドメインに分離することで、アプリケーションを使用するすべての顧客の特定の階層をサポートします。

ドメインの基本

ドメインセパレーションは、ServiceNow マルチテナントプラットフォームアーキテクチャとも呼ばれ、インスタンスの管理にかなりのオーバーヘッドが追加されます。ただし、ドメインセパレーションを正しく使用すると、効率が向上し、セキュリティが強化され、顧客のインスタンスのパフォーマンスが向上します。

一部のグローバル標準とプロパティ (システムプロパティやテーブルスキーマなど) をテナントごとに分離することはできません。

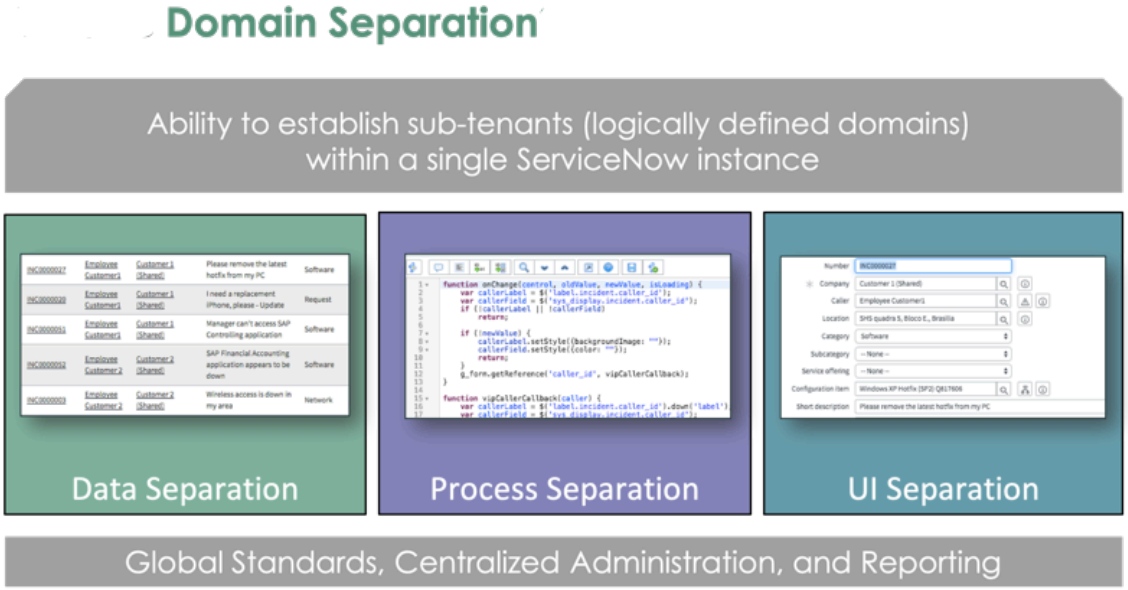
ドメインの分離を開始する前に、次のガイドラインをお読みください。

ドメインセパレーションでできること

- データ分離：ドメインのテナントには、表示権限のあるデータのみが表示されます。テナントには、他のテナントデータへのアクセス権を付与できますが、アクセス権がないテナントデータを照会することはできません。
  - データレコードを更新しても、更新セットレコードは生成されません。
  - 統合に使用される顧客アカウントも含め、ユーザーには、アクセス権限があるドメイン内のデータのみが表示されます。
  - 顧客、エージェント、および履行者には、サポートする顧客と組織に関するデータが表示されます。
- UI 分離：ビュー、リスト、ラベルといった UI 要素のテナント固有のエクスペリエンスをサポートします。
  - アプリケーションメニュー、リスト、フォーム、ダッシュボードなどのブラウザーベースのユーザーインターフェイスを上書きできます。基本的なプロセスロジックを保持したまま、特定のドメインまたは一連のドメイン用にカスタマイズすることもできます。
  - サービスプロバイダーは、表示されるブランディングと UI 要素を変更して、個々の顧客のニーズを満たすことができます。
- ビジネスロジック分離：メール通知、ビジネスルール、クライアントスクリプト、UI ポリシー、UI アクションなど、テナント固有のシステムポリシーを作成できます。
- 階層モデル：マルチテナントがネストされるため、親テナントが子テナントのリソースにアクセスできます。親テナントのビジネスロジックは子テナントに対して自動的に実行され、任意のレベルで上書きできます。
- クロステナントインテリジェンス：追加のテナントデータにアクセスできるテナントのデータ、メタデータ、ビジネスロジック、および処理コンテキストを自動的に処理します。

ドメインセパレーションの概要

次の図は、データの分割、プロセス、および UI の分離を示しています。これらの概念については、「推奨プラクティス」セクションで詳しく説明しています。

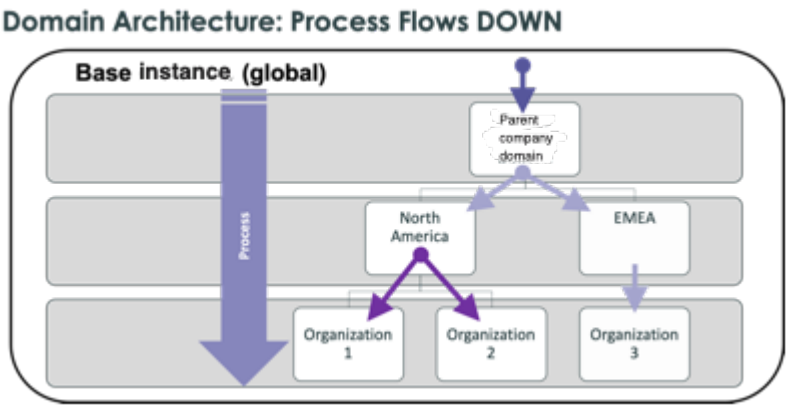


ドメインアーキテクチャ

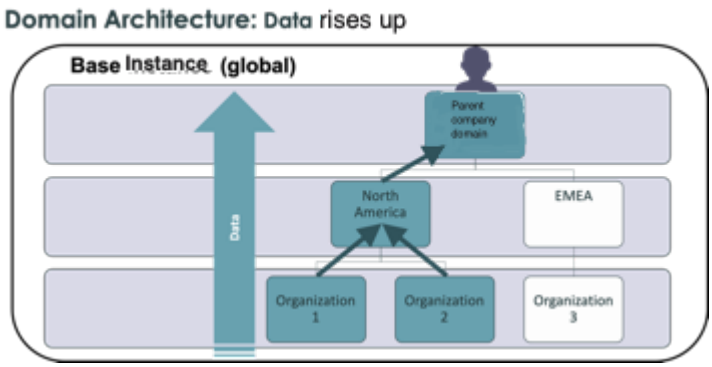
ユーザーレコードには、ユーザーのホームドメインを表すドメイン値がアサインされます。ユーザーは、親ドメイン、ピアドメイン、または階層の他のブランチ内のドメインのデータにアクセスできません。

追加のドメインの可視化を付与する詳細オプションについては、「包含クエリーとドメインアクセス」を参照してください。

次の図は、アーキテクチャプロセスが子ドメインにどのように移行するかを示して



います。



自動翻訳

ドメインセパレーションの価値提案

ドメインセパレーションは、マルチテナントインスタンスアーキテクチャをサービスプロバイダーに提供し、そのクライアントに対して効率的かつ安全に製品を提供できるように設計されています。強力なユニバーサルプロセス標準、データ主導型のプロセス設計、厳格なガバナンス、および集中管理により、これらのメリットを最大限に活用できます。

ドメインセパレーションのメリット

ドメインのテナントは、迅速な投資利益率、低い管理経費、およびインスタンスオーナーが提供するビジネスサービスの活用によるメリットを得ることができます。

メリットのクイックビューを以下に示します。

インスタンスオーナー	ドメインテナント
サービスプロバイダーの従業員の生産性	セキュリティを強化できます
プロセスデルタのみが維持されます	事前にビルドされたプロセスと機能
管理効率	必要なスタッフの削減
クライアント統合数の削減	迅速なオンボーディング
アップグレード可能性と拡張性	最新リリースの活用
データ分離	インスタンスオーナーが提供するサービス
グローバルレポート	

ドメインセパレーションの定義

ドメインセパレーション (ServiceNow® マルチテナントプラットフォームアーキテクチャとも呼ばれる) を使用すると、クロステナント (顧客) インテリジェンスを使用して、アプリケーションデータ、UI、およびビジネスロジックを単一の顧客インスタンスに分離できます。

ドメインセパレーションのプロパティ

ドメインセパレーションされた ServiceNow アプリケーションは、次のプロパティで定義されます。

データ分離

顧客が表示権限を持つデータのみを表示できるようにします。顧客に他の顧客データへのアクセス権を付与できますが、アクセス権がない場合、顧客データを照会することはできません。

UI 分離

ビュー、リスト、ラベルといった UI 要素の顧客固有のエクスペリエンスをサポートします。

ビジネスロジック分離

メール通知、ビジネスルール、クライアントスクリプト、UI ポリシー、UI アクションなど、顧客固有のシステムポリシーをサポートします。

階層モデル

親テナント (顧客) が子顧客リソースにアクセスできるように、ネストされたマルチテナントをサポートします。親顧客のビジネスロジックは子顧客に対して自動的に実行され、任意のレベルで上書きできます。

クロスカスタマーインテリジェンス (ドメインスコープ)

追加のテナントデータにアクセスできるテナントのデータ、メタデータ、ビジネスロジック、および処理コンテキストを自動的に処理します。

ドメインセパレーション階層

ドメインアーキテクチャを定義するときに、プロセスとワークフローを追跡するための階層を作成します。

ドメインセパレーション階層の例

次の図は、ドメインアーキテクチャを定義するための適切な開始点を示しています。これは、TOP および下位のドメイン間の関係性と、プロセス、データ、およびビジネスルールが親および子ドメインにどのように影響するかを示しています。

- 次の例では、TOP がプロセスドメインです。これにユーザーを含めることはできません。むしろ、TOP には、インスタンスオーナーが開発する新しいプロセスと、これらのプロセスへのグローバルドメインからの上書きを含める必要があります。
- サービスプロバイダー (SP) のみがデフォルトドメインにアクセスできます。このドメインには、アクティブなユーザーは含まれません。ここには、正しいドメインに再アサインする必要がある「消失」データのみがあります。

  注: データが特定のドメインにアサインされていない場合、デフォルトドメインに移されます。これは一時的に「消失」した状態であり、正しいドメインにアサインする必要があります。

- ドメインの作成時または更新時に、ドメインのないタスクとユーザーは自動的にデフォルトドメインに配置されます。このアクションを上書きするには、このレコードの [デフォルト] オプションをクリアするか、別のドメインレコードの [デフォルト] オプションを選択します。デフォルトドメインをまだ設定していない場合、ドメインのないタスクとユーザーはグローバルドメインに移されます。
  - インスタンスを使用している間は、ドメイン間でデータを移動しないでください。
  - デフォルトドメインで終了するデータがある場合は、対応するべき構成または手続きの問題があることを意味します。

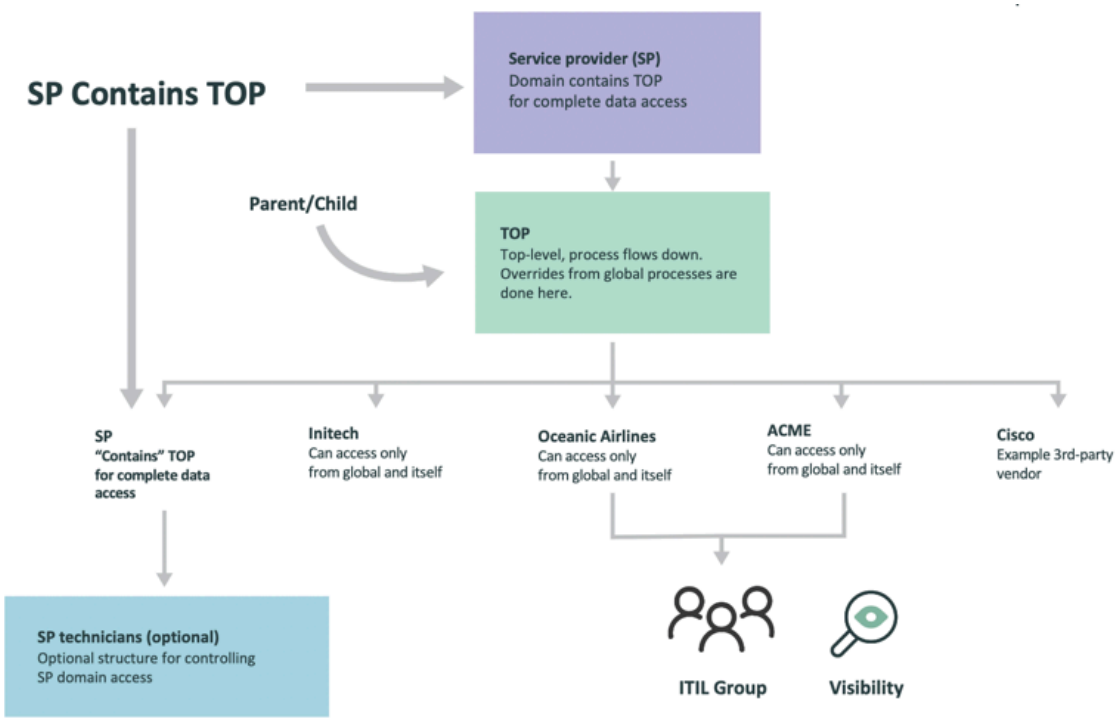
グローバルドメインは存在しないため、この図には「グローバル」という語は表示されません。「グローバル」とは、レコードにドメインが存在しないことを意味します。

たとえば、[ドメイン] フィールドを持たないテーブルは、テーブルにすべてのグローバルレコードが含まれていることを意味します。[ドメイン] フィールドを含むテーブルは、ドメインのないレコードがグローバルドメインであることを意味します。

[ドメイン] フィールドには「グローバル」という語があります。レコードにドメインがない場合は、自動的にそこに配置されます。

インスタンスのすべてのユーザーは、セキュリティ構成によって制限されていない限りグローバルレコードを利用できます。

- グローバルレコードを含めるべきでないテーブルにグローバルドメインに留まるレコードが存在しないように、デフォルトドメインを使用してください。
- インスタンス所有者は、デフォルトドメイン内のレコードをトリアージし、正しいドメインに移動する必要があります。



ドメイン階層

- 親/子： 影響を受けるプロセスとデータ
  - プロセスフローに基づく設計。
  - 親ドメインが子ドメイン内のすべてのデータにアクセスできることに注意してください。
- ドメインを「含む」： 影響を受けるのはデータのみです。たとえば、図の SP に TOP が含まれるようにしても、TOP ドメイン以下で SP のプロセスは実行されません。
  - 特定のドメインへの専用アクセスが必要なグループ内の個々のユーザーに、データアクセス権を付与します。
  - データベースクエリーに追加される原因または条件が含まれ、大規模なドメインやデータセットでパフォーマンスの問題が発生する可能性があります。
- 可視化：アクセス権を付与したユーザーに常に表示される階層。影響を受けるのはデータのみで、プロセスは影響を受けません。

- 親/子階層のビルド時にアクセス権が付与されなかった別のドメインに対して、ドメインのデータアクセス権を付与します。
- ユーザーがどのレコードを使用しているかにかかわらず、可視化アクセス権を付与されているドメイン内のすべてのデータの表示をユーザーに許可します。

> **注:** 可視化は、意図しないような完全なアクセスが許可されてしまうため、慎重に使用してください。

ドメイン階層を定義する基本原則

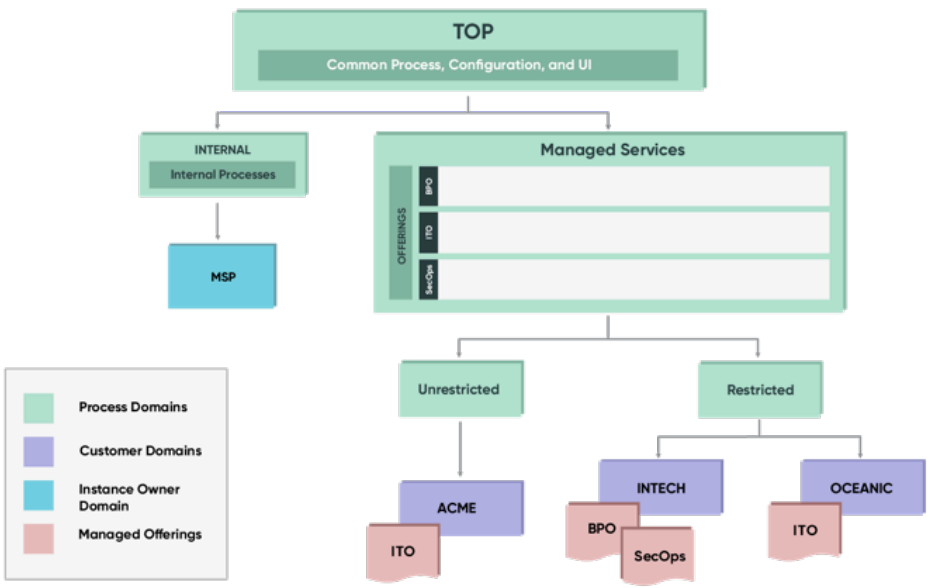
ドメインセパレーションの無制限および制限付きユースケース

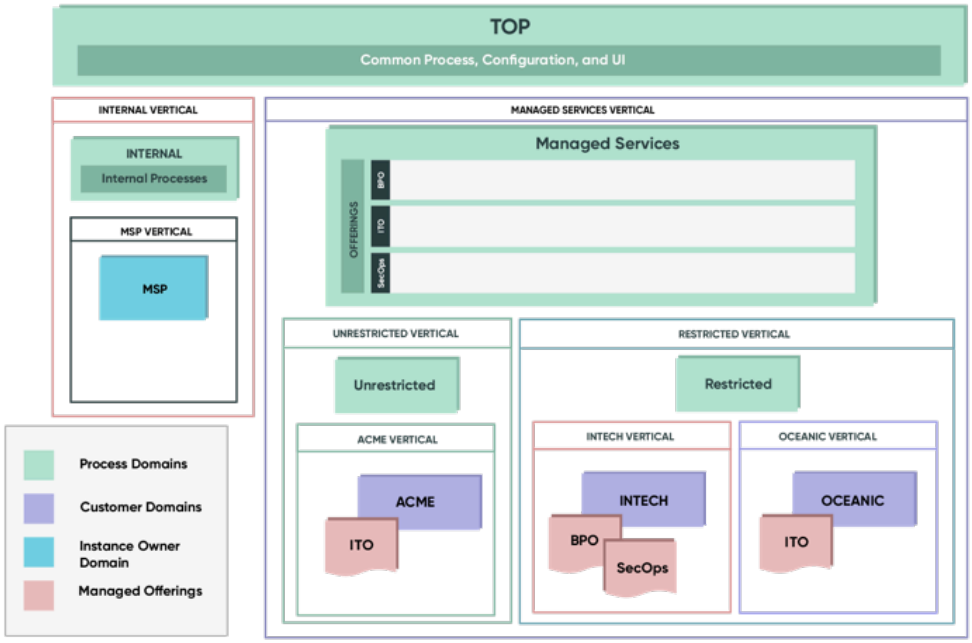
多くの SP には、ドメインへのアクセスを厳密に制限する必要があることを暗黙的に示す顧客が含まれています。この場合、TOP ドメインでの「包含」機能の使用が制限されます。次の図では、ドメインを制限付きドメインと無制限ドメインに分割することで、その規制を軽減する方法について説明します。

1. 顧客は、ドメインセパレーション階層の特定の「バーティカル」に存在します。つまり、そのドメインと、階層内でそのドメインの上位にあるすべての親ドメインで定義されたプロセスのみを消費します。直線的な親子階層にないドメインで定義されたプロセスは適用されません。

   > **注:** 顧客または「テナント」は、互いに完全に分離されたエンティティであり、互いにリソースを共有する部門や事業部門とは異なります。

2. スーパーバーティカル (制限付き、マネージャーサービスなど) は、顧客がどちらか一方にしか属さないのであれば、許可されます。
3. すべての顧客が水平に利用できる必要があるサービス、製品、またはオファリングは、個別のドメイン階層内で定義されていません。



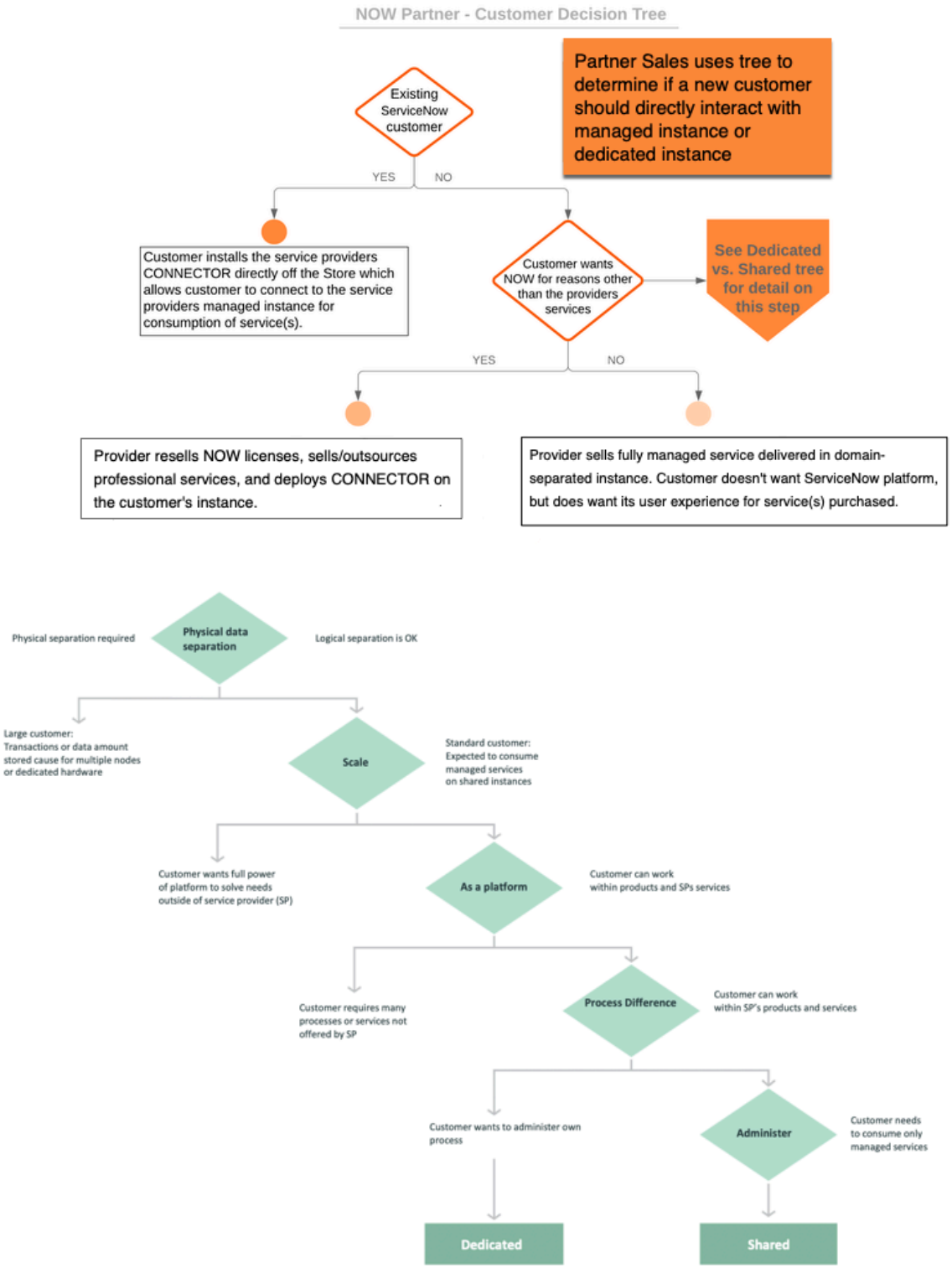


サンプルのユースケースは以下のとおりです。

- TOP の下で、[制限なし] と [制限付き] の 2 つのドメインを作成できます。
  - SP の可視化制限がない顧客とそのドメインを [制限なし] ドメインに配置します。
  - この要件がある顧客とそのドメインを [制限付き] ドメインに配置します。
- これにより、システムアドミニストレーターは、効率的かつターゲットを指定する方法で「包含」および「可視化」機能を使用できます。
  - 「包含」を [制限なし] に適用すれば、1 つの「包含」により、ほとんどの顧客に可視化を付与することができます。
  - ドメインの可視化は、必要に応じて特定のドメインに対して「ドメイン可視化グループ」を使用して適用します。

顧客の決定木

次の図は、適切な階層モデルを選択する方法を説明しています。ドメイン構造に必要なプロセスと機能に応じて、個別の階層、ハイブリッド型、または共有階層を選択できます。



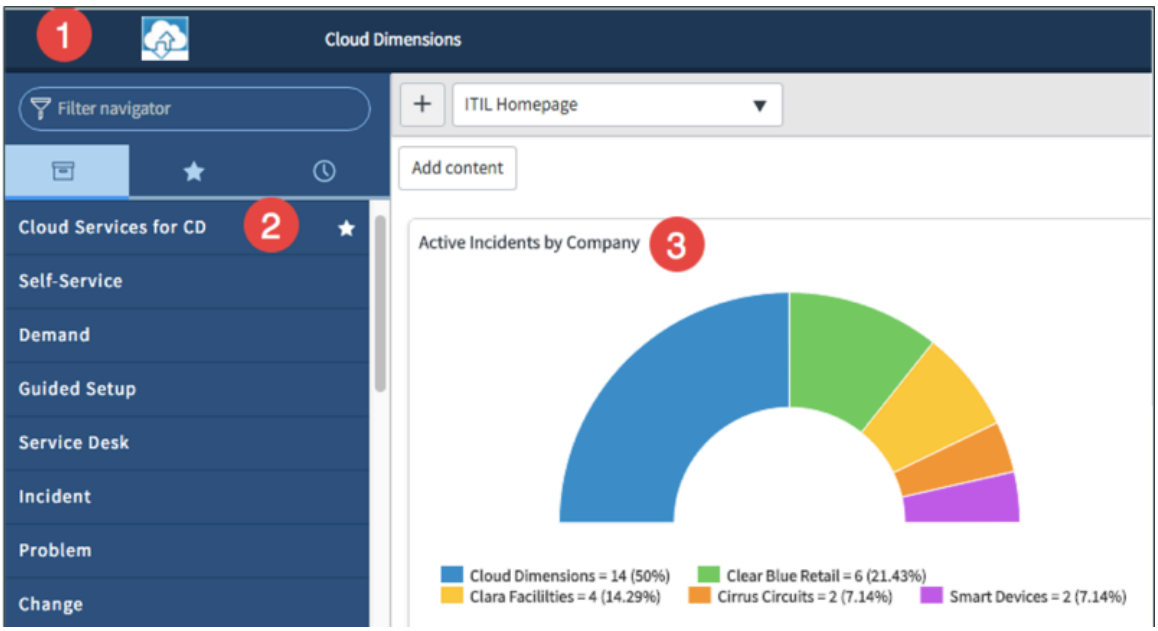
階層アーキテクチャの詳細については、「サービスプロバイダー参照アーキテクチャ」を参照してください。

コンテキストとドメインセパレーション

ユーザーのセッションのコンテキストによって、ユーザーがリストビュー、ホームページ、レポート、およびナレッジ記事を参照する際のプロセス、データ、およびユーザーインターフェイス (UI) が決まります。コンテキストは、作成するプロセス、設定するビジネスルール、ワークフロー、およびその他の要因によって決まります。

ユーザーセッションコンテキスト

ユーザープロファイル、グループ、会社基準など、多くの要因がユーザーセッションのコンテキストを決定します。次の図では、会社が作成したインシデントがコンテキストの一部であることがわかります。

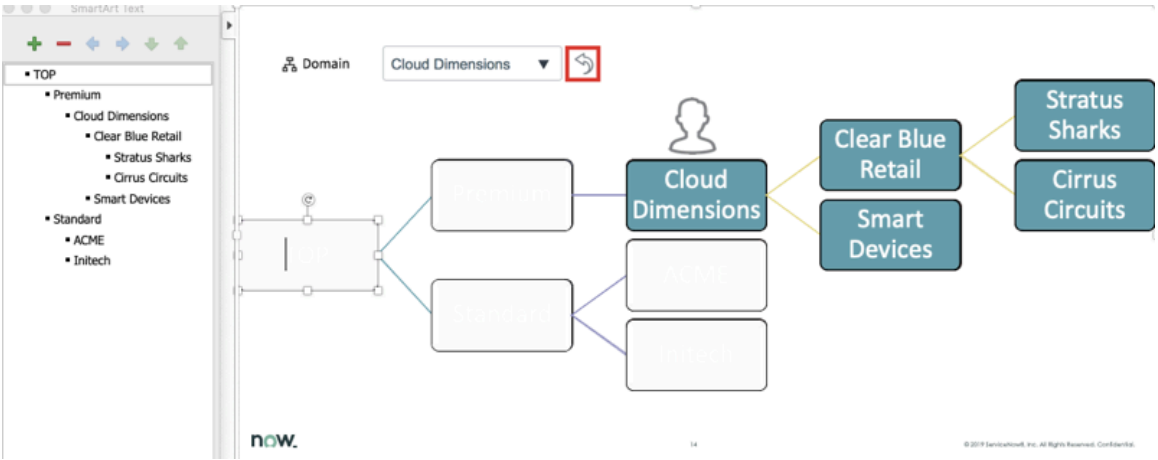


この例のユーザーのホームドメインは Cloud Dimensions です。

1. ブランディングには、Cloud Dimensions ドメインと会社レコードの設定が反映されます。
2. アプリケーションナビゲーターには、上位レベルのドメインから継承されたアイテムと、Cloud Dimensions ドメインで定義されているモジュールが表示されます。
3. ホームページとリストデータには、ユーザーに表示されるデータが反映されます。このデータは、ユーザーのセッションコンテキストに基づいています。この場合、Cloud Dimensions ドメインのユーザーは、Cloud Dimensions、子ドメイン、およびグローバルドメインのデータを表示できます。

ホームドメインでのユーザーセッションコンテキストの開始

次の図は、コンテキストの要素を示しています。



システムアドミニストレーターは、ユーザーのホームドメインをユーザーレコードに設定します。通常、ユーザーのホームドメインは会社のドメインと同じドメインに設定されます。ユーザーがログインすると、ドメインピッカーがユーザーのホームドメインに自動的に設定されます。ユーザーは、ドメインピッカーの矢印アイコンをクリックして、いつでもホームドメインに戻ることができます。

ドメインピッカーのリストには、ユーザーのセッションコンテキスト内のドメインが含まれています。ユーザーは、ピッカーで子ドメインを選択して、セッションコンテキストをさらに制限できます。

ユーザーセッションのコンテキストには、ユーザーのホームドメインと子ドメインが含まれます。ユーザーのセッションコンテキスト内のこの一連のドメインは、データベースに送信されるすべてのクエリに自動的に追加されます。これにより、結果はこれらのドメイン内のデータとグローバルデータのみに制限されます。このプロセスは、アクセスできないコンパイル済みコードに埋め込まれています。

統合に使用されるサービスアカウントにもユーザーセッションコンテキストがあります。ユーザーコンテキストとレコードコンテキストがあり、それぞれ独自のドメインに独自のデータがあります。これらのコンテキストは統合に影響します。データベースクエリー (レコード) は、インタラクティブユーザー (ユーザー) と同じ方法で制限されます。つまり、通常どおりに機能しますが、開発者が設定した制約によって制限されます。

ユーザーのセッションコンテキストにドメインを追加するその他の方法については、「サービスプロバイダー参照アーキテクチャ」を参照してください。

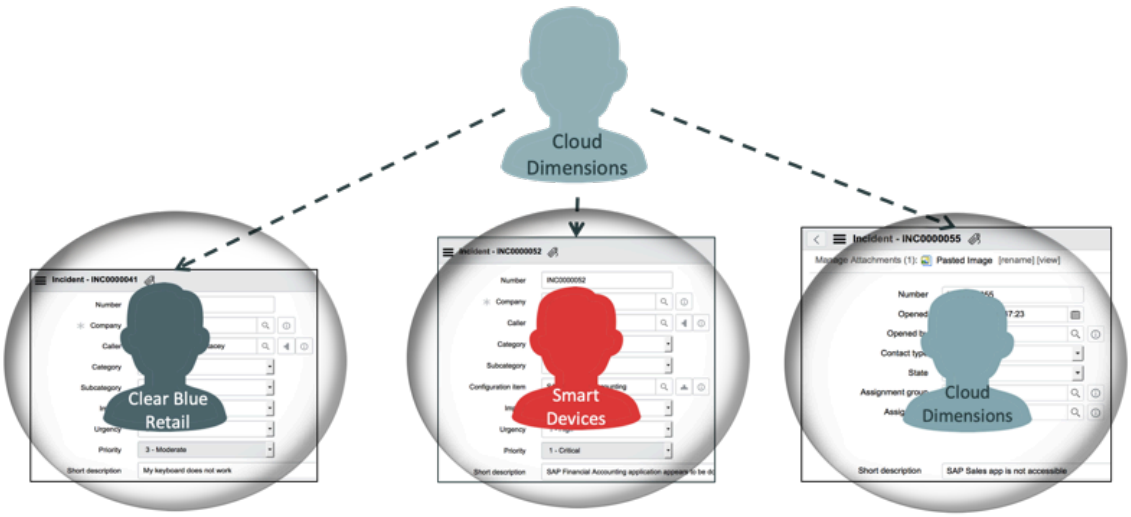
レコードコンテキスト

ユーザーが個々のレコードにドリルダウンすると、レコードコンテキストが有効になります。レコードコンテキストによって、レコードに適用する UI 要素とプロセスが決まります。

レコードのドメインによって、レコード内のプロセス、データ、および UI 要素の可用性が決まります。

注:

- レコードコンテキストは、ユーザーのドメインが変更されても保持されます。
- ユーザーは、独自のレコードコンテキストを維持しながら、複数のブラウザータブで同時にレコードを表示できます。

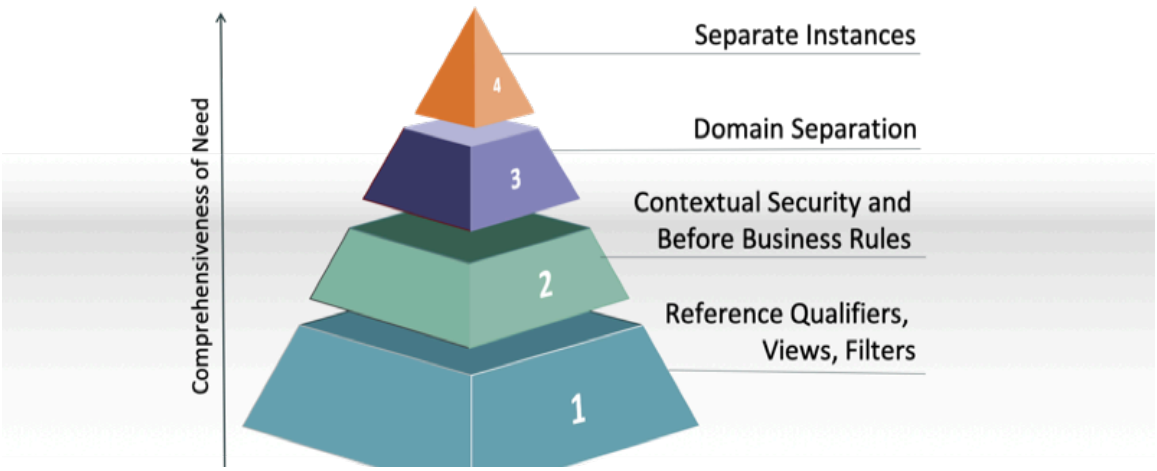


ドメインセパレーションによるデータの分離と保護

顧客のニーズに応じて、ServiceNow プラットフォーム上のデータを複数の方法で分離して保護することができます。

複数の方法でのデータ分離

次の図は、データを分離する 4 つの方法を示しています。データを分離する方法として、個別のインスタンス、ドメインセパレーション、コンテキストセキュリティとビジネスルール、および参照アーキテクチャ自体を使用できます。



次の 4 つの方法でデータを分離できます。

1. 社内の部門やグループが自分の作業に集中できるように、修飾子とフィルターを使用して 参照アーキテクチャ をカスタマイズします。これらの部門またはグループ間でデータを分離することで、部門またはグループは別の部門またはグループのレコードを表示できません。
2. データ侵害から保護するための追加のセキュリティレイヤーとして、コンテキストセキュリティとクエリー前のビジネスルールを追加します。ドメインセパレーションとビジネスルールの詳細については、「コンテキストとドメインセパレーション」および「クエリ前ビジネスルール」を参照してください。

3. ドメインセパレーションを使用して、会社に別のレベルのセキュリティを追加します。すべてのデータベースクエリーからのデータは、コンテキストセキュリティとビジネスルールが実行される前にドメインに表示されるデータに制限されます。
4. 個別のインスタンスを使用してデータベースとアプリケーションレイヤーでデータを分離します。

個別のインスタンス、ドメインセパレーション、コンテキストセキュリティとビジネスルール、および参照アーキテクチャがデータを分離する方法です。これらの 4 つの方法は、図のニーズの包括性矢印で示されているように互いに関連しています。各レイヤーが他のレイヤーとどのようにやり取りするかは、ドメインセパレーション構成の設定方法によって異なります。

すべての組織でドメインセパレーションが必要なわけではありません。個別のインスタンスやドメインのない単一のインスタンスなど、他の代替手段が見つかる場合があります。これらの代替方法の詳細については、「ドメインセパレーションの必要性の評価」を参照してください。

クロステナントインテリジェンス

マルチテナントアーキテクチャでは、単一のインスタンスが複数のテナントに対応します。テナントのデータ、メタデータ、ビジネスロジック、および処理コンテキストは、他のテナントデータにアクセスすることで自動的に処理されます。

単一テナントと複数テナント

単一テナントインスタンス

ServiceNow の顧客であるあなたは、ライセンスを購入したので、自分で必要なサービスを決定できます。必要に応じてアップグレードし、すべての優れた新機能のプレビューを確認して、すぐにインスタンスを設定できます。単一テナントには、次のメリットと制限があります。

- 初期コストと管理費は高くなりますが、自由に再ビルドや拡張ができます。
- インスタンスを取得して維持し、管理スタッフを用意すると、コストが高くなります。必要に応じて環境を自由にビルドできますが、ServiceNow の推奨プラクティスと標準に準拠する必要があります。

複数テナントインスタンス

他のユーザー (複数の顧客を持つサービスプロバイダー) がインスタンスを所有しています。必要に応じてアップグレードし、インスタンスに新しいサービスを提供します。あなたがサービスプロバイダーの顧客である場合、そのサービスプロバイダーが提供する製品を望んでいたため、そのサービスプロバイダーのインスタンスを使用している可能性が高くなります。複数テナントには、次のメリットと制限があります。

- 一元化されたスタッフが構成、統合、およびアップグレードを管理します。
- インスタンスオーナーが追加サービスを提供します。
- ドメインテナントは、ServiceNow プラットフォームを使用するための初期コストが低く、多くのテナントと共有しているため月次コストも低く、環境を管理するスタッフを雇用する必要がありません。
- 他のテナントが開始した要求または変更によりメリットが共有されます。

ドメインセパレーションの代替手段

顧客のドメインセパレーションの代わりに個別のインスタンスを使用できます。個別のインスタンスを使用すると、組織内のグループや部門内で他の人に影響を与えずにデータ分離の要件を柔軟に満たすことができます。

個別インスタンス

個別インスタンスの長所と短所

個別インスタンス	単一インスタンス - ドメインなし
長所	長所
各顧客/組織に合わせてビルド	単純なシナリオに対応可能
他のユーザーへのカスタマイズの影響を最小限に抑える	コスト
リリーススケジュールの調整	短所
明確な分離	ベースラインコードの大幅な変更
データセンターのある地域の選択	変更済みベースシステムコードがアップグレード中にスキップされる
短所	すべてのセカンダリテーブルとサポートテーブルにも対処する必要がある
コスト	広範なテストが必要
インスタンスの配置	カスタムコードを進化させる ServiceNow 製品チームがない
アップグレードのテストの労力	
労力の重複	
統合が必要	

インスタンスごとに個別にアップグレードとリリースの時間を設定できます。ただし、個別のインスタンスを使用する場合は、インスタンスを管理している他のユーザーと多くの調整を行う必要があります。コンテキストセキュリティ、フォームビュー、参照修飾子、フィルター、および堅牢な条件を使用してインスタンスを構成すると、会社でドメインセパレーションを使用する必要がなくなります。

個別のインスタンスを使用すると、データとプロセスの分離に対処できますが、インスタンス所有者は個別のインスタンスに必要な広範なカスタマイズを維持する必要があります。

関連トピック

- コンテキストとドメインセパレーション
- サービスプロバイダー参照アーキテクチャ

ドメインセパレーションの必要性の評価

ドメインセパレーションが顧客の組織で常に機能するとは限りません。顧客のニーズを考慮して、ドメインセパレーションを使用するかどうかを決定することをお勧めします。

ドメインセパレーションの必要性の評価

ドメインセパレーションを選ぶ理由

次の要因が、顧客の組織のドメインセパレーションを選ぶ際に役立つ可能性があります。

- 顧客のプロセス調整と一般的なプラットフォーム要件は中程度である。
- 顧客が要求者ではなく履行者としてタスクを処理することを計画している。

- 顧客がデータレコードの分離を要求する契約を結んでいるが、インスタンスオーナーは構成の別の場所でその要件に対処できると判断した。
- 会社のインスタンスオーナーは、物理的に分離された組織として運用され、データを共有しないエンティティを持つが、それでも完全なレポートを必要とする。別々のドメインを使用して、正しく設定すると、データを可視化できる。

ドメインセパレーションが設定されない理由

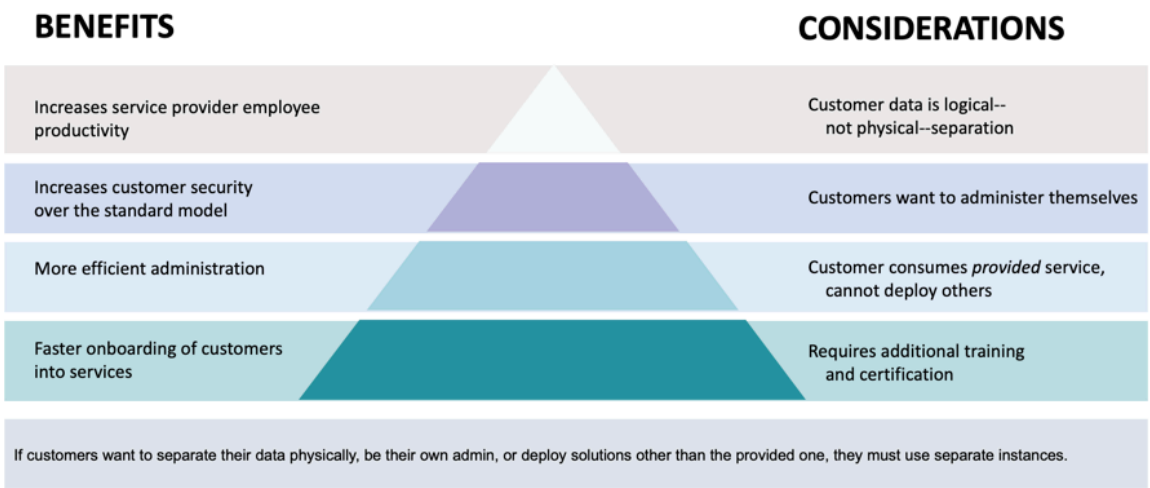
次の要因は、顧客の組織がドメインセパレーションを設定しない可能性がある理由を示しています。

- 顧客は環境を管理し、その完全な所有権を持ち、拡張のロードマップを設定することを望んでいる。
- 顧客は、物理またはデータベースレベルでデータとプロセスを完全に分離することを要求している。

注:

ドメインセパレーションされたインスタンスには共有データベースが含まれているため、これは分離要件に反しています。

- 顧客の組織内の部門がレコードを分離することを望んでいる(アクセス制御で十分な可能性がある)。
- すべての顧客が独自のプロセス、ビジネスルール、およびワークフローを必要としている。
- 企業文化は、顧客の組織間の非コラボレーションの 1 つである。
- 顧客はエンドユーザーとしてのみプラットフォームを操作する。

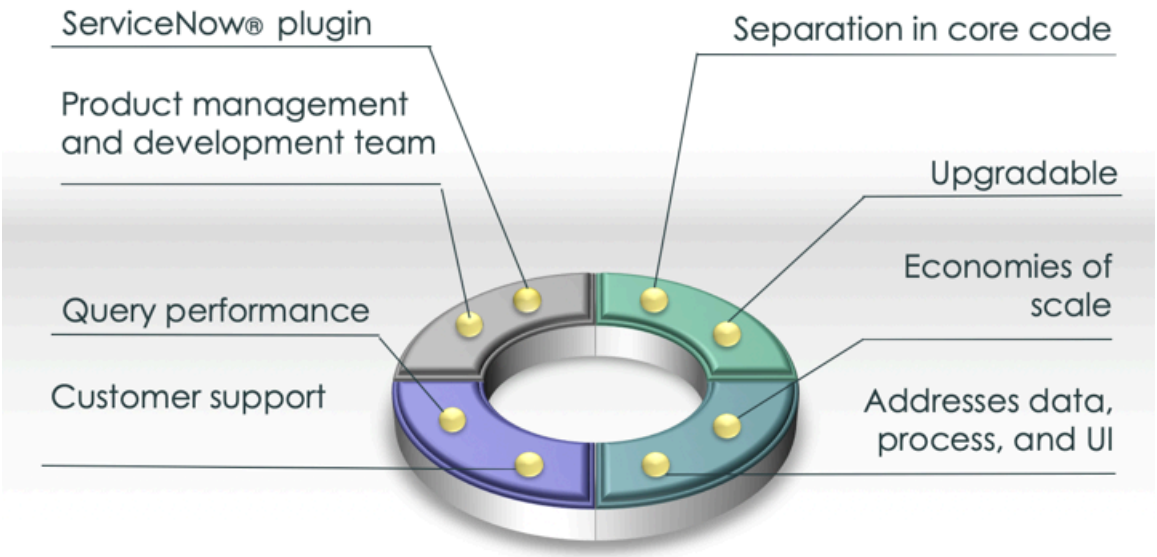


ドメインセパレーションのメリット

ドメインセパレーションは、グループや部門間でデータを分離する他の方法よりも顧客の組織で効果的に機能する可能性があります。

ドメインセパレーションのメリットの概要

コアプラットフォームに機能が組み込まれた ServiceNow プラグインを使用して、ドメインセパレーションを有効にすることができます。分離ドメイン構成は製品マネージャーが管理し、開発チームが製品マネージャーをサポートします。ServiceNow リリースには、ドメインセパレーション機能の拡張と修正が含まれており、顧客が使用できるようになりました。インスタンスオーナーは、ドメインセパレーションのサポートについて、サービスポータル など、カスタマーサービス & サポート リソースを参照できます。



ドメインセパレーションでのデータベースクエリーの仕組み

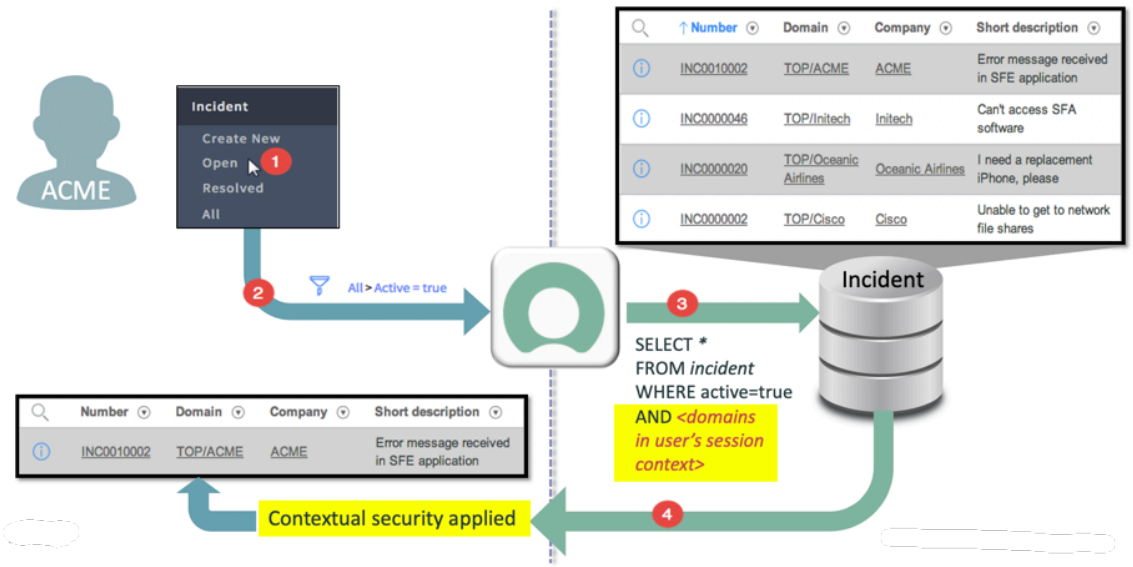
顧客のアプリケーションでドメインセパレーションを利用してデータベースクエリーを使用すると、顧客のデータを保護するのに役立ちます。これらのクエリーにより、構成とビルドのプロセスが迅速化されます。

ドメインセパレーションによるデータ保護方法

次の図には、インシデントテーブル [incident] に、インシデントのタスクから継承されたドメインフィールドがあります。このドメインフィールドが表示されている場合は、テーブル内のレコードにドメインが割り当てられていることがわかります。

ユーザーがログインすると、アクセス可能なドメインのセットとともにホームドメインが表示されます。これは、ユーザーのセッションコンテキストと呼ばれます。セッションコンテキストの詳細については、「コンテキストとドメインセパレーション」を参照してください。

ドメインセパレーションでのデータベースクエリー



1. ブラウザで、Acme 社のユーザーがオープンインシデントモジュールを選択し、active=true に設定されているすべてのインシデントを表示します。
2. active=true フィルターがアプリケーションに送信されます。
3. アプリケーションは、active=true に WHERE 句を追加して、データベースにクエリーを送信します。WHERE 句は、返されるインシデントレコードを、ユーザーのドメイン内にあるレコードまたはユーザーがアクセスできるドメインに制限します。これらのドメイン内のレコードのみが処理のためにアプリケーションに返されます。
4. コンテキストセキュリティが適用され、ユーザーに返されるデータがさらに制限されます。インシデントレコードがオープンインシデントリストに表示されます。

> **注:**
>
> コンテキストセキュリティを適用すると、ユーザーに返されるデータが制限されます。これらの制限は、ユーザーに表示しない他のコンテンツを保護します。

コンテキストセキュリティの詳細については、「コンテキストとドメインセパレーション」を参照してください。

注: この処理ロジックは、統合を使用してトリガーされるクエリーを含む、データベースに対するすべてのクエリーに適用されます。

ドメインセパレーションのサポートレベル

顧客の組織のアプリケーションのドメインセパレーションに対し、3 つのカテゴリから選択します。

ドメインセパレーションをサポートするアプリケーションは、データとデータルーティングの分離のみをサポートするか、高度なビジネスロジック分離機能があるか、アプリケーションのテナント (顧客) レベルの管理をサポートすることができます。これらの定義は、実際のユースケースとそれを実装するユーザーの観点からサポートレベルを示しています。

増分 **ServiceNow** サポートレベル

Basic	Standard	Enhanced
• Data is domain-separated • Logic exists to ensure proper data routing, caching, rollups, and aggregations • Global configuration operational for multiple tenants	• Application properties are domain-aware as needed • Business logic can be domain-separated by the instance owner per tenant	• Data-driven process enables failsafe configuration by tenants through the UI to drive business logic

レベル	タイプ	サマリー
サポートはありません		• ドメインフィールドがデータテーブルに存在している可能性がありますが、データを管理するビジネスロジックはありません。 • このレベルでは、ドメイン分割は考慮されません。

レベル	タイプ	サマリー
基本	顧客データ管理	• アプリケーションサービスプロバイダーのユースケースに合わせてデータが適切なドメインに送られるようにするビジネスロジックが存在します。 • アプリケーションでは、ユーザーインターフェイス、キャッシュキー、レポート、ロールアップ、集計など、すべてにおいて、実行時にドメインのプロパティが考慮されます。 • インスタンスの所有者は、複数のテナント間で正常に機能するようにアプリケーションを設定できる必要があります。 ユースケース：サービスプロバイダーがチャットを使用して顧客のメッセージに応答する際に、クライアントが応答を確認できるようにする必要があります。
標準	顧客プロセス管理	• ベーシックレベルを含みます。 • ビジネスロジック：サービスプロバイダーによって顧客ごとにプロセスを作成または変更できます。ユースケースには、単一のインスタンスで複数のサービスプロバイダー顧客がアプリケーションをどのように使用するかが反映されます。 • インスタンスのオーナーは、特定のアプリケーションに、顧客ごとに MVP (minimum viable product) ビジネスロジックとデータパラメーターを設定できる必要があります。 ユースケース：アドミニストレーターは、レコードを 1 人の顧客に対してはクローズしないが、別の顧客に対してクローズする場合、コメントを必須にすることができる必要があります。
拡張機能	顧客自己管理構成	• ベーシックレベルと標準レベルを含みます。 • サービスプロバイダーの顧客は定義されたユースケースに基づくビジネスロジックを変更できます。これらの構成は UI ベースでフェイルセーフであるため、1 人の顧客による構成が別の顧客に影響を与えることはありません。 • インスタンスの顧客は、自身で MVP ビジネスロジックとデータパラメーターを設定できる必要があります。 ユースケース：共有環境の顧客は、ドメイン内で影響度、緊急性、または優先度に基づいて変更できる必要があります。

レベル	タイプ	サマリー
有効なドメイン*		ドメインフレームワークが使用されていなくても、プラットフォーム機能またはアプリケーションがサービスプロバイダーのユースケースをサポートできる場合があります。ユースケースにはドメインセパレーションのサポートが詳述されている必要があります。サポートレベルの後のアスタリスク (*) は、この種類の構成を示します。 ユースケース：New York リリースより前は、サービスカタログ はドメインをサポートしていませんでしたが、インスタンスオーナーはユーザー基準を使用して、ドメインセパレーションされたインスタンス内の各テナントに個別のカタログとアイテムを設定できました。その結果、各テナントは標準レベルでサービスカタログを使用することができました。

サポートレベル別に一覧表示されたすべてのアプリケーションを表示するには、「アプリケーションでのドメインセパレーションのサポート」を参照してください。

サマリー

ドメインセパレーションは、アプリケーションで顧客を認識するために使用する必要があるフレームワークです。

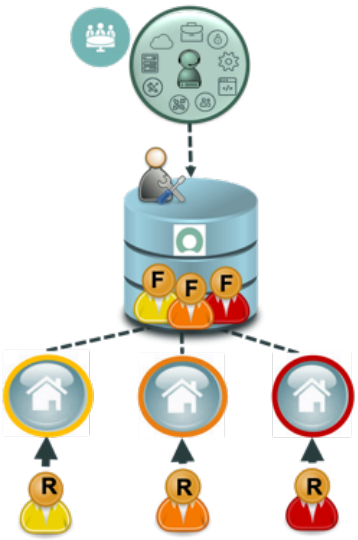
フレームワークを使用してアプリケーションをサポート可能にする前に、ドメインフレームワークの機能、アプリケーションのビジネスユースケース、ペルソナの概要、およびアプリケーションの使用方法を検討してください。

サービスプロバイダー参照アーキテクチャ

顧客は、ドメインセパレーションされたインスタンスに到達するために設計されたポータルを使用して、サービスプロバイダー (SP) サービスにアクセスできます。

サービスプロバイダー参照アーキテクチャの基本属性

- 履行者をドメインに割り当てません。代わりに、ドメイン間で共有します。これにより、ドメインあたりの履行者の数を監査することが困難になります。
- ドメイン管理を共有して活用できます。これは、オーバーヘッドがなく、ライセンスを最適化できることを意味します。
- 新しい顧客を取得すると、インスタンスのユーザー数が変更される可能性があります。新しい顧客を取得することで、システム上に数万人から数十万人の新規ユーザーが増える可能性があります。1 つの共有環境での合計ユーザー数は実質的に無制限です。



- SP customers access SP services via a portal to the SP Domain Separated instance
- SP uses ServiceNow shared instance(s) to manage their service delivery
- SP could have a shared instance per region to support data sovereignty requirements

Legend:

Centralized governance at the SP

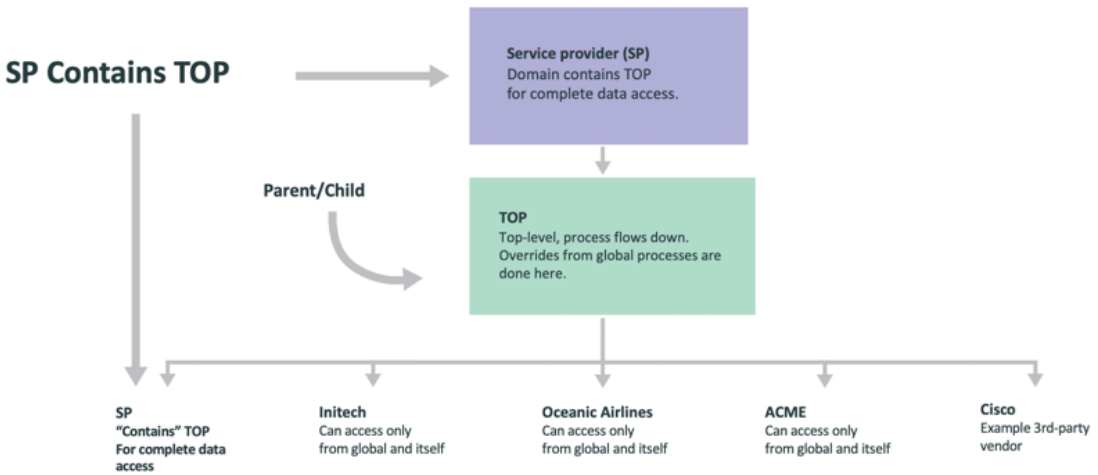
Centralized administration at the SP. Shared business requirements and configurations

Both SP and customer fulfillers on one instance

Customer requesters on each instance

SP サービスのポータルは、SP 共有インスタンス専用または共有です。サービスプロバイダーは、ServiceNow 共有インスタンスを使用してサービスデリバリーを管理します。

ドメインセパレーションされたインスタンスの参照階層



自動翻訳

SP 参照アーキテクチャの比較

Standalone (SA)		Multi-Tenant (MT)		Hybrid (MT+SA)		SIAM (Multi-Vendor)	
Platform as a Service	Enterprise scale	Processes simplified	Cost reduction	Managed Services (MT)	Custom Services (SA)	Service Integration	Multi-supplier governance
Customer wants full power of platform to Solve own needs outside those offered by SP.	Large customer: Transactions or data amount stored cause for multiple nodes or dedicated hardware.	Global governance & admin for business use case & process can be developed & maintained on the instance. Best when used w/ minor-to-moderate process differences among customers or sub-entities.	Lower administration, transaction, & onboarding costs, & ongoing operational costs for resources & licensing. Faster onboarding = more revenue.	Standardized re-usable services. procedures. processes, & resources delivered on a single ServiceNow instance to multiple Customers.	Services for sole use of customer, offerings specific to industry or vertical solutions, government, or regulatory requirements, applications that do not support Domain Separation.	Multiple best in class Service providers for individual services bringing together tooling, reporting, SLAs, strategy, & design for unified customer experience.	Unified governance across all providers to ensure organization requirements are met & that all providers cooperate on behalf of the customer.

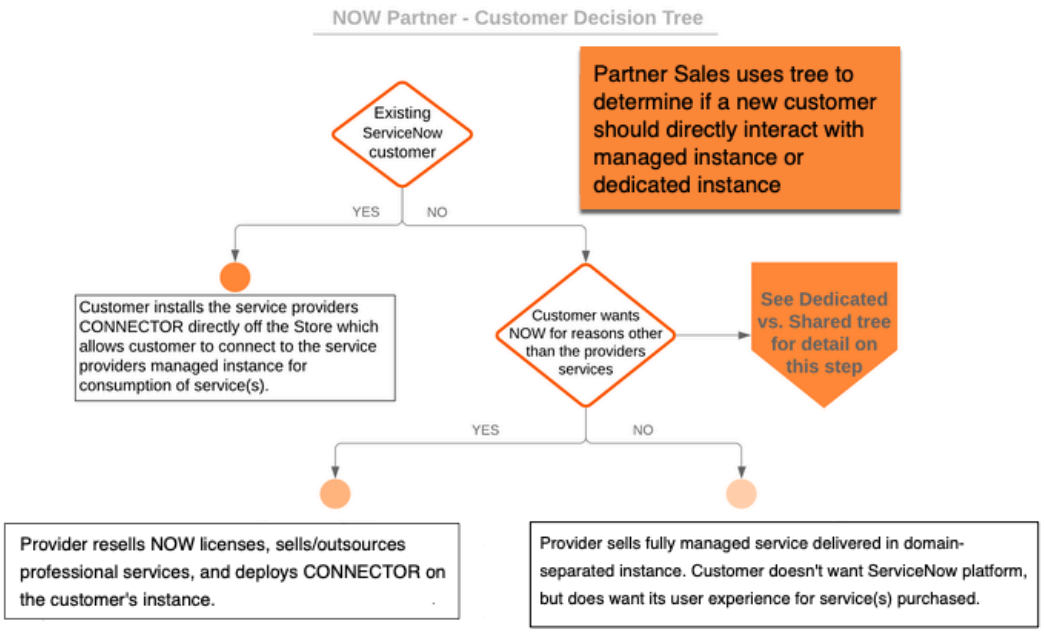
サービスプロバイダー参照アーキテクチャの意思決定ツリー

意思決定ツリーと比較チャートを使用して、新しい顧客を共有インスタンスまたは独自の専用インスタンスに追加する必要があるかどうかを判断できます。

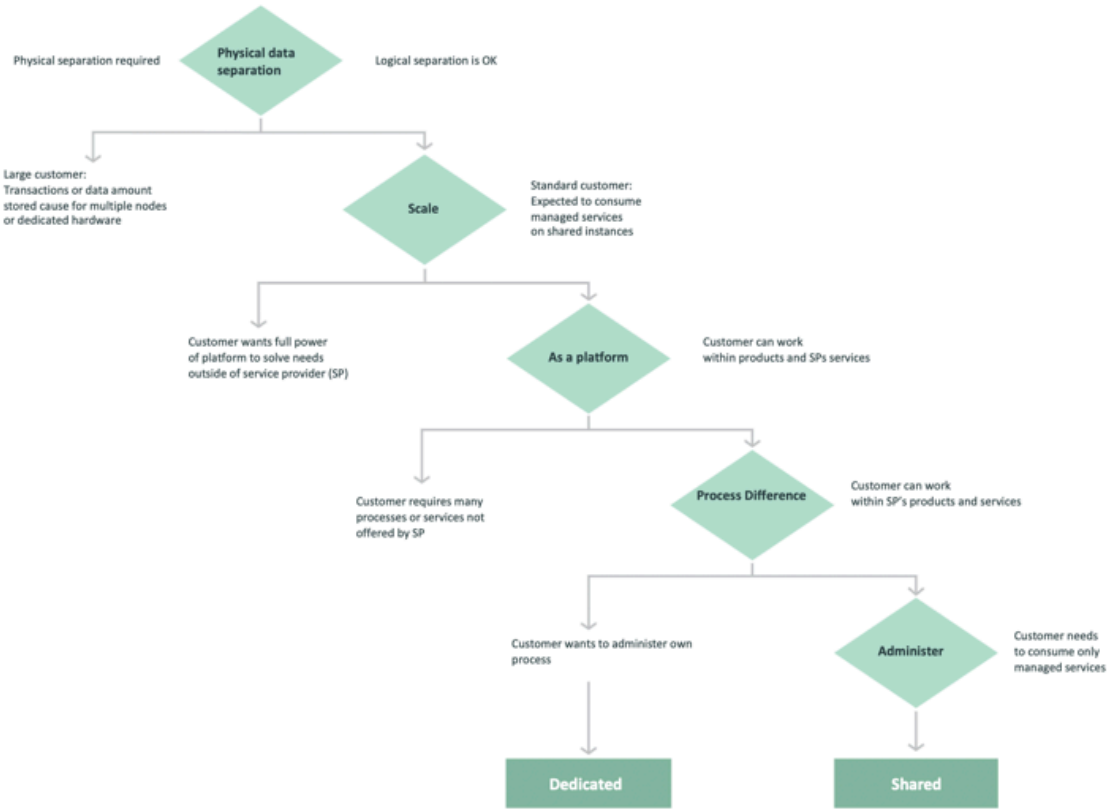
意思決定ツリー

これらの意思決定ツリーを使用して、顧客が管理対象インスタンスを使用するか専用インスタンスを使用するかを決定できます。

顧客の意思決定ツリー



自動翻訳



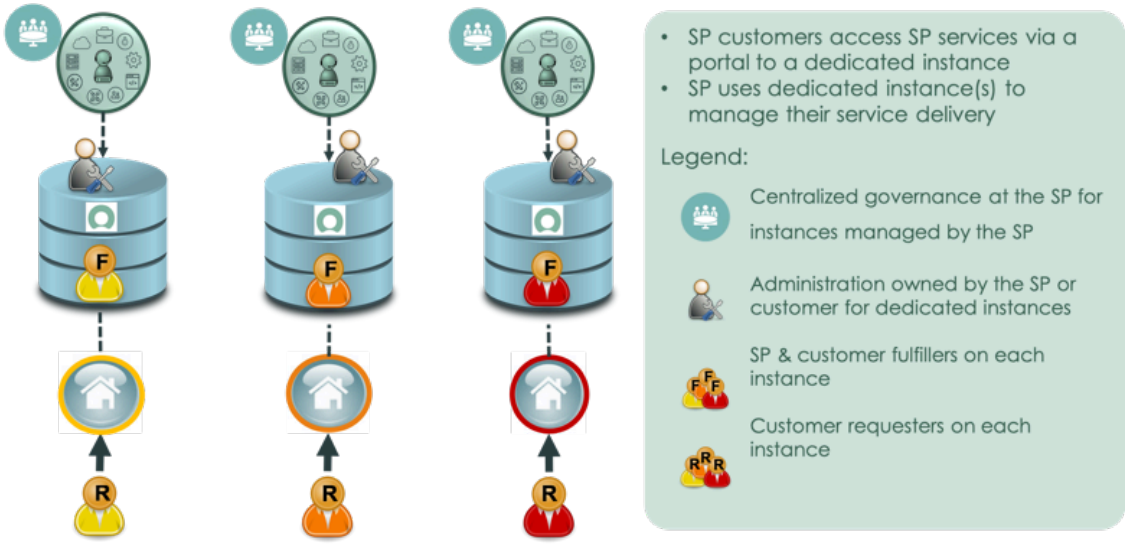
SP 参照アーキテクチャの比較

Standalone (SA)		Multi-Tenant (MT)		Hybrid (MT+SA)		SIAM (Multi-Vendor)	
Platform as a Service	Enterprise scale	Processes simplified	Cost reduction	Managed Services (MT)	Custom Services (SA)	Service Integration	Multi-supplier governance
Customer wants full power of platform to Solve own needs outside those offered by SP.	Large customer: Transactions or data amount stored cause for multiple nodes or dedicated hardware.	Global governance & admin for business use case & process can be developed & maintained on the instance. Best when used w/ minor-to-moderate process differences among customers or sub-entities.	Lower administration, transaction, & onboarding costs, & ongoing operational costs for resources & licensing. Faster onboarding = more revenue.	Standardized re-usable services. procedures. processes, & resources delivered on a single ServiceNow instance to multiple Customers.	Services for sole use of customer, offerings specific to industry or vertical solutions, government, or regulatory requirements, applications that do not support Domain Separation.	Multiple best in class Service providers for individual services bringing together tooling, reporting, SLAs, strategy, & design for unified customer experience.	Unified governance across all providers to ensure organization requirements are met & that all providers cooperate on behalf of the customer.

専用インスタンスのサービスプロバイダー参照アーキテクチャ

サービスプロバイダー (SP) の顧客は、専用インスタンスへのポータルを使用して SP サービスにアクセスできます。SP は、 専用インスタンスを使用してサービスデリバリーを管理します。

専用インスタンス

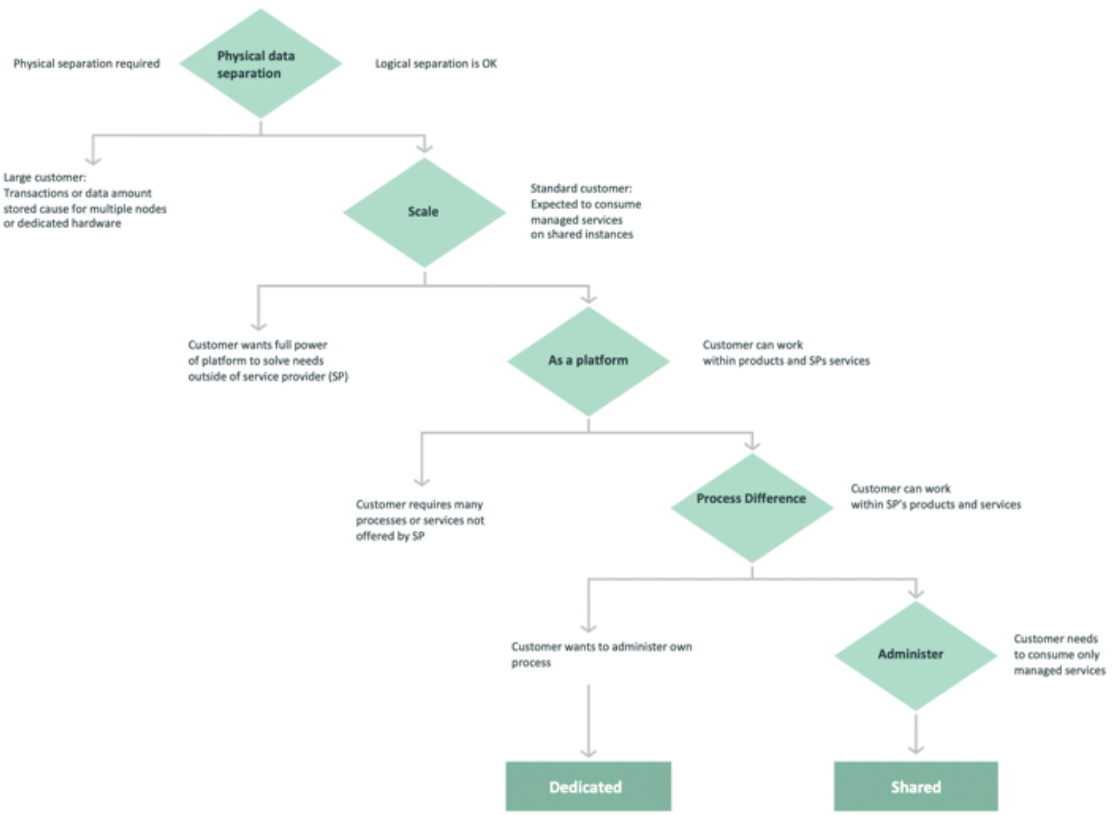


属性

- 専用インスタンスには、個別の管理チームと専門チームが必要です。複数のインスタンスにログインするアドミニストレーターと開発者には、複数のライセンスが必要です。
- 各インスタンスには、限られた数の要求者と履行者がいます。新規顧客を獲得する場合は、会社の規模とスケールに基づいてインスタンスを調達する必要があります。

共有インスタンス

専用インスタンスと共有インスタンス



SP 参照アーキテクチャの比較

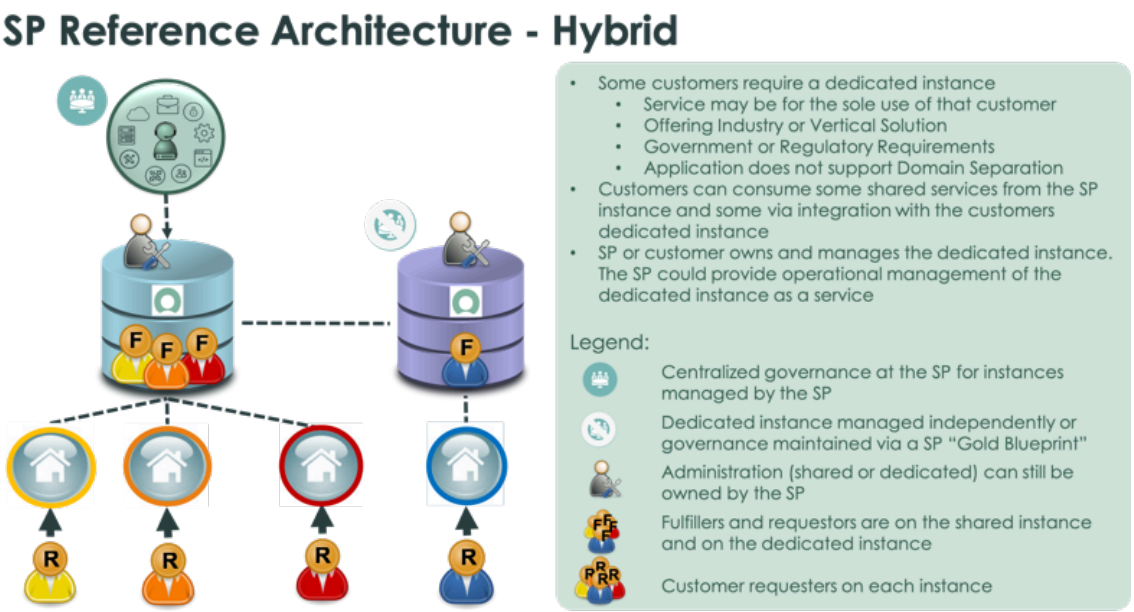
Standalone (SA)		Multi-Tenant (MT)		Hybrid (MT+SA)		SIAM (Multi-Vendor)	
Platform as a Service	Enterprise scale	Processes simplified	Cost reduction	Managed Services (MT)	Custom Services (SA)	Service Integration	Multi-supplier governance
Customer wants full power of platform to Solve own needs outside those offered by SP.	Large customer: Transactions or data amount stored cause for multiple nodes or dedicated hardware.	Global governance & admin for business use case & process can be developed & maintained on the instance. Best when used w/ minor-to-moderate process differences among customers or sub-entities.	Lower administration, transaction, & onboarding costs, & ongoing operational costs for resources & licensing. Faster onboarding = more revenue.	Standardized re-usable services. procedures. processes, & resources delivered on a single ServiceNow instance to multiple Customers.	Services for sole use of customer, offerings specific to industry or vertical solutions, government, or regulatory requirements, applications that do not support Domain Separation.	Multiple best in class Service providers for individual services bringing together tooling, reporting, SLAs, strategy, & design for unified customer experience.	Unified governance across all providers to ensure organization requirements are met & that all providers cooperate on behalf of the customer.

ハイブリッドのサービスプロバイダー参照アーキテクチャ

カスタマイズされたソリューションには、ハイブリッドサービスプロバイダー (SP) の参照アーキテクチャを使用します。顧客には特定のサービスの専用インスタンスが必要です。他のサービスに対しては共有 SP インスタンスを引き続き使用できますが、各インスタンスの統合が必要です。

ハイブリッドアーキテクチャ

顧客がこの追加サービスを直接提供する責任を負う場合があります。顧客が提供する複数の属性のハイブリッドソリューションをビルドすることが必要になります。



自動翻訳

属性

- インスタンスの管理を共有して活用できます。これは、オーバーヘッドがなく、ライセンスを最適化できることを意味します。
- 分散環境に新しいインスタンスがある場合、プログラムチームはそのインスタンスの専用アドミニストレーターユーザーとして責任を負い、それに応じて資金が提供されます。すべてのインスタンスが詳細計画に由来する一元化された環境では、複製した管理ライセンスが必要です。
- 履行者をドメインに割り当てません。代わりに、ドメイン間で共有できます。
- 顧客が共有環境と専用環境の両方を共有している場合は、両方の環境で履行者が必要です。共有インスタンスと専用インスタンスのプロセスではインスタンスごとに異なる作業が必要になるため、各チームの作業量が増加します。
- 新しい顧客を取得すると、インスタンスのユーザー数が変更される可能性があります。新しい顧客を取得することで、システム上に数万人から数十万人の新規ユーザーが増える可能性があります。1 つの共有環境での合計ユーザー数は実質的に無制限です。
- 各インスタンスには、限られた数の要求者と履行者がいます。新規顧客を獲得する場合は、会社の規模とスケールに基づいてインスタンスを調達する必要があります。

SP 参照アーキテクチャの比較

Standalone (SA)		Multi-Tenant (MT)		Hybrid (MT+SA)		SIAM (Multi-Vendor)	
Platform as a Service	Enterprise scale	Processes simplified	Cost reduction	Managed Services (MT)	Custom Services (SA)	Service Integration	Multi-supplier governance
Customer wants full power of platform to Solve own needs outside those offered by SP.	Large customer: Transactions or data amount stored cause for multiple nodes or dedicated hardware.	Global governance & admin for business use case & process can be developed & maintained on the instance. Best when used w/ minor-to-moderate process differences among customers or sub-entities.	Lower administration, transaction, & onboarding costs, & ongoing operational costs for resources & licensing. Faster onboarding = more revenue.	Standardized re-usable services. procedures. processes, & resources delivered on a single ServiceNow instance to multiple Customers.	Services for sole use of customer, offerings specific to industry or vertical solutions, government, or regulatory requirements, applications that do not support Domain Separation.	Multiple best in class Service providers for individual services bringing together tooling, reporting, SLAs, strategy, & design for unified customer experience.	Unified governance across all providers to ensure organization requirements are met & that all providers cooperate on behalf of the customer.

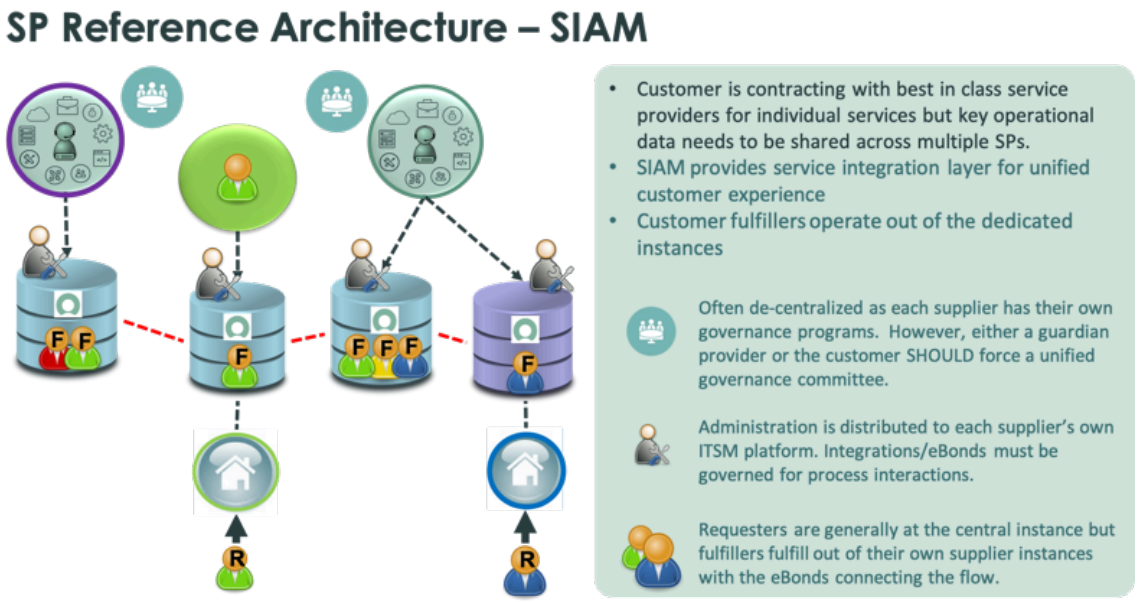
サービスインテグレーション管理 **(SIAM)** のサービスプロバイダー参照アーキテクチャ

サービスプロバイダー (SP) アーキテクチャの サービスインテグレーション管理 サービスインテグレーションおよびマネジメント (SIAM) は、統一されたカスタマーエクスペリエンスのためにサービスを統合します。

SIAM アーキテクチャの属性

- SP サービスの顧客は、SP 共有インスタンス専用または共有ポータルです。
- SP は、ServiceNow 共有インスタンスを使用してサービスデリバリを管理します。

SIAM アーキテクチャの概要



SP 参照アーキテクチャの比較

Standalone (SA)		Multi-Tenant (MT)		Hybrid (MT+SA)		SIAM (Multi-Vendor)	
Platform as a Service	Enterprise scale	Processes simplified	Cost reduction	Managed Services (MT)	Custom Services (SA)	Service Integration	Multi-supplier governance
Customer wants full power of platform to Solve own needs outside those offered by SP.	Large customer: Transactions or data amount stored cause for multiple nodes or dedicated hardware.	Global governance & admin for business use case & process can be developed & maintained on the instance. Best when used w/ minor-to-moderate process differences among customers or sub-entities.	Lower administration, transaction, & onboarding costs, & ongoing operational costs for resources & licensing. Faster onboarding = more revenue.	Standardized re-usable services. procedures. processes, & resources delivered on a single ServiceNow instance to multiple Customers.	Services for sole use of customer, offerings specific to industry or vertical solutions, government, or regulatory requirements, applications that do not support Domain Separation.	Multiple best in class Service providers for individual services bringing together tooling, reporting, SLAs, strategy, & design for unified customer experience.	Unified governance across all providers to ensure organization requirements are met & that all providers cooperate on behalf of the customer.

自動翻訳

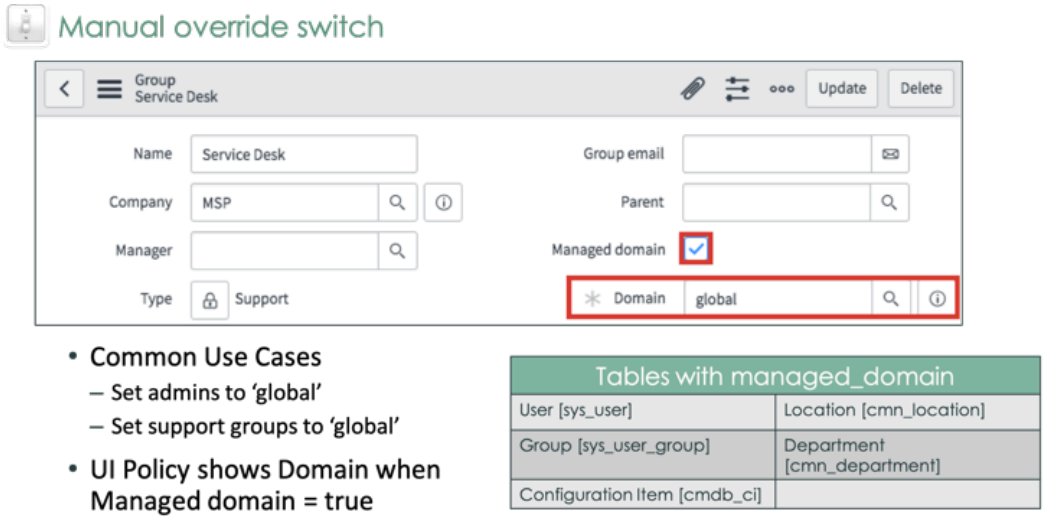
ドメインセパレーションの用語

ServiceNow インスタンスを使用すると、効率を向上させ、セキュリティを強化し、顧客の組織のパフォーマンスを向上させることができます。最も一般的な用語を理解しておくと構成を作成する際に役立ちます。

管理対象ドメイン

管理対象ドメインの [管理対象ドメイン] フィールドで、ドメインアドミニストレーターは、会社レコードから自動的に割り当てられたドメインを使用するのではなく、ユーザー、グループ、部門、場所、または CI レコードのドメインを手動で選択できます。

これらのプロパティを変更する場合は、上書きして、各ドメイン内のアプリケーションの機能をさらにカスタマイズできます。



プロセステーブル

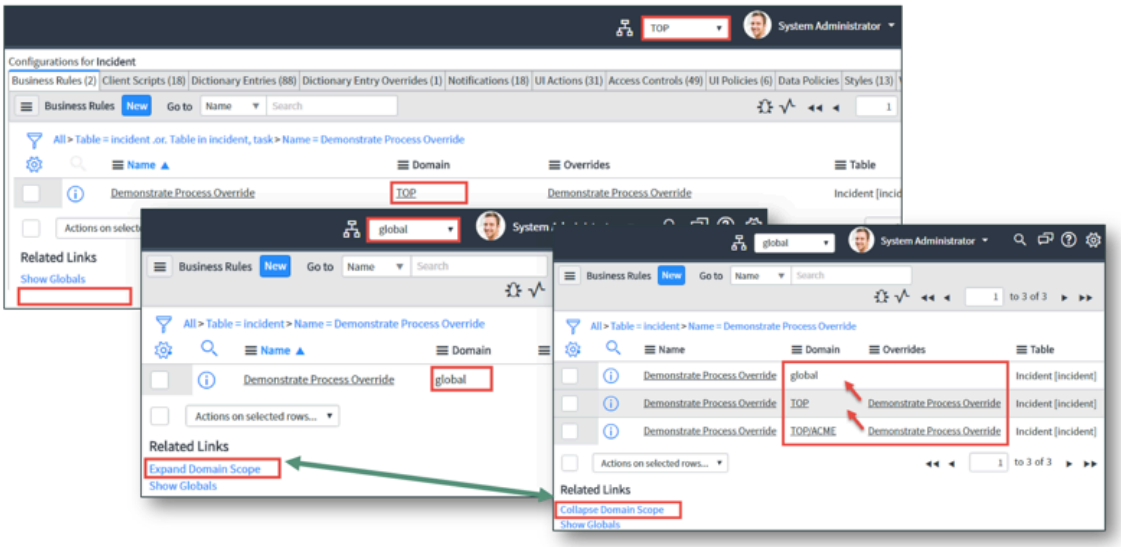
プロセステーブルで上書き **[sys_overrides]** フィールドに値が表示されている場合は、プロセス上書きレコードが存在します。これは、アドミニストレーターがドメイン固有のポリシーを設定できる委任管理が有効であることを意味します。グローバルドメインのアドミニストレーターは、ドメインスコープの展開/折りたたみ関連リンクを使用して上書きレコードを表示できます。

注: レポートはドメインに分離され、[上書き] フィールドが含まれています。グローバルドメインからすべてのレポートを表示するには、[ドメインスコープを展開] 関連リンクを使用します。

ドメインからプロセステーブルを表示すると、選択したドメインの関連するプロセスレコードのみが表示されます。グローバルドメインからプロセステーブルを表示すると、[ドメインスコープを展開] 関連リンクが表示され、上書きを含むすべてのプロセスレコードを表示できます。グローバルに関連するプロセスレコードのみを再表示するには、[ドメインスコープを折りたたむ] 関連リンクを使用します。

ドメインスコープ機能はプロセステーブルにのみ使用され、テーブルのデータの可視化が逆方向に変わります。たとえば、親ドメインのレコードは子で表示できますが、親は子レコードを表示できません。これにより、プロセスが子ドメインに移行します。

Overrides [sys_overrides] – Process Tables Only



ドメインのタイプ

さまざまなタイプのドメインは、プロセスとデータ、およびそれらがアプリケーションまたは機能でどのように機能するかを整理するのに役立ちます。

顧客ドメイン

顧客のドメインには、ユーザーインターフェイスと、データの使用方法を制御するプロセスがあります。

次の画像の ACME ドメインは顧客ドメインです。

プロセスドメイン

データの使用方法とドメイン内で実行することのプロセスを作成します。このプロセスには次の属性が必要です。

- 一連のドメインに対する特定のプロセスと UI 設定
- いかなる種類のコアデータ (特定のユーザーデータなど) もありません。
- 次の画像の TOP ドメインはプロセスドメインです。

データドメイン

データドメインは、複数の顧客に関連するデータを保持します。実際の顧客ドメインを共有せずにそのデータを共有できます。各顧客は独自のデータドメインを持ち、そのドメインにアクセスできます。

注:

この種のドメインは一般的ではなく、過度に使用するとパフォーマンスの問題が発生する可能性があります。使用する前に SP アーキテクトに相談してください。

たとえば、ドメインは、ACME、Cisco、および SP のすべてがやり取りする必要があるタスクを保持できます。

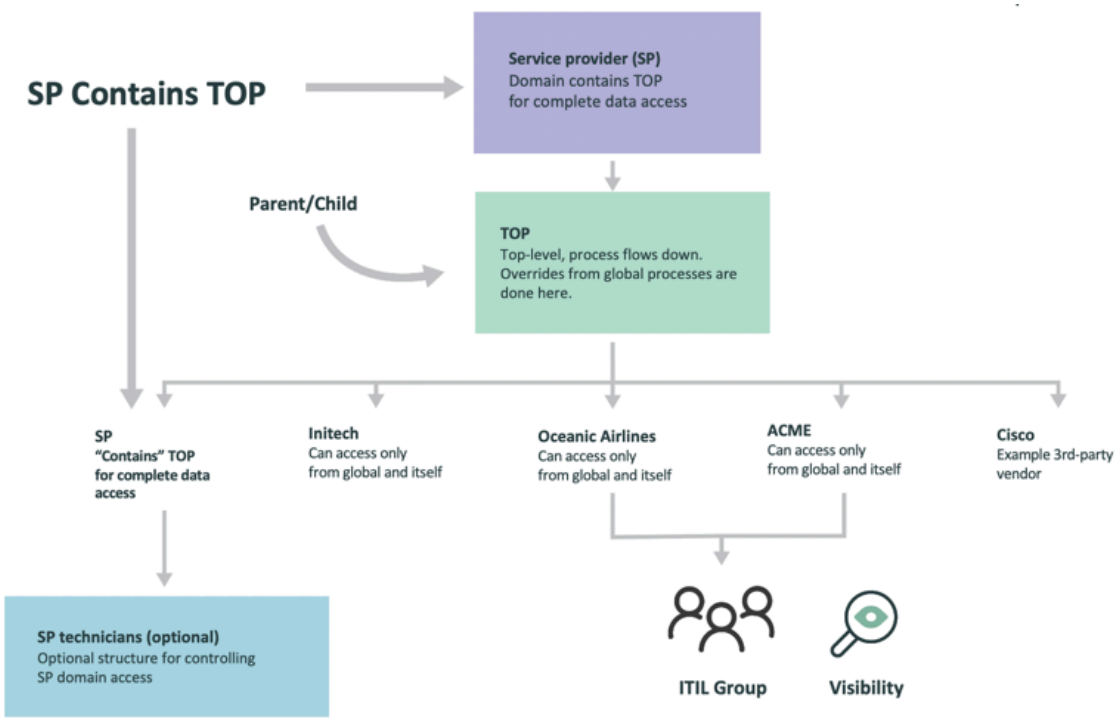
次の画像のデフォルトドメインはデータドメインです。

ユーザーデータ

ユーザーレコードデータがグローバルドメインまたはプロセスドメインに属することはありません。ユーザーは主に顧客ドメインで作成され、場合によってはデータドメインで作成されます。

admin アカウントは、インスタンスの日常のユーザーとして使用するべきではなく、管理機能を容易にするためにグローバルドメイン内にある必要があるため、特別なアカウントです。

ドメイン階層



リスト、admin、グローバルプロセス

リスト

グローバルドメインから、任意の選択フィールドのラベルを右クリックし、[選択を設定] を選択して新しい選択肢を追加すると、そのフィールドのすべてのドメイン固有のリストに選択肢が自動的にプッシュされます。新しいオプションが [選択済み] としてマークされている場合は、アクティブとして追加されます。新しいオプションが [利用可能] としてマークされている場合は、非アクティブとして追加されます。

インスタンス管理

インスタンスオーナーのアドミニストレーターは、ドメインセパレーションされたインスタンスですべての通常のプロセスの作成、変更、およびメンテナンスを処理する必要があります。個々のドメインマネージャーは、データ駆動型プロセスの一部を維持できます。ドメインマネージャーのタイプは、ユーザー管理、サポートグループメンバーシップ、および場所を維持するか、テナント管理を念頭に置いて設計されたアプリケーションを管理します。

グローバルプロセス/パラメーター

グローバルドメインに影響を与えるプロセスを作成および管理したり、パラメーターを設定したりすることができます。これらのプロパティは、ドメインセパレーションされたインスタンスのすべてのユーザーに共通です。

例：システムプロパティ、辞書オーバーライド、sys_documentation (フィールドラベル)、データモデル (クラス、CI タイプなど)、テーブルとフィールド [sys_dictionary] (アクセスを制限可能)、インデックス作成 (テキストインデックスおよびデータベース) 、ACL、インストールイグジット、受信アクション、公開ページ、およびインターセプター。

カスタムテーブルをドメインセパレーションする

個別のドメインにカスタムテーブルを作成する必要がある場合があります。このトピックでは、カスタムテーブルのドメインセパレーションの背後にある手順と概念の両方について説明します。

1. sys_domain フィールドを作成します。

注: システムテーブルまたはテーブルが ドメインセパレーションプラグインによってドメインセパレーションされていない場合は、ドメインセパレーションしないことをお勧めします。

これらのポイントを指針として使用して、sys_domain フィールドを作成します。

- 新しいフィールドを *domain_id* タイプとして作成します。
  - 列名：sys_domain
  - その他の属性：自動的に定義
- sys_domain_path が自動的に作成されます。

列名「**sys_domain**」は ServiceNow AI Platform で予約されています。これは、システムがこれを認識し、適切なフィールドタイプと属性を自動的に適用することを意味します。この自動構成では、対応する **sys_domain_path** フィールドも作成されます。

- ラベルを使用するのではなく、列名を「sys_domain」に設定します。
- ドメインセパレーションはすべてのテーブルに適しているわけではありません。一般に、テーブルがベースインスタンスの一部であり、そのテーブルに **sys_domain** フィールドがない場合は、そのままにしておく必要があります。

「sys_domain」という domain_id タイプのフィールドを作成すると、**sys_domain** フィールドが自動的に作成されます。

2. ドメインを設定するビジネスルールを追加します。

ビジネスルールなし

ドメインは、レコードを作成するユーザーの現在のドメインに設定されます。

ビジネスルールあり

ドメインは、通常は [会社] フィールドに基づいて、スクリプト化されたロジックを使用して割り当てられます。

sys_domain フィールドに加えて、カスタムテーブルには、ドメインフィールドの値を設定するドメイン - ドメインを設定 - タスクに似たビジネスルールが必要です。さらに、最初のルールがドメインの割り当てに失敗した場合に、ドメインのないレコードをデフォルトドメインに移動するドメイン - デフォルト - タスクが必要です。

タスクテーブルで、ドメインのビジネスルールを確認します。[順序] フィールドに特に注意してください。実行の優先度は、低から高まで、[順序] フィールドによって指定されます。

実行される最初のドメイン - ドメインを設定 - タスクルールは、レコードの会社のドメインに基づいてレコードのドメインを設定しようとします。

最初のルールで適切なドメインが見つからない場合は、2 番目のドメイン - デフォルト - タスクルールが実行されます。このルールは、レコードのドメインをデフォルトのドメインに設定します。

最後に、タスクレコードのドメインが変更されると、ドメイン - カスケードドメイン - タスクビジネスルールによって、ワークフロー、メトリクス、SLA、添付ファイルなどのタスクに関連するすべてのレコードのドメインが変更されます。

3. ステップ 2 が失敗した場合のビジネスルールを追加します。

最初のビジネスルールがドメインの設定に失敗し、ドメインがまだ空またはグローバルである場合は、2 番目のビジネスルールが実行されます。このルールは、caller または requested_for フィールドに基づく task_for フィールドを調べます。このルールは、ユーザーのドメインに基づいてレコードのドメインを設定できるかどうかを確認しています。設定できない場合は、ビジネスルールによってドメインがデフォルトドメインに設定されます。

ビジネスルールのサンプルスクリプトは次のとおりです。

```
/* essentially
If (task_for is set)
  set the domain to the user's domain
ELSE
  set the domain to the default domain
*/
```

4. ドメイン - カスケードドメイン - タスク

タスクには、事業達成目標のために連携する多くの関連テーブルを含めることができます。これらの関連レコードには、ワークフロー、SLA、承認、添付ファイル、およびメールが含まれます。タスクのドメインが変更された場合は、関連レコードのドメインも変更して、新しいドメイン内のユーザーに表示されるようにする必要があります。

このカスケードルールは一般に、デフォルトのドメインからレコードを消去するときにトリガーされます。

スクリプトに含まれるカスケードドメインの関連レコードは、次の例のように表示されます。

```
/*
* Keep domains in sync w/related records for:
* workflow context
* workflow history
* approver tables and related workflows
* attachments
* emails
*/
```

ドメインのプロパティとテーマのカスタマイズ

構成したドメイン内の顧客の会社プロパティとテーマをカスタマイズできます。カスタマイズにより、インスタンスを会社の全体的なルックアンドフィールに合わせることができます。

会社のテーマとロゴのカスタマイズ

会社レコードでは、会社ごとに特定のデザインテーマとロゴをカスタマイズできます。

ドメインをカスタマイズする

デフォルトでは、サービスプロバイダープラグインがインストールされている場合、コアテーブルに含まれる標準ビジネスルールにより、レコードのドメインがレコードに関連付けられている会社のドメインに設定されます。会社は、タスク、ユーザー、グループ、場所、部門などのテーブルを制御できます。

タスクテーブルを除くこれらすべてのテーブルで、レコードが作成されるドメインを上書きできます。これにより、より多くのカスタマイズオプションが提供されます。

特定の用途のためのドメインセパレーションの管理

メール通知に個別のドメインを設定し、カタログ、テーブル、ユーザー、グループ、およびビューのプロパティをカスタマイズすることができます。これにより、各ドメインでより具体的な動作を提供できるため、顧客に柔軟性を提供できます。

メール

メール通知と上書きに個別のドメインを使用できます。通知に個別のドメインを使用する場合、ユーザーのドメイン全体ではなく、添付されたレコードのドメインに基づいて上書きすることができます。

サービスカタログ

サービスカタログはドメインセパレーションされているため、顧客はカタログを表示してアクセスできます。複数のアイテムが使用される場合、アイテムは OR 条件として処理されます。サービスプロバイダーは、独自の基準に合うようにカテゴリとアイテム自体を管理する必要があります。

ユーザーとグループ

アドミニストレーターはすべてのドメインにアクセスする必要があるため、グローバルドメインでのみ admin アカウントを使用してください。グローバルドメインではなく、実際のドメインからすべてのアプリケーションテストを実行します。上書きはグローバルドメインで適切に処理されません。アドミニストレーターがアプリケーションを使用する場合は、本番環境のユーザーアカウントも指定する必要があります。

フィールドの操作

フィールドを操作する場合は、考慮すべき点がいくつかあります。これらのフィールドには構成に影響する多くのバリエーションが含まれている可能性があるため、注意してください。

リスト
: 個人リスト、グローバルリスト、ドメインリスト、およびそれぞれの複数のビューがあります。

フォーム
: グローバルリストとドメインリスト、およびそれぞれの複数のビューがあります 。

1 つのデータベース
: 作成したフィールドはすべて、すべてのユーザーに対して 1 つのデータベース内に存在します。グローバルな影響度を考慮してから作成してください。

注: ACL スクリプトは実行されないため、フィールドをリストに表示しないようにすることはできません。ACL がロールベースのみの場合は、READ ACL を追加してユーザーからフィールドを非表示にすることができます。

テーブルの作成

テーブルを作成するときは、sys_domain または sys_overrides フィールドを追加する必要があります。インスタンスユーザーがアクセスする必要があるデータを含むテーブルには、sys_domain フィールドが必要です。プロセスを拡張またはサポートし、子ドメインに移行する必要があるテーブルにも sys_domain フィールドが必要です。

ドメインピッカーを使用したドメインセパレーションの構成

ドメインピッカーを適切に使用し、80/15/5 のアプローチに留意して、インスタンスの過剰なカスタマイズとパフォーマンスへの影響を避けるようにしてください。

変更を行う前にドメインを検証する

ドメインピッカーは、選択できるように、すべてのドメインをリストに収集します。

セッションがタイムアウトすると、ログアウトしていない場合でも、セッションはユーザーレコードのドメインにフォールバックします。同時に、昇格したロールもすべて失われます。この場合、リストの一番上のフレームが再ロードされていなければ、ドメインピッカーには最後に選択したドメインが表示される可能性があります。そのため、一時的にインスタンスから離れている場合は、リストを完全に再ロードする必要があります。

TOP ドメインまたはグローバルドメインでの設定

ドメインセパレーションは、構成およびユーザーとグループの定義がほぼ標準であるサービスを顧客に提供する場合に最適です。カスタマイズして「ワンオフ」ソリューションを作成すればするほど、エラーに対するマージンが増加します。プロセスとビジネスロジックを作成するときは、各顧客に対して自動的に機能するプロパティにバリエーションを設定する必要があります。プロセスは必要に応じて調整できますが、1 人の顧客に一意の構成を作成するタイミングとその程度を決定する際は十分注意してください。

カスタマイズのためのマージンが大きすぎてエラーが発生しないようにするには、ドメインを構成する際に「80-15-5」アプローチを使用する必要があります。

- 構成に推奨されるアプローチ
  - **80%** 以上の標準
  - **15%** 以上のパラメトリック
  - **5%** 未満の構成
- 提案された変更をグローバルプロパティにするか、構成可能なプロパティにするかを決定します。
- 管理が必要なカスタマイズを増やしてビルドしすぎないでください。代わりに、次の操作を行います。
  - ベースシステム機能から開始し、変更を加える前にギャップを確認します。
  - コードなしのソリューションを探します。
  - サーバー側のスクリプトを使用して、モジュール式の API をビルドし、ドメインセパレーションされたプロパティをビルドします。
  - クライアントスクリプトを使用する必要がある場合は、ServiceNow API のみを使用してください。「同期」呼び出し (クライアントとサーバー間でやり取りをするもので、AJAX とも呼ばれます) を制限します。
  - すべてのスクリプトを論理的に記述して、シンプルで効果的なものにします。コード変更のピアレビューを実施し、全員がこのセクションの「サービスプロバイダー向けのドメインセパレーションの推奨プラクティス」に従っていることを確認します。

ドメインセパレーションのパフォーマンスに関する考慮事項

アプリケーションとサービスでドメインセパレーションを構成するときは、作成するドメインの数とプロパティを必ず考慮してください。プロパティの負荷が大きいドメインが多すぎると、インスタンスのパフォーマンスに影響を与える可能性があります。

プロパティの負荷が大きいドメインの制限

必要な数のドメインを作成できますが、そのインスタンスに不要なドメインを作成しないようにします。多数のプロパティを持つドメインがインスタンス上に多すぎると、インスタンスのパフォーマンスに影響を与える可能性があります。

パフォーマンスに影響を与えるのはドメインに含まれているものではなく、ドメインの数です。プロパティが多すぎると、ドメインピッカーの速度が低下し、顧客の全体的なユーザーエクスペリエンスが低下します。ドメインピッカーをロードしているとき、多くのプロパティを持つ多数のドメインが既に存在する場合、ドメインピッカーはすべてのドメインをロードしてから、セッションのコントロールをユーザーに渡す必要があります。このプロセスにより、ドメインピッカーが終了するまでインスタンス上のものにアクセスできなくなる可能性があります。新しいドメインを作成する前に、下のドメイン階層に移動します。 ドメイン管理 > ドメインマップ 実際に新しいドメインを作成する必要があるか、既存の ドメイン階層 が機能するかを確認します。

コア **UI** ドメインピッカーの使用

ドメイン参照ピッカーは コア UI で使用できます。参照ピッカーでは、一度にすべてのドメインをロードするのではなく、ドメインの名前をドメインピッカーに入力し始めるとドメインが検索されます。

次の手順に従って、コア UI でドメイン参照ピッカーを有効にします。

1. アプリケーションナビゲーターで、「sys_properties.list」と入力します。
2. *glide.ui.domain_reference_picker.enabled* プロパティを true に設定します。
3. ブラウザを更新します。

> **注:** 最初にテストを行わずに統合またはインポートセットを使用して多数のドメイン (30 以上) をアップロードしないでください。インスタンスが停止する可能性があります。

ドメイン階層の設定

ドメイン階層がどのように機能するかを理解し、適切に設定することで、インスタンスの速度低下やパフォーマンスへの影響を回避できます。

ドメイン階層に基づいて、ユーザーはホームドメインと子ドメインのデータにアクセスできます。プロセスは子ドメインに向かって下降し、データは上昇します。

既存のドメイン階層の変更は必要な場合にのみ行ってください。ドメインの親を更新すると、すべての子ドメインとともに親ドメインが再確立され、それによってドメイン階層が変化します。ドメイン階層が更新されると、そのドメインに作成されたレコードのドメインに関連するすべてのテーブルでカスケード更新がトリガーされます。そのため、多数のサポートテーブルも更新する必要があります。

同じ理由から、ドメイン階層を変更する必要がある場合でも、一括更新は行わないでください。ドメイン階層を変更するためにシステムが実行する必要があるクエリーの数を想定してください。更新は常に小さなバッチで行います。次の更新バッチを開始する前に、ドメイン作業要求 (DWR) レコードが処理されていることを確認してください。DWR は、ドメイン階層を変更した後にエラーがあるかどうかを表示するレポートです。

DWR レコードの追跡

syslog_domain テーブルで、**DWR** の実行が完了したことを示すメッセージ列の情報エントリを探して、DWR が完了したことを確認します。

ドメインログでのエラーと警告の確認

ドメインログを確認して、ドメインパスプロセスと階層構成のエラーまたは警告を特定します。

ドメインログは、ドメインログ [syslog_domain] テーブルにあります。ドメイン階層が更新されると、ドメインパスを再計算するスケジュール済みジョブがトリガーされます。ドメインログテーブルは結果をキャプチャします。

このテーブルでエラーと警告を探します。このテーブルを確認した後、これらのエラーを解決し、ドメインパス検証ツールを再度実行する必要があります。

このログの例では、sys_ui_list テーブルで 10 件の孤立レコードが検出されました。ドメインパスを正常に実行するには、これらのレコードのエラーを修正する必要があります。

Error	10 records detected in 'sys_ui_list' that are not in any existing domain and Domain Paths. Fix the domain value in these records & run validator again.	com.glide.domain.validator
Error	10 records detected in 'sys_ui_list' that are not in any existing domain and Domain Paths. Fix the domain value in these records & run validator again. Entries causing the error are as follows:sys_id:1d13ae40470002007f47563dbb9a7170, sys_domain:11722b01473231007f47563dbb9a7154 sys_id:22e34b23470002007f47563dbb9a718c, sys_domain:11722b01473231007f47563dbb9a7154 sys_id:4c85fa809f233100fc6cd4b4232e706b, sys_domain:11722b01473231007f47563dbb9a7154 sys_id:94648b23470002007f47563dbb9a711b, sys_domain:11722b01473231007f47563dbb9a7154 sys_id:a0844b23470002007f47563dbb9a71ef, sys_domain:11722b01473231007f47563dbb9a7154 sys_id:a95ff2c09f233100fc6cd4b4232e7096, sys_domain:11722b01473231007f47563dbb9a7154 sys_id:bd25f4719f233100fc6cd4b4232e70da, sys_domain:11722b01473231007f47563dbb9a7154 sys_id:c1263e809f233100fc6cd4b4232e70f6, sys_domain:11722b01473231007f47563dbb9a7154 sys_id:d8a72c45475002007f47563dbb9a71bb, sys_domain:11722b01473231007f47563dbb9a7154 sys_id:e4c3ce39eb10020045e1a5115206fea0, sys_domain:60e014f69f013100fbf01f80a57fcf00	com.glide.domain.validator

ドメインセパレーションエラーの詳細については、「ドメインセパレーションエラーのトラブルシューティング」を参照してください。

デフォルトドメインの重要性

ドメインの整理はドメインセパレーションプロセスの重要な部分です。デフォルトドメインを設定していない場合、新しいタスクとユーザーレコードはグローバルドメインに移動します。グローバルドメイン内のレコードは誰でも表示できるため、本来は表示されないはずのデータが表示される可能性があります。

デフォルトドメインを設定すると、そのレコードはアドミン以外のユーザーには表示されません。

> **注:** デフォルトのアクセス権は、デフォルトドメインまたは親ドメインへの可視化をユーザーに付与することで変更できます。

インスタンスのドメインレコードには常に 1 つのデフォルトドメインを設定する必要があります。ドメインにまだアサインされていないタスクレコードとユーザーレコードが、デフォルトドメインに自動的にアサインされます。

[ドメイン管理] 画面からデフォルトドメインを作成する場合は、他のドメインと区別するために [名前] フィールドに「デフォルト」という名前を追加します。レコードの [デフォルト] チェックボックスをオンにします。

デフォルトドメインで作成したレコードを定期的に管理し、適切なドメインに移動させます。デフォルトドメインでレコードが頻繁に表示される場合は、その理由を調べる必要があります。すべてのレコードが適切なドメイン (グローバルまたはデフォルトドメインではない) に作成されるようにするのが理想的です。

包含クエリーとドメインアクセス

「包含」クエリーは特別な場合にのみ使用します。たとえば、ユーザーまたはグループがアクセス権のないドメインのデータを表示する必要があるが、それらのユーザーをドメインに移動させない場合などです。ドメインの「包含」およびユーザーアクセスまたはグループアクセスは、絶対に必要な場合にのみ例外的に作成できます。

「包含」は多対多のドメイン間関係であり、プロセスのフローには影響しません。多数のドメイン「包含」関係を作成したり、幅広いアクセス権を提供したりすると、OR 条件の多すぎるクエリーが生成されます。OR 条件を指定すると遅くなり、インスタンスのパフォーマンスに影響を与えます。過多の「包含」関係を使用する代わりに、次のようにドメイン階層を設定します。

サンプルクエリ

```
SELECT ... FROM task task0 ignore index(number) WHERE task0.`sys_class_name`
= 'incident' AND (task0.`sys_domain_path` = '/' OR task0.`sys_domain_path` LI
KE '!!$/!!(/%' OR task0.`sys_domain_path` LIKE '!!$/!!$/!!&/%') ORDER BY task
0.`number` DESC limit 0,20
```

ユーザーをドメインに移動させる前に、ユーザーがそのドメインに本当にアクセスする必要があるか確認してください。メリットと制限事項について熟考します。上記のクエリーは、単に 1 つの包含関係のためのものです。別のドメインが含まれるドメインがあり、そのドメインが他の多数のドメインの親である場合は、さらに多くの OR 条件が発生します。ドメインマップを作成するときは、インスタンスのパフォーマンスに影響を与えないように注意してください。

ドメインパスのクエリー方法

ドメインパスを使用して効果的なクエリーを作成できます。

ドメインスプール (sys_domain) またはドメイン番号付けの代わりにドメインパスを使用します。ドメインパスを使用するクエリーは、スプールや番号付けよりもはるかに高速です。

ドメインパスは、ドメインセパレーションが有効になっているインスタンスのデフォルトのクエリー方法です。

インスタンスのクエリー方法を検証するには、アドミンダッシュボードで次のシステムプロパティを探します。

- *If domain path is enabled*：システムプロパティテーブルに、glide.sys.domain.provider=domain_paths および glide.sys.domain.paths.installed=true が表示されます。
- *If domain path is not enabled*：システムプロパティテーブルに、glide.sys.domain.provider != domain_paths,glide.sys.domain.paths.installed=false が表示されます。

遅いクエリーと SQL のデバッグ

SQL と遅いクエリーをデバッグすると、インスタンスでの遅い問題を解決できます。

インスタンスをデバッグするときは、SQL デバッグを有効にして遅いクエリーを探すか、[システム診断] > [統計] > [遅いクエリー] に移動して遅いクエリー [sys_query_pattern] テーブルを確認します。このテーブルは、すべての遅いクエリーをインスタンスに格納します。

テーブルを検索するときに、domain_path を含むクエリーを探して、インスタンス内のドメインパスが原因であるかどうかを判断します。

遅いクエリーが見つかった場合は、遅い理由を分析してください。

クエリーが遅い一般的な理由

- クエリーに含まれる OR 条件が多すぎます (詳細については、「包含クエリーとドメインアクセス」を参照してください)。ドメイン階層で、包含または可視化の必要がない階層レベルにユーザーまたはドメインを配置します。
- クエリー方法は、ドメインパスクエリー方法ではありません (詳細については、「ドメインパスのクエリー方法」を参照)。ドメインパスのクエリー方法を使用していない場合は、カスタマーサービス & サポートに連絡してください。
- クエリーでは、データベースの内容をすばやく確認できるようにデータベースをインデックス化する必要があります。遅いクエリーを特定できる場合は、「explain plan」を実行して、インデックス作成のオプションがあるかどうかを確認します。「explain plan」は、クエリーとその処理内容を表示する SQL の機能です。

クエリ前ビジネスルール

クエリ前ビジネスルールを使用して、インスタンスでデータ分離をサポートすることができます。ドメインセパレーションをサポートする ServiceNow アプリケーションは、データとデータルーティングの分離のみをサポートするか、高度なビジネスロジック分離機能があるか、アプリケーションのテナント (顧客) レベルの管理をサポートする可能性があります。

クエリ前ビジネスルールは、ドメインセパレーションされた環境内でデータ分離をサポートするために使用する補足コードです。

警告: ドメインセパレーションプラグインの代わりにクエリ前ビジネスルールを使用しないでください。このビジネスルールは、プラグインほど安全にデータ漏洩を防止しません。

データ分離のためのクエリ前ビジネスルールの使用

次の状況では、クエリ前ビジネスルールをデータ分離と併用できます。

- ドメインセパレーションが ServiceNow アプリケーションでサポートされておらず、サービスプロバイダー組織外の 1 人以上の外部顧客にテーブルまたは行へのアクセスを許可または制限する必要がある場合。

  注: 開発を開始する前に、その製品のアプリケーションロードマップについて ServiceNow カスタマーサポートにお問い合わせください。ドメインサポートの改善が今後のリリースで計画されている可能性があります。

- テーブルがドメインセパレーションされていても、システム内の一連のドメインにのみ適用される特定の条件に基づいて、その行へのアクセスを許可または制限する必要がある場合。

  注: たとえば、X ドメインの顧客にはそのドメインをサポートする複数のベンダーがあり、それらのベンダーには、割り当てられたレコードのみを表示するためのアクセス権が付与されます。

クエリ前ビジネスルールを作成する前の考慮事項

ユーザー情報、グループメンバーシップ、会社、ロール、またはレコード固有のフィールド条件の組み合わせに基づいて、クエリ前ビジネスルールをスクリプト化して、親テーブルと子テーブルにアクセスできないようにすることができます。クエリ前ビジネスルールは個別のドメインに配置され、ドメイン階層の特定の分岐にグローバルに適用されるように作成されます。

- 可能な場合は、適用されるユーザーに対してのみルールが実行されるように、ドメイン階層のできる限り低いところでクエリ前ビジネスルールを作成します。
- ビジネスルールが実行されないシナリオや、ユーザーがトリガーしたインタラクションが実行するビジネスルールをトリガーしないシナリオが存在することを理解してください。たとえば、[ビジ

ネスルールを実行] がオフになっている変換マップがある場合、またはワークフローが無効になっているスクリプトがある場合、ビジネスルールは実行されません。

- 必ず条件フィールドを入力して、ルールを実行するタイミングを指定します。たとえば、ビジネスルールをドメイン内の特定のベンダーにのみ適用するかどうかを指定できます。

> ⚠ **警告:** ビジネスルール (特にクエリービジネスルール) を設計およびコーディングするときは、OR 句とインデックスのないフィールドの検索を制限します。インデックス化されていないフィールドで OR 句や検索が多すぎると、クエリが遅くなったり、インスタンスのパフォーマンスに影響したりする可能性があります。

- クエリ前ビジネスルールは必要な場合にのみ使用します。クエリ前ルールが多すぎると、インスタンスのパフォーマンスに影響を与える可能性があります。

クエリ前ビジネスルールは、アクセス制御リスト (ACL) の前に実行され、全体的なパフォーマンスを向上させます。これは、システム内の複数のドメインにアクセスできるサービスプロバイダー (SP) 環境のユーザーに返される結果を制限する場合に特に当てはまります。

> ℹ **注:** データのフィルタリングは (ACL とは異なり)、データのやり取りをするときに「データセキュリティは…を制限します」というメッセージが表示されないユーザーにとって透過的になります。

クエリ前ビジネスルールと **ACL** を使用しない場合

クエリ前ビジネスルールと ACL を使用して顧客データを分離する場合は注意してください。ビジネスルールと ACL の両方を使用して、カスタマイズを作成し、それを維持する必要があります。カスタマイズはパフォーマンスの問題を引き起こす可能性があります。開発チームは、システムが壊れないようにプロセスを作成する必要があります。

ドメインセパレーションは、広くサポートされているフレームワークである現在のドメインパスクエリー方法 (v3) を使用して、スケーラビリティとガバナンスの両方を提供します。ServiceNow プラットフォームチームとアプリチームは、フレームワークを維持する責任を負い、顧客の負担を軽減します。

多くのインスタンスで多数の顧客を持つ企業では、クエリ前ルールと ACL を過度に使用すると、データベースクエリのパフォーマンスが低下する可能性があります。

ドメインセパレーションを有効にする方法

ServiceNow プラグインを使用してドメインセパレーションを有効にできます。製品マネージャーが機能を管理し、開発チームが製品マネージャーをサポートします。ServiceNow リリースには、ドメインセパレーション機能の拡張と修正が含まれています。インスタンスオーナーは、ドメインセパレーションのサポートについて、https://support.servicenow.com の サービスポータル など、カスタマーサービス & サポートリソースを参照できます。

スクリプトでのドメインパスの回避

ドメインパスによってスクリプトの値が変更されたり壊れたりする可能性があるため、スクリプトでは使用しないでください。

ドメイン階層を変更すると、ドメインパスが再計算されてその値が変更されるため、スクリプトがドメインパスに依存しないようにしてください。このようなことが発生すると、スクリプトが役に立たないか、エラーがスローされたり壊れたりする可能性があります。最適な方法は、ドメインパスに基づいてスクリプトを作成しないことです。

ドメインパスに依存するのではなく、スクリプトで **sys_domain** フィールドを使用します。ドメイン階層を変更すると、ドメインパスが再計算されてその値が変更されるため、スクリプトが役に立た

なくなったり、エラーがスローされたり、壊れたりする可能性があります。独自のスクリプトを作成する前に、**[sys_domain]** フィールドを使用するベースシステムのビジネスルールを検索してください。

ServiceNow プラットフォームは、各インスタンスのドメイン階層の違いによる問題を回避するために、更新セットの sys_domain_path 値をキャプチャしません。したがって、更新セットをインポートした後にドメイン階層を検証して、レコードのドメインパス値が正しいことを確認する必要があります。

ドメインパスの詳細については、「ドメインセパレーションを要求する」および「ドメインセパレーションセンター」を参照してください。

ドメインアサイン

ドメインのアサイン方法は sys_domain フィールドの値に影響します。アサインには、各ドメインでのアプリケーションの機能に影響するデザインとビジネスプロパティが含まれています。

sys_domain フィールドの値

sys_domain フィールドの値には、次のいずれかでレコードにアサインされたドメインが含まれます。

- ユーザーが所属する会社
- レコード作成時に使用されるビジネスルール
- レコード作成時に使用されるモジュール
- レコード作成時に使用されるフォームテンプレート
- 親レコードのドメイン
- ユーザーレコードにアサインされたドメイン
- 作成したユーザーのドメイン

ドメインアサインの戦略と設計が適切に文書化され、テストされていることを確認して、これらの戦略と設計が正しいドメインに挿入されるようにします。これにより、各ドメインのプロパティを複製または変更する必要があることを確認できます。

ドメインセパレーションと カスタマーサービス管理 **(CSM)** プラグイン

最良の結果を得るには、CSM プラグインのプロパティがどのように機能するかに注意してください。プラグインを有効にすると、ドメイン内のレコードのステータスを確認できます。

インスタンスオーナーは、*csm_auto_account_domain_generation* プロパティを有効にするためにカスタマーサービス & サポートに連絡する必要があります。

注: このベースシステムプロパティは、システムプロパティテーブルにあり、CSM プラグインを有効にした後に使用できます。

プロパティの機能

カスタマーサービスアプリケーションで新しいアカウントが作成されると、ドメインが作成され、TOP ドメインの下に配置されます。アカウントフォームの親フィールドに値が入力され、新しいレコードが挿入されると、そのアカウントが親のサブドメインとして作成されます。

このプロパティが true でなく、ドメインが有効になっている場合の動作

ドメインセパレーションされた環境の新しいアカウントレコードは、デフォルトのドメインに自動的に配置されます。

ヘッダーバーで、プラグインが有効になっているレコードのステータスを確認できます。

ドメインセパレーションのヘルプ

ドメインセパレーションに関する追加のヘルプ

ドメインセパレーションの概要ビデオ

探索、学習、開発

推奨プラクティス

ドメイン構造を適切に作成および開発するためのヒント

サービスプロバイダーの概念

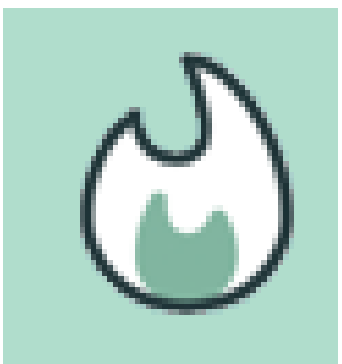
一般的なユースケースの解決に役立つ ServiceNow プラットフォームを扱う概念

アプリケーションによるサポートレベル

アプリケーションでドメインセパレーションはサポートされていますか？サポートレベルとユースケースを参照してください。

クラス

- 開発者向け: ServiceNow 開発者サイトのドメインセパレーション
- サービスプロバイダーの場合: サービスプロバイダーのドメインセパレーション (ServiceNow University へのログインが必要)



セーフワークプレイススイートとドメインセパレーション

ServiceNow セーフワークプレイスアプリケーションは、緊急時や COVID-19 のようなパンデミックの後に職場を再開し従業員の衛生安全をサポートするために役立ちます。このスイートには、組織の動員、復旧、および再ビルドに役立つ多くのアプリケーションが搭載されています。

自動翻訳

セットアップと管理

- アップグレード
- ドメインセパレーションを要求する
- ドメインを作成する
- 内部や外部の顧客に委任できる構成

トラブルシューティング

- Known Error Portal で既知のエラー記事を検索
- 連絡先 カスタマーサービス & サポート
- コミュニティで質問や回答をする

ドメインセパレーションのセットアップと管理

ドメインセパレーションを設定するには、プラグインのアクティベーションを要求し、オプションを設定し、ユーザーとレコードをドメインに割り当てる必要があります。

ドメインセパレーションを設定するには、次の操作を行います。

1. ドメインセパレーションを要求する
2. ドメインを作成する
3. テーブルへのドメインフィールドの追加

ドメインでは、次の基本的な管理タスクを実行することもできます。

- ドメインを有効または無効にする
- ドメイン関係を表示する
- ドメインスコープを展開する
- ドメイン固有の選択リストを作成する

ドメインセパレーションを設定して基本的な管理を行った後に実行するタスクのリストについては、「詳細なドメインセパレーション管理」を参照してください。

ドメインセパレーションを要求する

すべてのドメインサポート機能は、**Domain Support - Domain Extensions Installer** と呼ばれるプラグインでアクティブ化されます。アドミニストレーターは、このプラグインのアクティブ化を要求できます。

始める前に

サブスクリプションを購入するには、ServiceNow アカウントマネージャーにお問い合わせください。アカウントマネージャーは通常数日以内に、組織の本番および準本番インスタンスで `com.glide.domain.msp_extensions.installer` プラグインがアクティブ化されるように手配することができます。

アカウントマネージャーがいない場合、購入後にアクティブ化を延期することを決定するか、無料で準本番インスタンスで製品を評価する場合には次の手順を実行します。

必要なロール：admin

このタスクについて

Domain Support - Domain Extensions Installer プラグインがすでにアクティブな場合、Domain Support - Domain Extensions Installer プラグインのコンテンツは、既存の実装との潜在的な競合を回避するためにインストールされません。

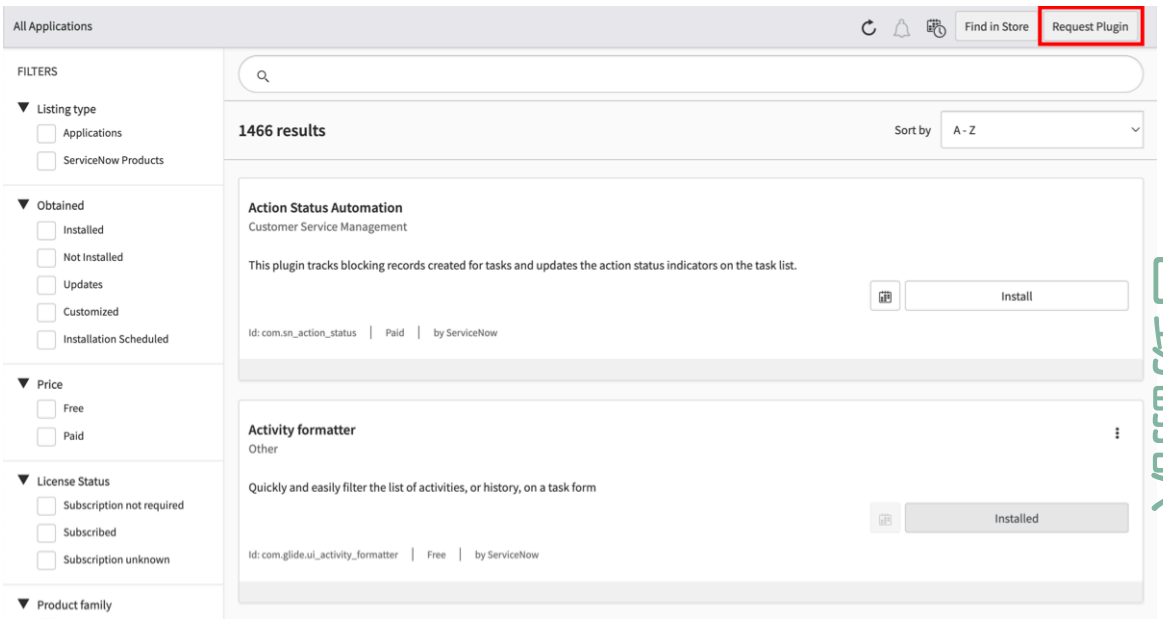
自動翻訳

ドメインセパレーションは Company Separation に替わるものです。Helsinki リリース以降、Company Separation プラグインをアクティブ化できなくなりました。ただし、ドメインセパレーションをアクティブ化したときに Company Separation が既にアクティブになっている場合は、両方のプラグインが同時にアクティブになります。`glide.db.separation.field` プロパティを使用して、Company Separation のアクティベーションステータスをコントロールできます。

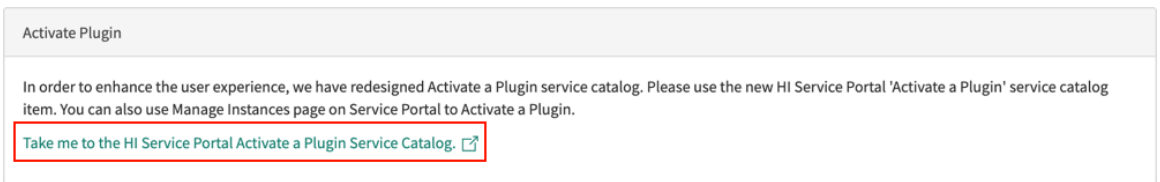
注: Helsinki 以降は、ドメインパスはすべての顧客に使用されます。ドメイン番号付けは使用されなくなりました。カスタマーサービス & サポートがアップグレードのサポートを行います。

手順

1. 移動先 すべて > システムアプリケーション > 利用可能なすべてのアプリケーション > すべて.
2. [すべてのアプリケーション] ページで [プラグインの要求] をクリックして、Now Support で [プラグインをアクティブ化] フォームを開きます。



3. Now Support で、Now Support サービスポータル サービスカタログ にアクセスするリンクを選択します。



4. インスタンスを選択します。
5. [アクション] > [プラグインのアクティブ化] を選択します。
6. [プラグインのアクティブ化] フォームで、次の情報を入力します。

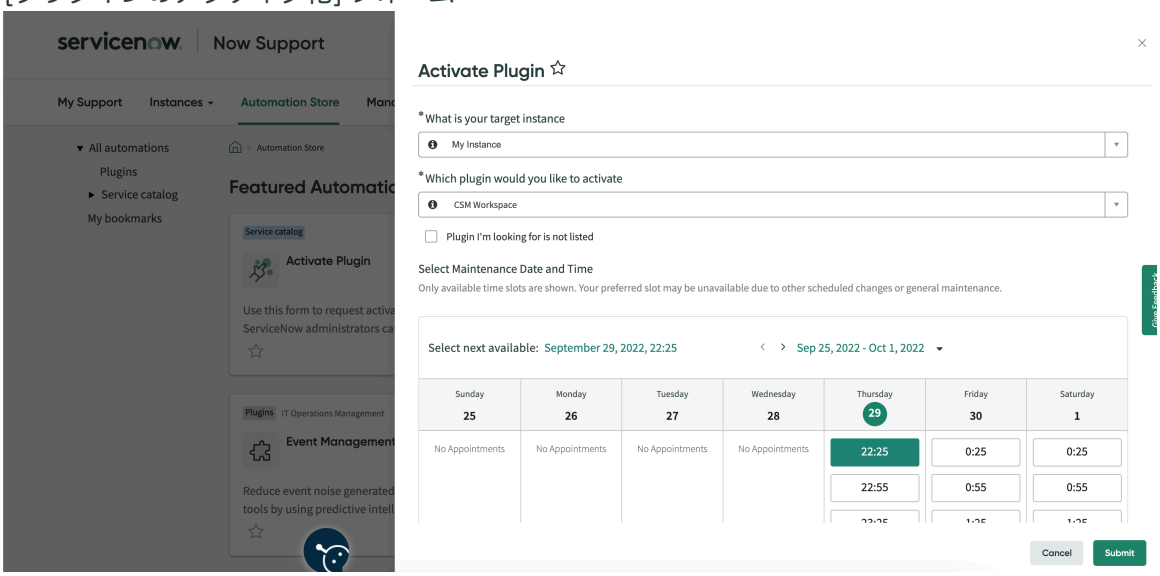
[プラグインのアクティブ化] フォーム

フィールド	説明
ターゲットインスタンスは何ですか	プラグインをアクティブ化するインスタンス。
どのプラグインをアクティブ化しますか	アクティブ化するプラグインの名前です。 **注:** 必要なプラグインが表示されない場合、または OEM またはオンプレミスのインスタンスでプラグインをアクティブ化している場合は、[探しているプラグインが表示されていません **(Plugin I'm looking for is not listed)]** チェックボックスをオンにして、プラグインの名前を入力します。
メンテナンスの日時を選択 (Select Maintenance Date and Time)	プラグインをアクティブ化する日時。 **注:** プラグインは、米国太平洋標準時で、毎営業日の朝と夕方の 2 回のバッチでアクティブ化されます。特定の時刻にプラグインをアクティブ化する必要がある場合は、[理由/コメント **(Reason/Comments)]** フィールドに要求を入力します。

Example

たとえば、 [自分のインスタンス (My Instance)] という名前のインスタンスで CSM Workspace プラグインをアクティブ化するには、次のフォームを参照してください。

[プラグインのアクティブ化] フォーム



7. **[Submit** (送信)] を選択します。
プラグインの要求の詳細については、次を参照してください。 のサービスカタログ [KB0751715] 記事からのプラグインの要求 Now Support ナレッジベース.

結果
Domain Extension Installer プラグインをアクティブにすると、次の機能が有効になります。

- ドメインセパレーションはドメイン [sys_domain] テーブルに基づいています。
- 委任管理では、各ドメインに個別のポリシーを設定できます。
- すべてのレコードはグローバルドメインの一部です。
- 現在のユーザーのドメインによって、使用するドメインが決まります。表示 または別のドメインのレコードで操作します。

関連トピック

ドメインセパレーションプラグイン

ドメインセパレーションプラグイン

Domain Support - Domain Extensions Installer プラグインは、複数のドメインセパレーション機能とプロパティを一度にアクティブにします。このプラグインは通常、ドメインセパレーションプラグインと呼ばれます。

ドメインセパレーションプラグインをアクティブにするための推奨プラクティス

ドメインセパレーション開発の一環として、アドミニストレーターはこのプラグインのアクティベーションを要求する必要があります。最良の結果を得るには、開発プロセスの開始時に、他のプラグインをアクティブにする前に、ドメインセパレーションプラグインをアクティブにします。

重要: Domain Extensions Installer プラグイン (com.glide.domain.msp_extensions.installer) のアクティブ化を要求してからドメインセパレーションプラグイン (com.snc.pa.domain_support) をアクティブ化するように要求します。

ServiceNow の実装の最後にドメインセパレーションを有効にした場合、またはインスタンスが稼働すると、アプリケーションのパフォーマンスとプロセスの両方がリスクにさらされます。確立されたインスタンスでは、開発の構造によっては、プラットフォームとその使いやすさに対するリスクが高くなる可能性があります。ドメインセパレーションプロセスの詳細については、「ドメインセパレーションの概要」を参照してください。

たとえば、ドメインセパレーションプラグインがアクティブになっている場合、ドメイン (sys_domain) 列がタスクテーブルに追加され、既存のすべてのレコードが自動的にグローバルに配置されます。スクリプトを使用してすべてのレコードを正しいドメインに割り当てるには、確立された親子階層が必要です。これらのタイプのスクリプト化されたアクションでは、大量のデータを移動するときにデータの破損や損失が発生し、本番環境でダウンタイムが発生する可能性があります。ビジネスルール、クライアントスクリプト、フォームビュー、ワークフローなどのプラットフォームコードの多くも、グローバルに配置されます。

顧客がコードを作成したり ServiceNow コードを変更したりすると、プラットフォームのパフォーマンスと使いやすさが低下します。インスタンスオーナーは、このタイプのアプローチでは、実装が大幅に遅れたり、長いダウンタイムが発生したりする可能性があります。

ドメインセパレーションプラグインの機能

プラグインを有効にすると、次の機能が有効になります。

- ドメインセパレーションはドメイン [sys_domain] テーブルに基づいています。
- 委任管理では、各ドメインに個別のポリシーを設定できます。
- すべてのレコードはグローバルドメインの一部です。
- 現在のユーザーのドメインによって、使用するドメインが決まります。表示 または別のドメインのレコードで操作します。

関連トピック

サービスプロバイダー向けのドメインセパレーションの推奨プラクティス

ドメインシステムのプロパティとユーザー設定

アドミニストレーターは、ドメインスコープをコントロールするプロパティとユーザー設定にアクセスできます。

プロパティ

ドメインセパレーションを新しくアクティブにすると、関連するすべてのデータまたはプロセスのレコードのドメインスコープが自動的に制限されます。ユーザーがフォームでレコードを表示すると、レコードの関連データ (参照ピッカーや関連リストデータなど) と適用されるプロセス (ビジネスルールやクライアントスクリプトなど) がレコードのドメインスコープに制限されます。複数のタブにレコードがある場合、各タブには、そのタブ内で開かれたレコードに基づく独自のドメインスコープがあります。次のプロパティは、ドメインスコープをレコードのドメインとユーザーの現在のセッションドメインのいずれかに制限します。

ドメインシステムプロパティ

プロパティ	詳細
glide.sys.domain.use_record_domain_for_processes	ドメインスコープをすべてのプロセスのレコードのドメインに制限します。このプロパティはビジネスルールには適用されません。ビジネスルールは常にドメインレコードから処理されます。 • タイプ：true | false • デフォルト値：true • 場所：システムプロパティ [sys_properties] テーブル
glide.sys.domain.use_record_domain_for_data	ドメインスコープをすべてのデータのレコードのドメインに制限します。 • タイプ：true | false • デフォルト値：Fuji 以降の新しいドメインアクティベーションでは true (Fuji より古いインスタンスからのアップグレードには、このプロパティはテーブルにありません) • 場所：システムプロパティ [sys_properties] テーブル

glide.sys.domain.use_record_domain_for_processes または *glide.sys.domain.use_record_domain_for_data* プロパティのいずれかが **true** に設定されている場合、次のプロパティは設定に関係なく使用されません。

- *glide.sys.domain.use_record_domain*
- *glide.sys.domain.use_record_domain_for_client_scripts*
- *glide.sys.domain.domain_change_notify*
- *glide.sys.domain.no_change_roles*

プロパティの完全なリストについては、「利用可能なシステムプロパティ」を参照してください。

注:

Jakarta リリース以降のドメインセパレーションの新しいアクティベーションでは、セッションドメインがドメインテーブルで実行されるビジネスルールを決定します。以前のバージョンでは、ドメインテーブルで実行されるビジネスルールは、新しく作成されたドメインの階層に基づいて設定されていました。この動作は *glide.sys.domain.skip_domain_insert_businessrules* プロパティによって変更されます。このプロパティを true に設定すると、ドメイン挿入のパフォーマンスが大幅に向上します。

ドメインテーブルで実行されるビジネスルールのドメインスコーププロパティ

プロパティ	詳細
glide.sys.domain.skip_domain_insert_businessrules	ドメインテーブルで実行されるビジネスルールのドメインスコープを指定します。ドメインセパレーションの新しいアクティベーションでは、プロパティのデフォルトは true であり、ビジネスルールはセッションドメインによって決定されます。既存の実装では、プロパティのデフォルトは false であり、ビジネスルールは新しく作成されたドメインの階層によって決定されます。 • タイプ：true | false • デフォルト値：Jakarta 以降、新しいドメインのアクティベーションでは true です。既存の実装では False です。
glide.sys.domain.skip_non_global_businessrule_if_nodomain	queryNoDomain() を使用する場合、またはテーブルがドメインセパレーションされていない場合に、グローバルドメインからの bus.rules のみが実行されるようにするため、他のビジネスルールはスキップできます

自動翻訳

ドメインテーブルで実行されるビジネスルールのドメインスコーププロパティ (続く)

プロパティ	詳細
	• タイプ：true | false • プロパティを false に設定すると、以前の動作が復元され、ServiceNow の推奨プラクティスに沿ったものにはなりません。 • 推奨：テーブルをドメインセパレーションします。セッションドメインではなく、常にレコードのドメインを使用するようにしてください。

ユーザー設定

さらに、ユーザーアドミニストレーターは、次のユーザー設定をグローバルまたはユーザーごとに設定できます。

ドメインスコープのユーザー設定

設定	カテゴリ	更新者	詳細
glide.domain.session_scope	ドメイン	アドミンのみ	true の場合、デフォルトのスコープをレコードのドメインではなくユーザーのセッションドメインに設定します。false の場合、デフォルトのスコープはレコードのドメインです。domain_expand_scope ユーザーロールのユーザーは、必要に応じてドメインスコープを変更できます。 • タイプ：true | false • デフォルト値：false
glide.domain.session_scope_notification	ドメイン	アドミンのみ	true の場合、レコード値に展開されたドメインスコープが含まれていることを示す視覚的なキューが表示されます。false の場合、通知は非表示になります。 • タイプ：true | false • デフォルト値：true

関連トピック

ドメインセパレーションアプリケーションプロパティ

ドメインを作成する

[domain] テーブルにレコードを作成することで、ドメインを作成できます。

始める前に
必要なロール：admin

このタスクについて
新しいドメインを作成するときは、次の点に注意してください。

- 1 つのドメインのみをデフォルトドメインにすることができます。
- 1 つのドメインのみをプライマリドメインにすることができます。

手順

1. 移動先 すべて > ドメイン管理 > ドメイン.
2. [**New**] をクリックします。
3. 必要なフィールドに入力します (表を参照)。
4. [送信] をクリックします。

ドメインフォームのフィールド

フィールド	説明
名前	ドメインの一意の名前を入力します。
タイプ	ドメインを説明するドメインタイプを選択します。デフォルトでは、ドメインタイプはベンダー、顧客、および *MSP* です。独自の選択肢を追加することもできます。
プライマリ	このドメインを階層のトップレベルドメインにする場合は、このチェックボックスをオンにします。トップレベルドメインには子ドメインのみがあり、親ドメインはありません。
デフォルト	このドメインを階層のデフォルトドメインにする場合に選択します。
親	このドメインを含む階層の上位のドメインの名前を選択します。このフィールドには、ドメインマップに表示されるドメインの値が必要です。
有効	このチェックボックスをオンにしてドメインを使用できるようにします。このドメインをドメインマップに表示するには、このオプションを選択する必要があります。
説明	ドメインの説明を入力します。

各ドメインレコードには、次の複数の関連レコードを含めることもできます。

- 会社
- 包含ドメイン
- 次のものに含まれる：

次のタスク
ドメイン階層を変更するには、包含ドメイン関連リストに移動し、包含関係の子 (含まれている) ドメインであるドメインレコードを選択します。

デフォルトドメインの設定

ドメインを、ドメインにまだアサインされていないタスクレコードおよびユーザーレコードが自動的にアサインされるデフォルトドメインにします。

始める前に
必要なロール：admin

このタスクについて

注: デフォルトドメインを設定していない場合、新しいタスクとユーザーレコードはグローバルドメインに配置されます。

手順

1. 移動先 すべて > ドメイン管理 > ドメイン.
2. たとえば Main など、デフォルトドメインにするドメインを開きます。
3. [デフォルト] フィールドを追加するため、フォームレイアウトを設定します。
4. [デフォルト] チェックボックスをオンにします。
5. [更新] をクリックします。

特定のレコードのドメインを手動で管理する

デフォルトでは、ユーザーの会社レコードに基づいてドメインが自動的に割り当てられます。ただし、場合によっては、ドメインアドミニストレーターが特定のレコードが属するドメインを手動で管理する必要があります。

始める前に
必要なロール：admin

このタスクについて
[管理対象ドメイン] フィールドで、ドメインアドミニストレーターは、会社レコードから自動的に割り当てられたドメインを使用するのではなく、ユーザー、グループ、部門、場所、または CI レコードのドメインを手動で選択できます。[管理対象ドメイン] フィールドは、次のレコードタイプで使用できます。

- ユーザーレコード
- グループレコード
- 部門レコード
- 場所レコード
- CI レコード

手順

1. 手動で管理するレコードに移動します。
2. [管理対象ドメイン] チェックボックスをオンにします。
3. [ドメイン] フィールドから、レコードのドメインを選択します。
4. [更新] をクリックします。

[管理対象ドメイン] チェックボックスをオフにすると、[ドメイン] フィールドが非表示になり、レコードではレコードの会社のドメイン値が使用されます。

関連トピック

サービスプロバイダー向けのドメインセパレーションの推奨プラクティス

自動翻訳

ドメインセパレーションテーブル

ドメインセパレーションテーブル機能を使用すると、インスタンス内のどのテーブルがドメインセパレーションされているかを一目で確認できます。

概要

ドメインセパレーションテーブル機能を使用して、ドメインセパレーションされているテーブルを確認します。[すべて] フィルターに「ドメイン」と入力し、左側のナビゲーションペインでドメインセパレーションテーブルにアクセスします。

このビューをフィルタリングして列名を表示または削除し、テーブルに含まれる特定のプロパティまたは属性を検索できます。リストに表示されるテーブルタイプは 2 つあります。

- 明示的な *sys_domain* 列が存在するテーブルで、列名にリストの *sys_domain* の値が表示されます。
- *domain_master* 属性を使用して参照先レコードからドメインセパレーションを取得しているテーブル。このテーブルには、属性列に *domain_master=ref-field-value* 値が含まれています。

[列名] メニューの [この値で絞り込み] または [除外] の選択肢を使用すると、次の 2 つのタイプのテーブルを表示できます。

ドメイン上書きビューアー

ドメイン上書きビューアーを使用すると、インスタンス全体ですべてのプロセス上書きを一度に表示および管理できます。

概要

スクリプト内でオーバーライドの詳細検索を行う代わりに、ドメイン上書きビューアーを使用してオーバーライドをすばやく検索することができます。[すべて] フィルターに「ドメイン」と入力し、左側のナビゲーションペインでドメイン上書きビューアーにアクセスします。

ドメイン上書きビューアーのテーブルセレクターには、レコードの上書きを含むテーブルのみのリストと、そのテーブル内の上書きの数が表示されます。テーブルを選択すると、上書きを含むすべての親レコードリストが返されます。すべての親レコード、親レコードのドメイン、および各レコードの上書き数をすばやく表示できます。

[ビューの上書き **(View Overrides)]** を選択すると、親レコードを含むすべての上書きが標準リストビューに表示される新しいタブがロードされます。ドロップダウンリストからテーブルを選択すると、すべてのレコードと上書きが表示されます。レコード固有の上書きをすべて表示するには、レコードの [ビューの上書き **(View Overrides)]** を選択します。

注: 上書きされたテーブルのみがリストされます。

詳細については、「ドメインセパレーションプロパティの上書きを作成する」を参照してください。

ドメインを有効または無効にする

ドメインをアクティブまたは非アクティブにすると、アクティベーションステータスがドメイン内の会社にカスケードされます。

始める前に
必要なロール：admin

このタスクについて
会社レコードをアクティブにすると、ドメインセパレーションによって会社の関連ドメインが自動的にアクティブになります。たとえば、ACME 社をアクティブにすると、TOP/ACME ドメインもアクティブになります。

手順

1. ドメインレコードに移動します。
2. [アクティブ] チェックボックスをオフまたはオンにします。
3. [更新] をクリックします。

⚠ 警告: ドメインを削除しないでください。不要になったドメインは削除するのではなく非アクティブ化します。

テーブルへのドメインフィールドの追加

アドミニストレーターは、sys_domain フィールドを追加することで、カスタムテーブルをドメインセパレーションできます。

始める前に
必要なロール：admin

このタスクについて
注:

ベースシステムテーブルにドメインを追加しないでください。

手順

1. テーブルのリストビューに移動します。
   たとえば、 <table name>ナビゲーションフィルターに「.list」と入力します。
2. リストヘッダーを右クリックし、 構成 > リストレイアウト.
3. [新規フィールドの作成] セクションで、[名前] に sys_domain を、[タイプ] にドメイン ID を入力します。
4. [追加] をクリックします。
5. [保存] をクリックします。

注:

フィールドを作成する他の方法では、列名に **u_** プリフィックスが追加されます。ただし、ドメインフィールドを使用すると、**u_** プリフィックスなしでフィールドが自動的に作成されます。次の機能をショートカットとして使用できます。**[sys_domain]**フィールドを作成するときは常に、「**sys_domain**」という名前を付け、フィールドタイプはそのままにします。フィールドタイプは自動的にドメイン **ID** に、フィールドラベルはドメインに設定されるため、数回クリックする手間が省けます。

ベースシステムテーブルにドメインを追加するには、非常に徹底的なテスト、更新、および新しいロジックの追加が必要です。また、多くの場合、顧客はソースコードにアクセスできません。

ドメイン関係を表示する

ドメインマップは、インスタンス上のアクティブなドメインとそれらの相互関係の読み取り専用表現をドメインアドミニストレーターに提供します。

自動翻訳

始める前に
必要なロール：admin

このタスクについて
すべてのドメインマップには、プライマリドメインとして設定された 1 つのドメインが必要です。さらに、ドメインマップ内の各ドメインは次の基準を満たす必要があります。

- [親] フィールドを入力する必要があります (プライマリドメインはこれに対する唯一の例外です)。
- [アクティブ] チェックボックスをオンにする必要があります。

ドメインマップは、マッピング基準を満たさないドメインのドメイン関係を描画しません。

手順

1. 移動先 すべて > ドメイン管理 > ドメインマップ.
2. ドメインヘッダーのプラス (+) またはマイナス (-) アイコンをクリックして、サブドメインを表示または非表示にします。

プライマリドメインの選択

プライマリドメインは、ドメインマップのトップレベルドメインを示します。

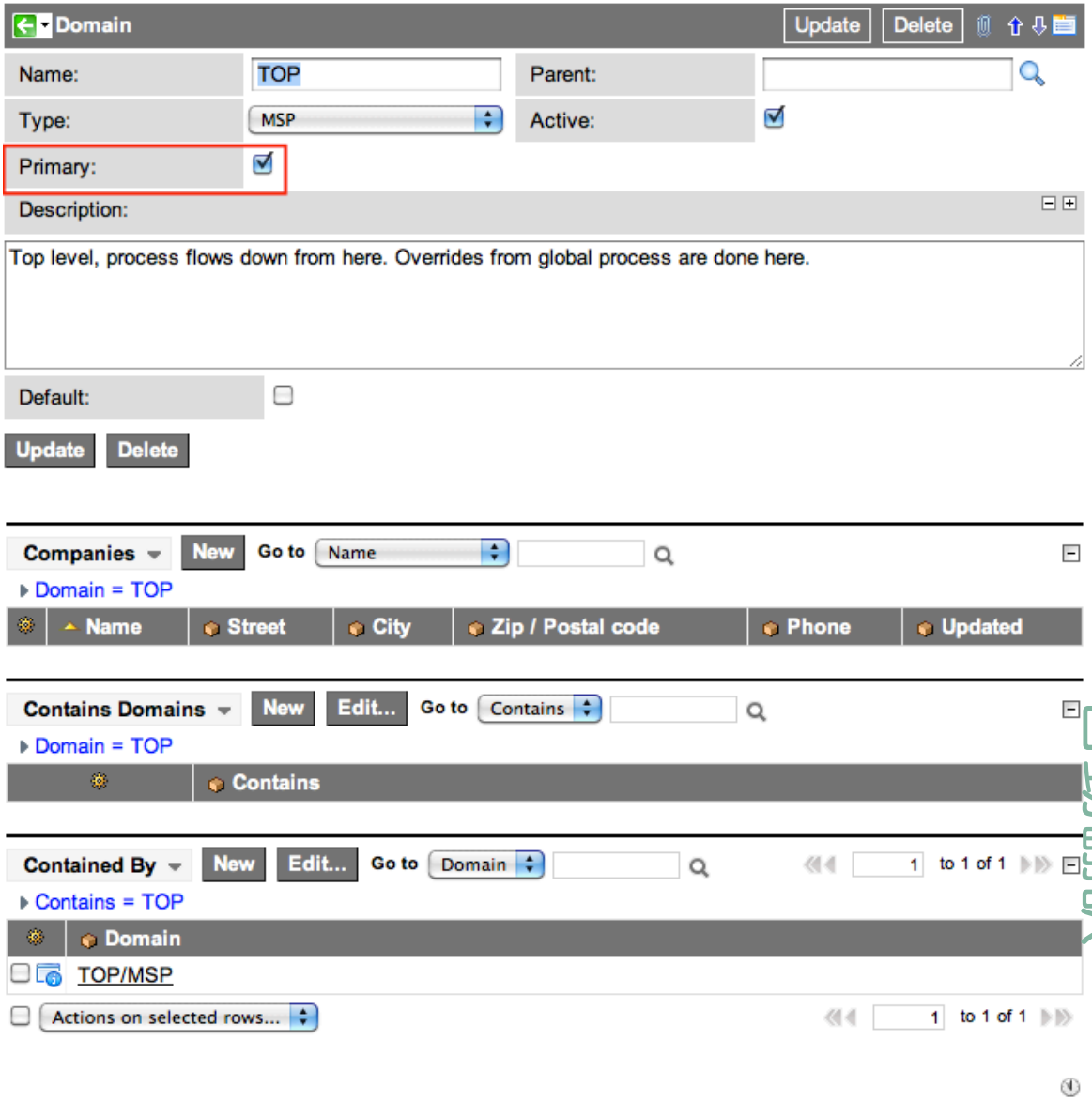
始める前に
必要なロール：admin

このタスクについて
プライマリドメインは親ドメインを持つことができず、少なくとも 1 つの子ドメインを持つ必要があります。一度に存在できるプライマリドメインは 1 つだけです。プライマリドメインとして別のドメインを選択すると、前のプライマリドメインが上書きされます。

手順

1. 移動先 すべて > ドメイン管理 > ドメイン.
2. たとえば TOP など、プライマリドメインにするドメインを選択します。
3. [プライマリ] チェックボックスをオンにします。

4. [更新] をクリックします。



ドメイン間の包含関係の作成

ドメイン間の「包含」関係を作成し、ドメイン階層を変更します。

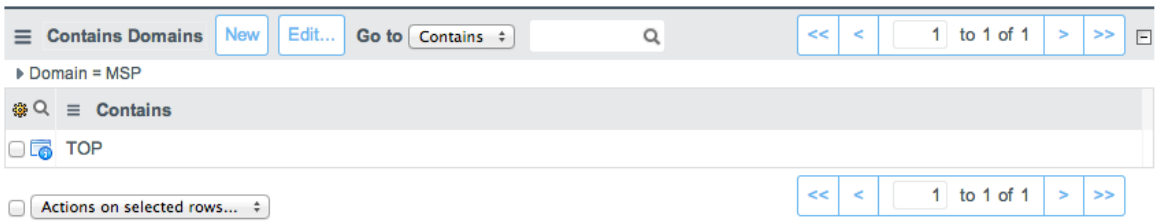
始める前に
必要なロール：admin

このタスクについて
包含関係のドメインは、包含ドメインの設定を継承します。包含関係ドメインにより、ユーザーは包含ドメイン内のデータとその子を表示できます。プロセスは包含関係の影響を受けません。

手順

1. ドメインテーブルに移動します。
2. 新しい包含関係の親 (コンテナ) ドメインであるドメインレコードを選択します。
3. 必要に応じて、ドメインスコープを切り替えて、セッションスコープとレコードスコープを切り替えます。

4. 包含ドメイン関連リストで [編集] をクリックします。
5. 包含関係の子 (含まれている) ドメインであるドメインレコードを選択します。
   ドメインピッカーが [グローバル] に設定されている場合、デフォルトでは子ドメインのみが表示されます。ドメインスコープを切り替えて、すべてのドメインを表示します。
6. [保存] をクリックしてから、[更新] をクリックします。



関連トピック

サービスプロバイダー向けのドメインセパレーションの推奨プラクティス

ドメインスコープを展開する

デフォルトでは、グローバルドメイン内のユーザーが **sys_overrides** 列を含むテーブルを表示すると、グローバルドメインのレコードのみが表示されます。グローバルドメイン内のアドミニストレーターがプロセステーブルを表示すると、そのアドミニストレーターにはそのプロセステーブル内のレコードのみが表示されます。

始める前に
必要なロール：admin

手順

1. *glide.sys.restrict_global_domain_processes* プロパティを **true** に変更します。
2. すべてのドメインのレコードを表示するには、[関連リンク] の [ドメインスコープを展開] をクリックします。
3. グローバルドメインのレコードのみの表示に戻るには、[ドメインスコープを折りたたむ] をクリックします。

ヴィジビリティドメインリストにドメインを追加する

ヴィジビリティドメインを追加すると、ユーザーまたはグループの通常のドメインメンバーシップに関係なく、ユーザーまたはグループは別のドメインのレコードを表示でき、場合によっては編集できます。

始める前に
必要なロール：admin

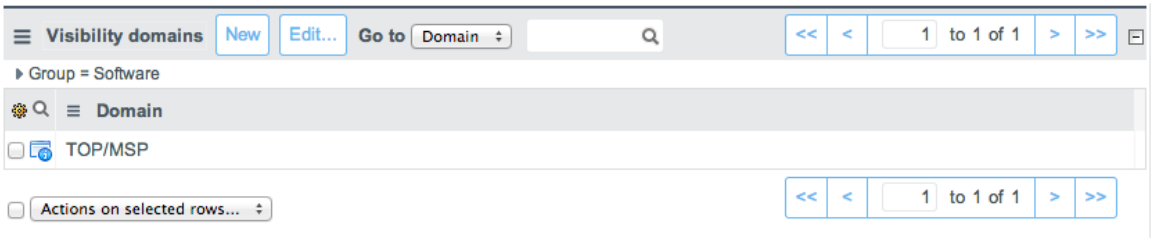
このタスクについて
ヴィジビリティドメインを個々のユーザーに付与するよりも、グループのすべてのメンバーに割り当てることをお勧めします。

> **注:** ヴィジビリティドメインを追加しても、テーブルまたはレコードのアクセス制御ルール要件は変更されません。

手順

1. グループテーブルに移動します。
2. ヴィジビリティドメインを提供するグループを選択します。

3. ヴィジビリティドメイン関連リストをフォームに追加します。
4. ヴィジビリティドメイン関連リストで [編集] をクリックします。
5. グループまたはドメインに表示するドメインレコードを選択します。
6. [保存] をクリックしてから、[更新] をクリックします。



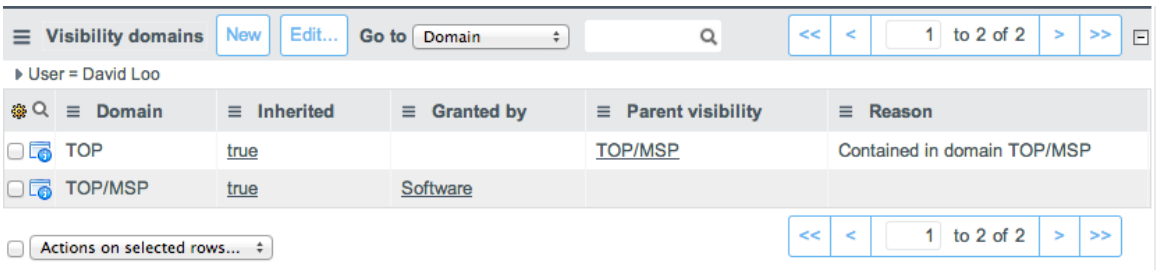
個々のユーザーにヴィジビリティドメインを付与する

ユーザーフォームで特定のユーザーのヴィジビリティドメインを追加することもできますが、グループを介してのみ追加することをお勧めします。そうすることで、個人が部門を変更したり退職したりする場合の権限とアクセス権がコントロールされます。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > ユーザー管理 > ユーザー.
2. ヴィジビリティドメインを提供するユーザーを選択します。
3. ヴィジビリティドメイン関連リストをフォームに追加します。
4. ヴィジビリティドメイン関連リストで [編集] をクリックします。
5. ユーザーに表示するレコードのドメインを選択します。
6. [保存] をクリックしてから、[更新] をクリックします。



ヴィジビリティドメインの埋め込みリストには、次のフィールドが含まれています。

フィールド	説明
ドメイン	グループまたはユーザーに表示されるドメイン。
継承	ドメインは、ドメインの可視化または親ドメインから継承されます。

フィールド	説明
許可者	ドメインの可視化を付与したグループの名前。
親の可視化	親ドメインの名前。レコードのグループ化に使用されます。親レコードが削除されると、同じ親を持つすべてのレコードも削除されます。

ドメイン固有の選択リストを作成する

アドミニストレーターは、特定のドメインに固有のエントリを含めるように選択リストを構成できます。

始める前に
必要なロール：admin

手順

1. 選択肢を追加するドメインピッカーからドメインを選択します。
2. カスタマイズする選択肢フィールドを右クリックし、[選択を構成] を選択します。
3. 選択肢を更新または追加します。
4. 更新セットなどの通常の変更プロセスを通じて変更をプッシュします。

> **注:** アドミニストレーターは、グローバルドメインでの管理上の混乱を防ぐために、ドメイン全体で選択肢が固有になるようにする必要があります。

アドミニストレーターがグローバルドメインから新しい選択肢を追加すると、階層内の下位のドメインのユーザーには、現在の選択リストの最後に新しい選択肢が表示されます。新しい選択肢がグローバルレベルでアクティブではない場合、ドメインユーザーは [選択を構成] で利用できますが、有効な選択肢としては表示されません。

詳細なドメインセパレーション管理

アドミニストレーターは、ドメインセパレーションに関する情報を表示し、潜在的な問題を特定し、構成設定を変更できます。

ドメインでは、次の高度な管理タスクを実行することができます。

- ドメイン選択メニューの使用
- ドメイン関係を表示する

ドメイン選択メニューの使用

インスタンスでは、2 つのメニュー形式でドメインを選択できます。

- ドメインセレクター：利用可能なドメインのシンプルなドロップダウンリストを提供します。
- ドメイン参照ピッカー：フィルタリングとオートコンプリート、先行入力機能を提供する参照フィールドを有効にします。長いリストにはこの形式を使用します。

これらのピッカーの配置と表示または非表示にする手順は、ユーザーインターフェイスのバージョンによって異なります。

コア UI のドメイン選択メニューを有効にする

コア UI でドメインピッカーを表示すると、デフォルトでドメインセレクターが有効になります。ドメインセレクターを有効にした後、システムプロパティを追加してドメイン参照ピッカーを有効にすることができます。

始める前に

注: ドメイン参照ピッカーを有効にするには、ドメインセパレーション (plugin com.snc.pa.domain_support) が必要です。

必要なロール：admin

このタスクについて

手順

1. ヘッダーの歯車アイコンをクリックします。
2. [全般] タブで、[ヘッダーにドメインピッカーを表示] スイッチをクリックします。
   ドメインセレクターが コア UI ヘッダーに表示されます。
3. オプション: ドメイン参照ピッカーを有効にします。

   注: ドメイン参照ピッカーを有効にすると、リストからグローバルオプションが削除されます。ホームドメインに戻るには、参照フィールドの横にある戻る矢印をクリックします。admin ユーザーは、戻る矢印をクリックしてグローバルドメインに戻ることができます。

   a. アプリケーションナビゲーターで「sys_properties.list」と入力します。
   b. まだ存在しない場合は、glide.ui.domain_reference_picker.enabled プロパティを追加し、値を **true** に設定します。
   c. ブラウザを更新します。
   ドメイン参照ピッカーが コア UI ヘッダーに表示されます。

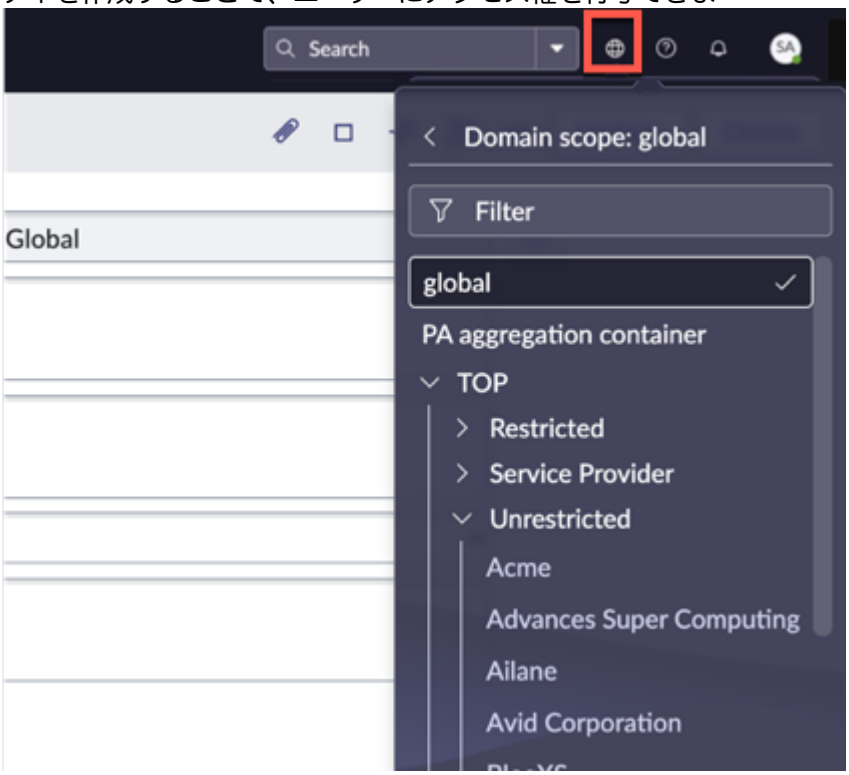
ドメインピッカーへのアクセスを制限する

システムプロパティを使用して、コア UI および ネクストエクスペリエンス のドメインピッカーへのアクセスを制限します。

始める前に
必要なロール：admin

このタスクについて
デフォルトでは、ITIL ロールを持つユーザー、および ITIL ロールを含むロール (アドミニストレーターなど) は、ネクストエクスペリエンス のドメインピッカーにアクセスできます。他のロールをプロパティに追加してアクセス権を付与するか、ロールを削除して制限することができます。ロールをアドミンのみに制限することをお勧めします。

アドミニストレーターは、sys_properties テーブルにシステムプロパティを作成することで、ユーザーにアクセス権を付与できます。



手順

1. システムプロパティ [sys_properties] テーブルを開きます。
2. glide.ui.polaris.domain_picker.role プロパティを追加します。
3. プロパティ値をロールのカンマ区切りリスト (admin、 **itil**) として設定します。
   詳細については、「Next Experience ピッカーを構成する」を参照してください。

ドメインセパレーションアプリケーションプロパティ

ドメインセパレーションプラグインには 2 つの新しいテーブルがあり、サービスプロバイダーはドメインセパレーションを使用するアプリケーションをより柔軟にカスタマイズできます。これらのテーブルは、システムアプリケーションプロパティテーブル [sys_application_property] とシステムアプリケーションプロパティ値テーブル [sys_application_property_value] です。

より多くのオプションを提供する新しいテーブル

サービスプロバイダー (SP) アプリケーションでは、ドメインによって特定のアクションが異なる場合があります。ただし、ServiceNow ベースシステムのシステムプロパティ [sys_properties] テーブルはドメインセパレーションされていないため、ドメインセパレーションを使用するアプリケーションの要件を満たしていません。

各 SP 顧客は、別の方法でのアプリケーションのカスタマイズを望む場合があります。以前は、カスタマイズ可能な機能は 1 つのグローバル値としてのみ定義されていました。アプリケーション開発者は、より柔軟なテーブルが必要です。機能を追加または変更するたびにコードを作成することなく、アプリケーションを変更できるようになりました。

新しいテーブルでの上書き方法

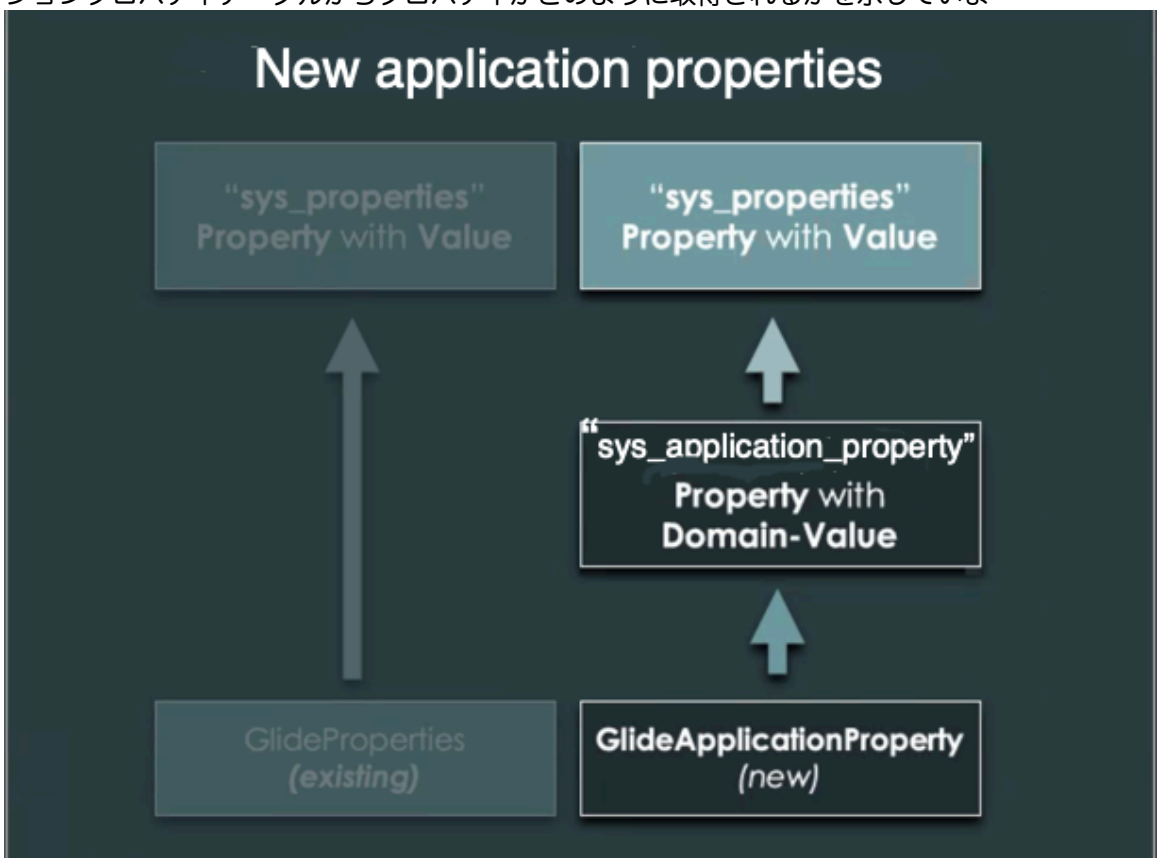
開発者は通常、ServiceNow システムプロパティ [sys_properties] テーブルを使用して、アプリケーションでさまざまな機能を作成します。異なるドメインで異なる動作をするアプリケーションを開発する場合は、自分でカスタマイズする必要があります。

Paris リリースでは、新しいアプリケーションプロパティ [sys_application_property] テーブルを使用するとそのカスタマイズが簡素化できます。システムプロパティテーブルの値に直接移動する代わりに、アプリケーションプロパティテーブルが最初にシステムアプリケーションテーブルに移動します。この新しいテーブルには、アプリケーションの設定に必要なロジックが格納されています。新しいテーブルでプロパティが見つかると、そのコンテンツが使用されます。そのテーブルに情報がない場合は、ベースシステムプロパティテーブルに移動します。

ドメインセパレーションのサポートを構成すると、この新しいアプリケーションプロパティテーブルにドメインロジックを追加できます。このテーブルには、システムプロパティテーブルに存在しないプロパティを含めることができます。または、システムプロパティテーブルで選択したプロパティを上書きできるプロパティを構成テーブルに追加することもできます。

たとえば、最初の曜日機能を使用してアプリケーションを構成するとします。週の最初の日を日曜日にしたい場合もあれば、別のケースでは、週の最初の日を月曜日にしたい場合もあります。ベースシステムテーブルには、1 日目のオプションが 1 つしかない (日曜日) 場合があります。新しいテーブルを使用して、別のプロパティで 1 日目を日曜日とし、子ドメインでは月曜日として格納することができます。

この図は、システムプロパティ [sys_properties] テーブルに移動する前に、アプリケーションプロパティテーブルからプロパティがどのように取得されるかを示していま



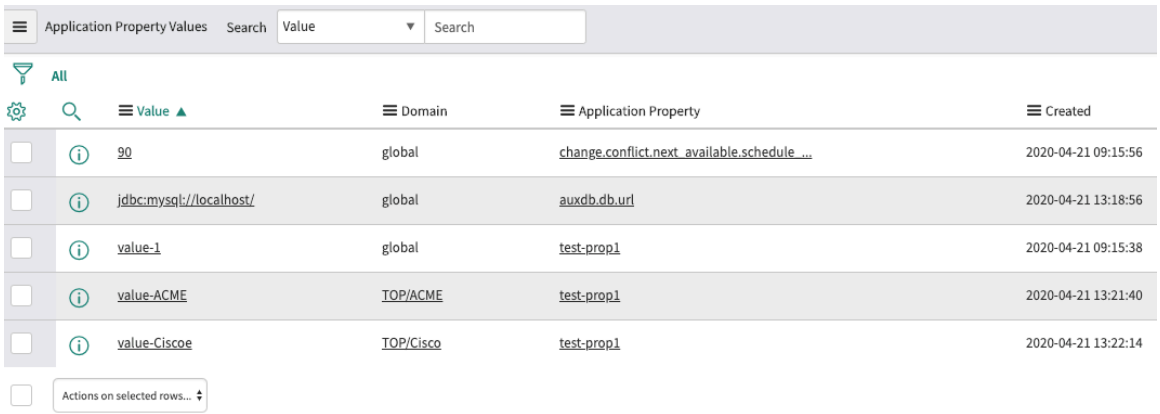
す。

新しいテーブルでのスコープ対象アプリの動作

新しいアプリケーションプロパティテーブルは、スコープ対象のアプリケーションからサポートされています。アプリケーションプロパティ名は、システムプロパティ名と同様に一意です。つまり、グローバルでない場合は、スコープ名がプリフィックスとして付きます。アプリケーションのスコープ

は構成に影響します。スコープによって、1 日目が日曜日と月曜日のどちらに定義されるかが決まります。同じプロパティを使用できますが、親ドメインでは 1 日目が日曜日、子ドメインでは 1 日目が月曜日になるようにカスタマイズできます。新しいテーブルにはドメイン列とスコープ列の両方があるため、それぞれに対してこれらのプロパティを設定できます。

次の画像に示すように、sys_application_property_value テーブルの [ドメインスコープを展開] ビューを使用して、すべての上書きを表示できます。



注:

これらのテーブルを利用できない場合は、ドメイン拡張インストーラー (com.glide.domain.msp_extensions.installer) プラグインが有効になっていることを確認してください。

新しいアプリケーションプロパティテーブル

新しいシステムアプリケーションプロパティ [sys_application_property] テーブルには、次のフィールドが含まれています。

- name
- 説明
- タイプ (文字列、true | false、整数、タイムゾーン、色など)
- default_value
- プロパティ (sys_properties への参照)
- usage_notes
- read_roles
- write_roles
- 一意のキー：(名前)

新しいシステムアプリケーションプロパティ値 [sys_application_property_value] テーブルには、次のフィールドが含まれています。

- sys_application_property (sys_application_property への参照)
- sys_domain
- sys_overrides
- value
- 一意のキー：(sys_application_property、sys_domain)

新しい API

新しい API はスコープ対象アプリでもサポートされています。ドメインセパレーションされたアプリケーションプロパティには個別の API があります。GlideApplicationProperty API には、グローバルアプリケーションとスコープ対象のアプリケーションの両方でスクリプト可能な 2 つの新しいメソッドがあります。これらの新しい API の詳細については、「GlideApplicationProperty - スコープ指定、グローバル」を参照してください。

関連トピック

- サービスプロバイダー向けのドメインセパレーションの推奨プラクティス

ドメイン移行ツール

ドメイン移行ツールを使用して、ドメインセパレーションされた環境から独自の専用インスタンスに顧客を移動させます。

ドメイン移行ツールプラグイン

ドメイン移行ツールプラグイン (com.glide.domain.migration_tool) により、ドメインセパレーションされた環境からより柔軟な専用インスタンスに顧客が移動するタスクを簡素化することができます。ServiceNow AI Platform の機能を最大限に活用するために、顧客は別のインスタンスに移行することもできます。ドメインセパレーションプラグインがインストールされていても、データとプロセスの分離プロパティはオフになっています。

> **注:** 使用するには、クローンされたインスタンスを要求し、ドメイン移行ツールプラグインのアクティベーションを要求する必要があります。

ドメイン移行ツールは、ドメインセパレーションされたインスタンスでデータとプロセスの両方の分離が有効になっている場合にのみ実行されます。

- *glide.sys.domain.partitioning* データプロパティは **true** に設定する必要があります。
- *glide.sys.domain.delegated_administration* プロセスプロパティは **true** に設定する必要があります。

移行ツールの機能

- 移行プロセスの多く、特にデータのクリーンアップを自動化します。
- ドメインセパレーションされたインスタンスを新しい専用インスタンスに移行します。
- 専用インスタンスからデータを削除します。

  > **注:** このツールは、グローバル、ターゲットドメイン、または追加のデータドメイン (指定されている場合) のデータを削除しません。

- プロセスデータを折りたたむか、折りたたむことができない場合はプロセスデータを削除します。
- ターゲットドメインに表示されるプロセスレコードを保持します。
- sys_choice、sys_ui_list、および sys_ui_related_list 特殊テーブルを更新します。
- ドメインセパレーションプラグインによって追加されたレコードをクリーンアップします。
  - ビジネスルール
  - UI アクション
  - スケジュール済みジョブ
  - インストレーションイグジット
  - ナビゲーションモジュール

- ドメインセパレーションを無効にし、クローンされたインスタンスからドメインを削除します。
  - クローンされたインスタンスで次のプロパティを **false** に設定します。
    - *glide.sys.domain.partitioning*
    - *glide.sys.domain.delegated_administration*
    - *glide.sys.domain.enabled*
  - ターゲットドメインと指定された追加のデータドメインを除くすべてのドメインを削除します。
- domain_migration_tool_status テーブルの [ステータス] フィールドを更新します。

テーブルの個々のステータス

ステータス	説明
処理待ち	移行中のドメインセパレーションテーブルのデフォルトのステータステーブルは移行される予定ですが、まだ移行が開始されていません。
失敗	テーブルレベルのエラー。移行プロセスがエラーで終了した場合、このステータスはどのテーブルにエラーがあるかを示します。
実行中	現在移行中のテーブルのステータス。1 つのテーブルのみがこのステータスになり、現在移行中です。
成功	正常に移行されたテーブルのステータス。
正常に完了しました	移行プロセスはエラーなしで終了しました。
エラーで終了	移行プロセスはエラーで終了しました。

- 進捗状況とステータスを syslog_domain に記録します。

  移行に関連するすべてのログエントリのソースは **MigrationTool** です。

  - 各データテーブルと残りのデータテーブルをログに記録します。
  - 各プロセステーブルと、現在非アクティブ化または削除されているドメインのレコードをログに記録します。

移行ツールに含まれない機能

- インスタンスをクローンします。
- 別のドメインセパレーションインスタンスを作成します。
- ツールを実行する前にデータまたはプロセス分離プロパティが無効になっている場合は、レコード (データまたはプロセス) を移行します。
- ソースインスタンスのデータを変更します。
- グローバル、ターゲットドメイン、または追加のデータドメイン (指定されている場合) のデータを削除します。

ツールの実行後に行う必要があること

ドメイン移行ツールは、目的のドメイン (ターゲットドメイン、追加のデータドメイン、およびグローバルドメイン) 以外のデータの削除を自動化します。残りの構成をすべて評価して、それらが適切であり、専用のインスタンスに対して機能することを確認する必要があります。たとえば、レコー

ドのドメインフィールドを設定するビジネスルールがある場合、このビジネスルールは役に立たないため、無効にすることができます。

ドメインセパレーションされたインスタンスを専用インスタンスに移行する

ドメインセパレーションされた環境から独自の専用インスタンス環境に顧客を移動します。

始める前に

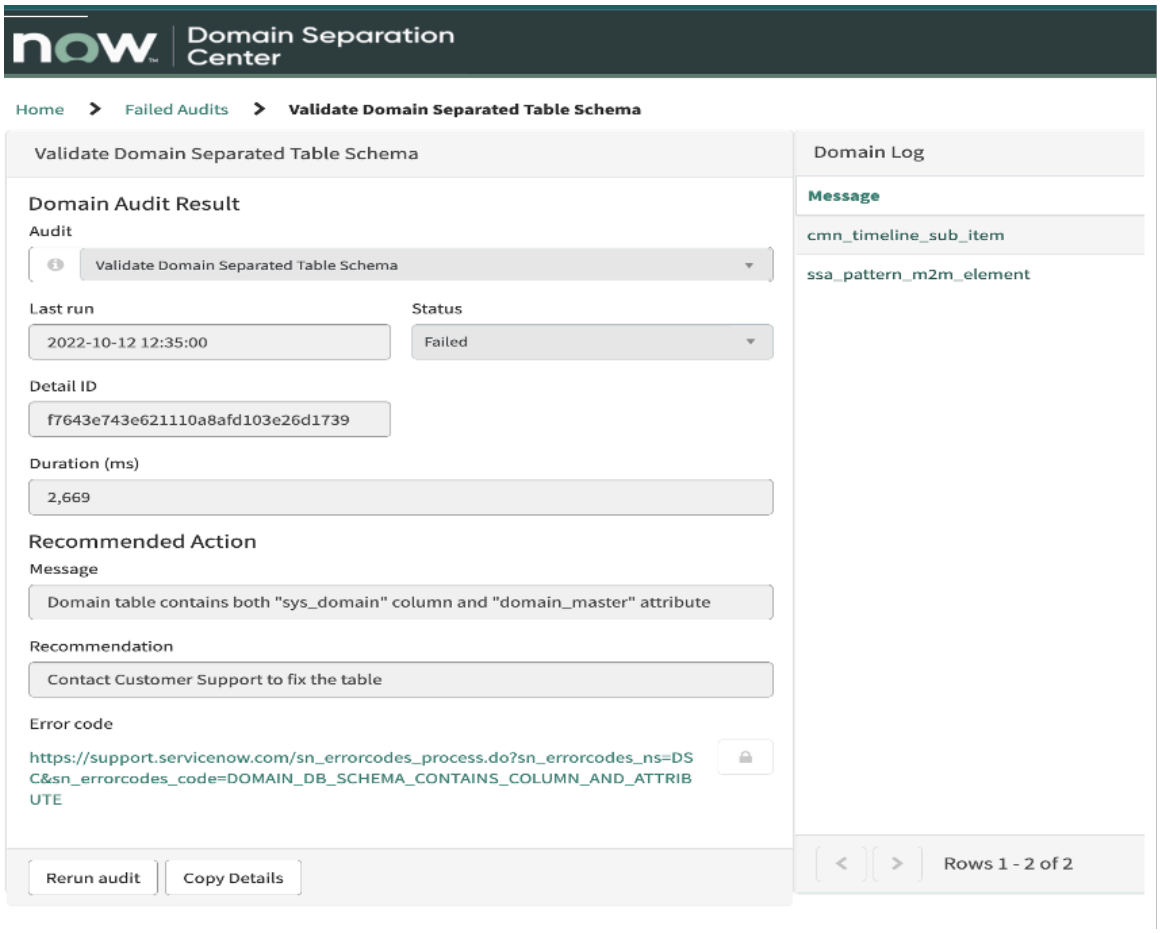
必要なロール：admin

手順

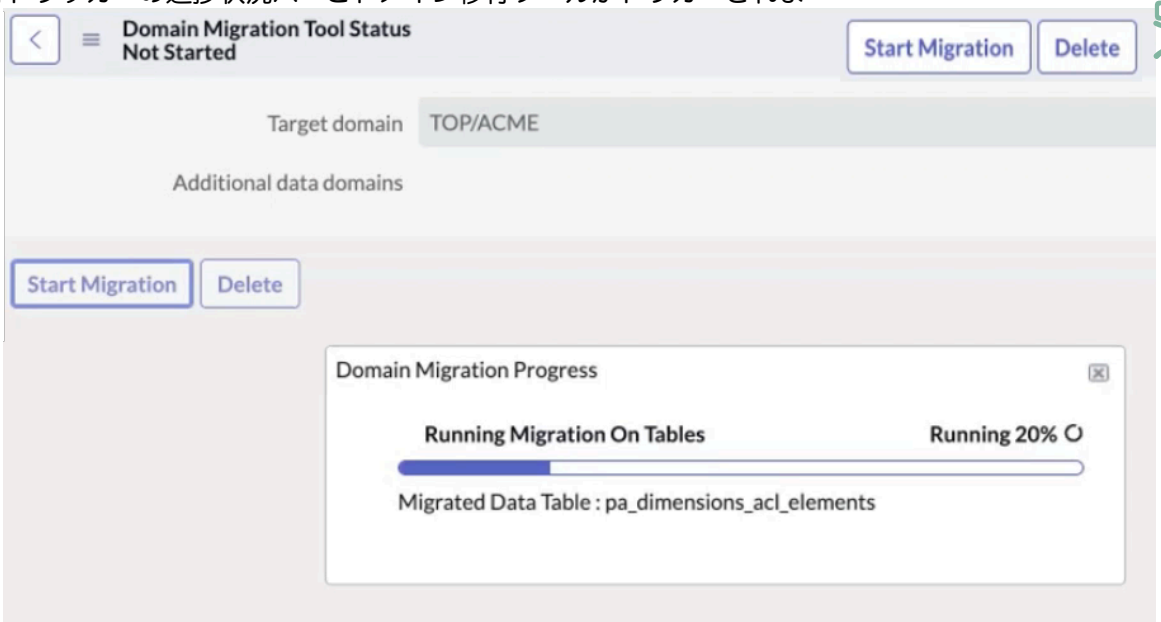
1. security_admin ロールに昇格させます。
   詳細については、「特権ロールへの昇格」を参照してください。
2. 移動先 すべて > ドメイン管理 > ドメイン移行ツール.
   domain_migration_tool_status.list にもアクセスします。
3. [新規] をクリックします。
4. フォームに入力します。

フィールド	説明
ターゲットドメイン	移行するプロセスとデータの両方に使用するドメインを指定します。[追加データドメイン] フィールドで指定されている場合を除き、子ではなくターゲットドメインのみが保持されます。
追加のデータドメイン	必要に応じて、移行する追加のデータドメインを指定します。ターゲットドメインとそのすべての子を移行する場合は、すべての子を指定する必要があります。

5. [送信] を選択します。
6. 送信したフォームを開きます。
7. 移動先 すべて > ドメインセパレーションセンター > 監査の設定 。
   詳細については、「ドメインセパレーションセンター」を参照してください。
8. [ドメインセパレーションテーブルのスキーマの検証 **(Validate Domain Separated Table Schema)]** 監査を [アクティブ] に設定し、スケジュールをアサインします。
   これは、ドメインセパレーションされたテーブルの健全性について事前にチェックするための予防的なドメイン監査です。これにより、移行を実行する前にエラーを修正できます。
9. スキーマを含む監査スケジュールを実行します。
   詳細については、「直ちに監査を実行する」を参照してください。
10. 監査から返された問題に対処します。



11. [移行開始] を選択します。
    - 実行トラッカーの進捗状況バーとドメイン移行ツールがトリガーされます。



    - 現在進行中のテーブルが、正常に移行されたテーブルの割合の合計とともに表示されます。
    - すべてのドメインセパレーションテーブルが、移行ステータス、各テーブルの合計レコード数、および移行されたレコードの数とともに記録されるテーブル。
    - 移行に失敗したテーブルの数も記録されます。
    - [ステータス] は、ツールがその機能を実行するときに更新されます。

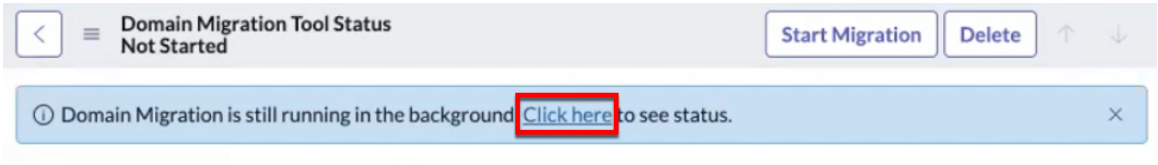
- 進捗状況バーを閉じても、ドメインの移行はバックグラウンドで実行されています。`sys_execution_tracker` テーブルにアクセスし、[テーブルで実行中の移行 **(Running Migration on Tables)**] を探して、バックグラウンドで実行されている移行プロセスを確認します。

ドメイン移行ツールのステータスには、次のフィールドが表示されます。

ドメイン移行ツールのステータスフィールド

フィールド	説明
ステータス	移行ステータスを表示します。 ○ データテーブルを移行しています...**(Migrating data tables...)**：進行中のステータス。 ○ 移行に成功しました **(Migration successful)**：移行が成功した後の更新されたステータス。 ○ ドメインセパレーションされたテーブルの **1** つで不適切な構成が観測されました **(Misconfiguration is observed on one of the domain separated tables)**：データ移行の失敗を示します。スキーマエラーが検出されました。移行は開始されません。[ドメインセパレーションテーブルのスキーマの検証 **(Validate Domain Separated Table Schema)**] 監査を実行します。監査は失敗し、スキーマ標準に従っていないテーブルが表示されます。 ○ テーブルの再アクセスでエラーが発生して終了しました **(Finished with Errors in Revisit Tables)**：スキーマの移行に対処するためにテーブルの数が表示されます。[再アクセステーブル数] でエラーを特定し、スキーマの問題に対処します。 ○ テーブルのドメイン移行に失敗しました...**(Domain Migration Failed for Tables...)**：失敗したテーブルからターゲット以外のレコードを手動で削除する必要があります。 ○ 正常に完了しました：すべてのテーブルが移行されました。
ターゲットドメイン	移行のために選択されたターゲットドメイン。
追加のデータドメイン	複数の移行ドメインが選択されている場合に入力します。
再アクセステーブル数	このフィールドは、テーブルの移行に失敗した場合にのみ入力されます。エラーがない場合、この数はゼロです。この場合、テーブルに再アクセスして移行を再試行します。エラーがない場合は、テーブルに再アクセスしたり、移行を再試行したりする必要はありません。
現在の進捗中のテーブル	現在移行中のテーブルの名前を表示します。移行が成功すると、このフィールドは空になります。

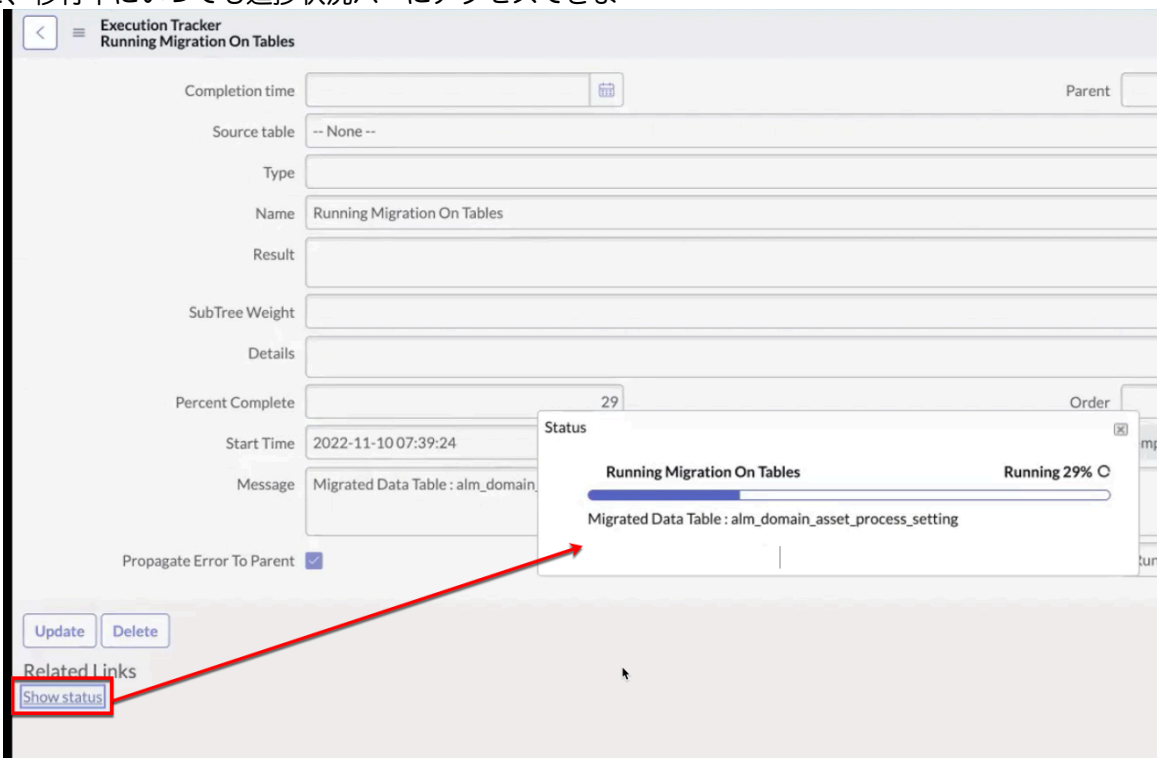
12. [ドメイン移行ツールのステータス] ページで [ここをクリック] リンクを選択して、実行トラッカーを開きま



す。

`sys_execution_tracker` テーブルにアクセスし、[テーブルで実行中の移行 **(Running Migration on Tables)]** を探して、バックグラウンドで実行されている移行プロセスを確認することもできます。

13. [関連リンク] セクションの [ステータスを表示] を選択すると、移行中にいつでも進捗状況バーにアクセスできま

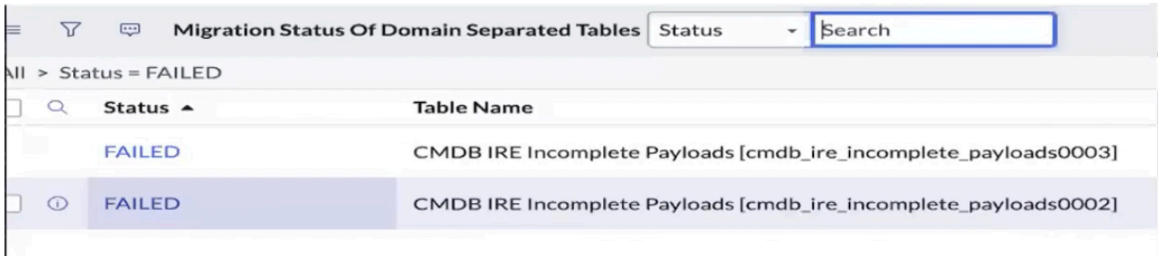


す。

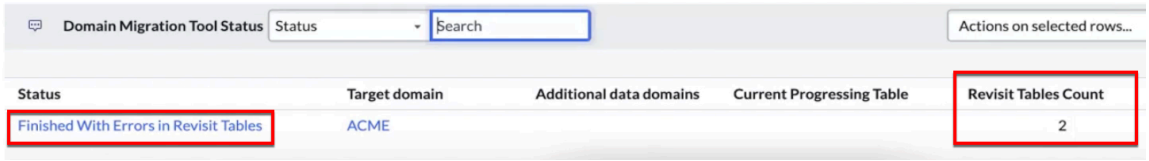
テーブルがスキーマチェックに失敗した場合、ドメインセパレーションテーブルの全体的な移行ステータスは [失敗] になりま

す。

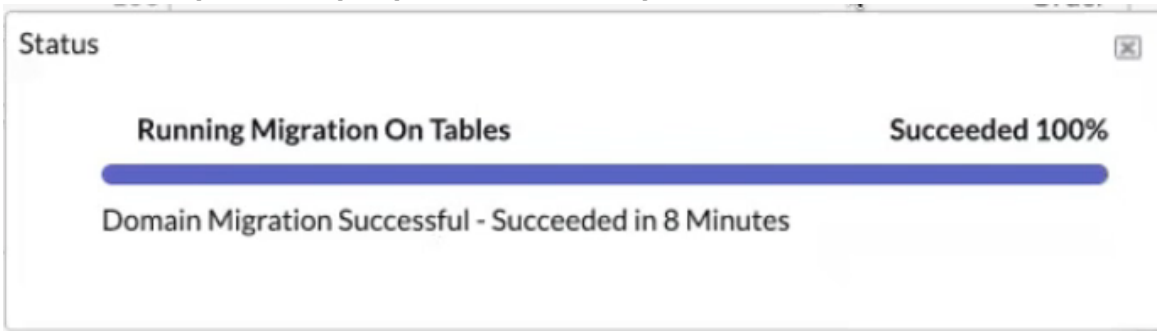
対応するテーブルごとに失敗したエントリがあります。



移行の残りの部分は続行され、失敗したすべてのテーブルのサマリーと再アクセステーブル数の合計数が、ドメイン移行ツールのステータスに入力されます。



移行が完了すると、[ステータス] は [正常に完了しました] になります。



プロセス管理

プロセス管理では、アドミニストレーターはドメイン固有のポリシーを設定できます。

ドメイン階層の上位に設定されたポリシーは、ドメイン階層の下位に設定されたポリシーで上書きされます。アドミニストレーターは、ドメイン内で次のグローバルポリシーと設定のドメイン固有のバージョンを設定できます。

- クライアントスクリプト
- システムポリシー
- アプリケーション名とモジュール名
- アプリケーションロール
- モジュールフィルター

⚠ **警告:** アドミニストレーターは ACL ルールをオーバーライドしてすべてのロールチェックに合格できるため、admin ロールを持つすべてのユーザーは、すべてのシステム機能およびデータへの特別なアクセス権を持ちます。この権限は慎重に付与してください。

ユーザーが **admin** ロールを持っている場合、割り当てられたドメインに関係なく、インスタンス内のすべてのポリシーを使用できます。特定のドメインに入ることができ、そのドメインとその上位ドメインのポリシーのみが、関連するトランザクション中に表示および処理されます。アドミニストレーターが上位のドメインまたはグローバルドメインにあるポリシーを変更すると、そのアドミニストレーターの現在のドメインの新しいレコードが自動的に作成されます。元のポリシー、アプリケーション、またはモジュールレコードは変更されません。元のレコードはこの新しいレコードで上書きされます。

下位レベルのドメインでポリシーを変更するには、そのドメインに移動してポリシーを変更します。このアプローチでは、元の上位レベルのポリシーレコードを上書きするポリシーレコードがドメインに作成されます。

上位レベルのポリシーを変更した後にそのポリシーの [ドメイン] フィールドを変更しないでください。このアプローチでは、下位レベルのドメインにポリシーレコードは作成されず、上位レベルのドメインのポリシーレコードは保持されません。

[sys_overrides] フィールドは、上位レベルのレコードが階層内の下位レベルのポリシー、アプリケーション、またはモジュールで上書きされることを示します。アドミニストレーターが階層の上位の別のドメインに属するポリシー、アプリケーション、またはモジュールを変更しようとすると、このフィールドが自動的に設定されます。

更新の試行は、上位レベルのレコードの変更ではなく挿入に変わり、**[sys_overrides]** フィールドは上書きされる上位レベルのポリシー、アプリケーション、またはモジュールを示すように設定されます。後で関連するトランザクションのレコードがロードされると、元のものの代わりに、上書きするドメイン固有のポリシー、アプリケーション、またはモジュールが使用されます。

プロセス管理のドメイン

デフォルトでは、プロセス管理は常にレコードのドメインを使用して、適用するポリシーを決定します。

レコードのドメインがユーザーのドメインよりも優先されます。レコードのドメインにポリシーがない場合、委任管理はドメイン階層の次に高いレベルのポリシーをチェックします。ドメインポリシーの検索は、グローバルドメインに到達するまでドメイン階層の上方向に続行します。ドメイン階層の下位にドメインポリシーがない場合、プロセス管理はグローバルドメインのポリシーを使用します。

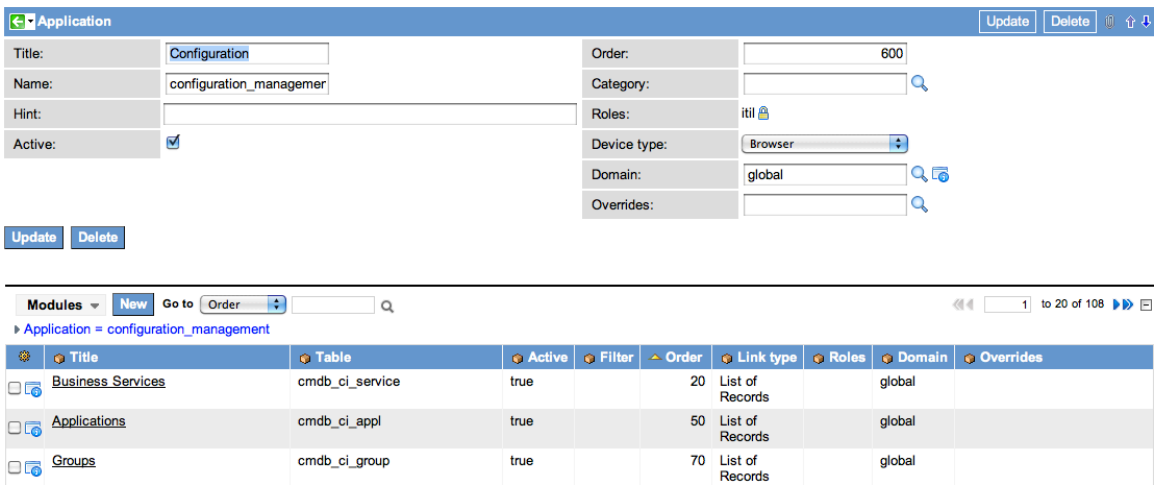
たとえば、Fred Luddy は Acme: Atlanta、Acme: San Diego、Acme: NY の子ドメインのレコードを表示できる Acme ドメインのユーザーであるとします。このユーザーが Acme: San Diego ドメイン内のレコードを開くと、プロセス管理では最初に Acme: San Diego ドメイン内のポリシーがチェックされます。ドメイン階層のこのレベルにポリシーがない場合、プロセス管理は Acme ドメインのポリシーをチェックします。Acme ドメインにポリシーがない場合、ドメイン階層の上位に他のドメインがないため、プロセス管理はグローバルドメインのポリシーを使用します。

ドメイン固有のアプリケーションを使用したプロセス管理の例

次の例は、ドメイン固有のアプリケーションとモジュールを使用したプロセス管理を示しています。

David Loo は、Oceanic ドメインのアドミニストレーターとして、構成アプリケーションをカスタマイズすることにしました。まず、David は構成アプリケーションモジュールで利用可能なモジュールを確認します。

構成アプリケーションの開始ビュー



David は、構成アプリケーションの名前を CMDB に変更し、inventory_admin ロールがアプリケーションを表示できるようにすることにしました。

構成アプリケーションに対するドメイン固有の変更の例

Applications New Go to Title 1 to 20 of 46

All > Active = true > Device type != Mobile

Title	Active	Order	Roles	Name	Domain	Overrides
Asset Management	true	900	asset	asset	global	
BSM Map	true		admin itil	bsm_map	global	
Change	true	400	itil	change_management	global	
CMDB	true	600	itil inventory_admin	configuration_management	Database	configuration_management
Content Management	true		content_admin	cms	global	
Contract Management	true	1,000	asset contract_manager	asset_contracts	global	
Domain Admin	true		domain_admin	domain_admin	global	
ECC	true		admin	ecc	global	
Homepage Admin	true		admin	home	global	
Incident	true	200	itil	incident_management	global	
Instance Clone	true		clone_admin	instance_clone	global	
Knowledge Base	true	800	knowledge	km	global	
Metrics	true		itil_admin metric_admin	metrics	global	
MID Server	true		admin	MID	global	
Organization Management	true	875	asset	organization_management	global	
Problem	true	300	itil	problem_management	global	
Reports	true	1,100	itil	reports	global	
SAML 2 Single Sign-on	true		admin	saml_2_single_sign_on	global	
SAML Single Sign-on	true		admin	SAML Single Sign-on	global	

Activate Deactivate Actions on selected rows... 1 to 20 of 46

次に、David は、開く - 「新規」ステータスモジュールを有効にして、Oceanic カテゴリのオープンインシデントを表示するように新しいフィルターアイテムを追加して、インシデントアプリケーションを変更することにしました。

オープン - 「新規」ステータスモジュールへのドメイン固有の変更の例

Module = Required field Update Delete

Title: Open - in "New" state
Link type: List of Records
Table: Incident [incident]
View name:
Order: 200
Roles:
Application: incident_management
Hint:
Active:
Image:

Filter:

Incident state is New
and Active is true
and Category is Database

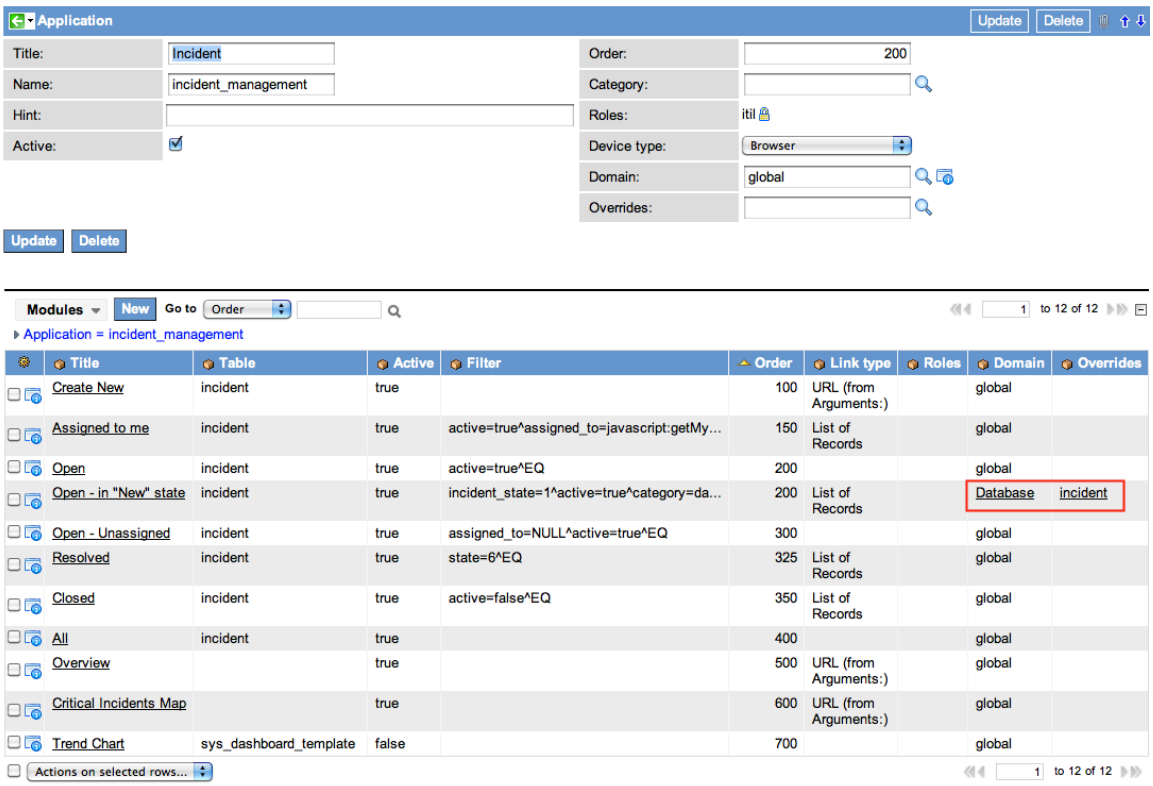
Arguments:

```
incident_state=1^active=true
```

Update Delete

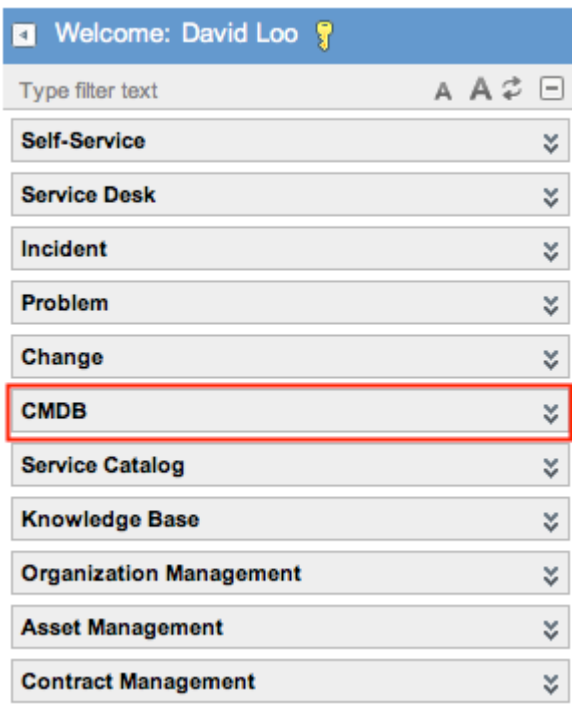
これにより、グローバルドメインの既存のモジュールを上書きするのではなく、アプリケーションに新しいモジュールエントリが作成されます。

インシデントアプリケーションのドメイン固有のビュー

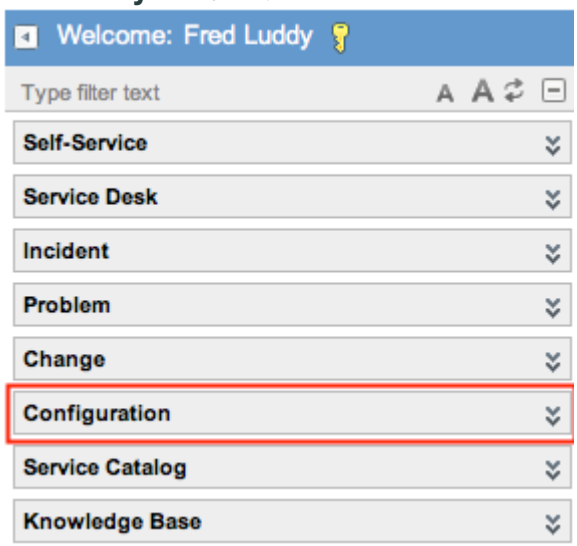


Fred Luddy など、別のドメインの別のアドミニストレーターがログインして構成アプリケーションを確認すると、グローバルドメインの設定が表示されます。

David Loo のアプリケーションビュー



Fred Luddy のアプリケーションビュー



詳細なドメインログ記録とデバッグメッセージを有効にする

ドメインログとデバッグメッセージを使用して、ドメイン構成エラーのトラブルシューティングを行うことができます。

始める前に

必要なロール：admin
最高のパフォーマンスを得るには、最新のインスタンスを使用してください。

手順

1. ドメインセパレーションセンターで、[ドメイン管理] に移動します。
2. [ドメインセンターの構成] をクリックします。
3. [詳細なドメインログ記録を有効にする] で、[はい] チェックボックスをオンにします。
4. [更新] をクリックします。

リアルタイムにドメインメッセージを表示する

システムログからリアルタイムにドメインメッセージを表示できます。

始める前に

必要なロール：admin

手順

1. 詳細なドメインログ記録を有効にする:移動先 すべて > ドメイン管理 > ドメインセパレーションセンター > ドメインセンターの構成 > 詳細なドメインログ記録を有効にします > はい (またはプロパティ `glide.sys.domain.verbose` を **True** に設定します)。
2. 移動先 システム診断 > セッションのデバッグ > すべて有効化.
   これはリアルタイムのレビューであるため、ログファイルをチェックする前にデバッグセッションを実行する必要はありません。
3. セッションデバッグコンソールに移動して、詳細なシステムドメインログを表示します。
4. 「 テーブルに対してクエリーを実行する (Query against table)」というテキストを検索します。

   このクエリーは、次の形式のログメッセージを検索します。

```
08:36:43.974: [Domain Paths] Query against table incident restricted by domain values
 [Database Atlanta[db53580b0a0a0a6501aa37c294a2ba6b],
Database[287ee6fea9fe198100ada7950d0b1b73],
Database San Diego[db53a9290a0a0a650091abebccf833c6], global, NY
 DB[5f74727dc0a8010e01efe33a251993f9]]
```

この例では、インシデントテーブルを表示しているユーザーは、Database Atlanta、Database、Database San Diego、グローバル、および NY DB の各ドメインに一致するレコードのみを表示しています。

履歴ドメインメッセージの表示

ログファイル内の履歴ドメインメッセージを表示して、ドメインセパレーションの問題をトラブルシューティングします。

始める前に
必要なロール：admin

手順

1. 詳細なドメインログ記録を有効にする:移動先 すべて > ドメイン管理 > ドメインセパレーションセンター > ドメインセンターの構成 > 詳細なドメインログ記録を有効にします > はい (またはプロパティ glide.sys.domain.verbose を **True** に設定します)。
2. 移動先 システム診断 > セッションのデバッグ > すべて有効化.
3. ログファイルをチェックする前に、1 日などの期間デバッグセッションを実行します。
4. 移動先 システムログ > ユーティリティ > ノードログファイルのダウンロード.
5. 表示する日のレコードを開きます。
   ログファイルでは、localhost_log.<yyyy-mm-dd>.txt という命名形式が使用されます。
6. ダウンロードログ関連リンクをクリックします。
7. ダウンロードしたログファイルをテキストエディターで開き、次の形式のログメッセージを検索します。

   ドメイン値によって制限されたテーブルインシデントに対するクエリー [global, Software[8a4dde73c6112278017a6a4baf547aa7]]

   この例では、グローバルドメインとソフトウェアドメインに一致するインシデントテーブルのレコードのみが表示されます。

ドメインセパレーションエラーのトラブルシューティング

ドメインセパレーションの問題が発生した場合は、このソリューションのリストを確認してください。

エラーまたは症状	解決策
ドメインの sys_id が存在しないドメインを指している	このエラーは、ユーザーレコードやタスクレコードなどのデータレコードに、現在のドメインテーブルに sys_id が存在しない sys_domain 列値がある場合に発生します。ドメインの sys_id が誤って削除されたか、ドメインテーブルを変更した場合は以前のドメインテーブルを参照している可能性があります。 エラーを修正するには、エラーを含むテーブルのリストを開き、無効な sys_domain 値をフィルタリングします。次に、正しい sys_domain 値を手動で入力するか、削除します。

エラーまたは症状	解決策
	注: ドメインテーブルを参照するすべてのテーブルに、無効なドメインの sys_ids を含めることができます。たとえば、無効なドメイン ID がユーザーヴィジビリティドメイン [sys_user_visibility]、グループヴィジビリティドメイン [sys_user_group_visibility]、および包含ドメイン [domain_contains] テーブルで発生する可能性があります。
ドメインパスまたはドメイン番号 sys_id が間違ったドメインを指している	このエラーは、ドメイン番号またはドメインパスクエリーが実際のドメイン名と同期していない場合に発生します。このエラーは、ドメインを追加する際に番号の付け直しが必要な場合、またはドメイン番号からドメインパスへの変換中にドメイン番号で発生する可能性があります。 エラーを修正するには、ドメインセパレーションセンターで結果を確認します。エラーが解決しない場合は、sys_domain_path 列または sys_domain_number 列の値を手動で編集して、適切なドメインを指すようにします。
ドメインツリー構造が破損している	このエラーは、一連のドメインにドメイン間で無限ループが発生する関係が含まれている場合に発生します。 エラーを修正するには、ドメインテーブルのリストを開き、ループを形成しないようにドメインが含まれている値を手動で編集します。

本番後のドメインセパレーション有効化ユーティリティ

本番後のドメインセパレーションアクティベーションユーティリティは、稼働環境におけるドメインセパレーションのアクティベーションを支援します。

Post-Production ドメインセパレーション有効化ユーティリティプラグイン

本番後のドメインセパレーション有効化ユーティリティプラグイン (com.glide.domain.activation_utility) を使用すると、本番後のライブ環境でドメインセパレーション環境を作成するタスクが簡素化されます。お客様は、 ServiceNow AI Platform 機能をさらに活用するために、本番後の環境でドメインセパレーションをアクティブ化することをお勧めします。このユーティリティは、ドメインを作成するためのステップバイステップのガイド付きセットアップを提供します。

注: ドメインセパレーションされたテーブルにドメインレコードを入力するプロセスには、ダウンタイムまたはインスタンスへのアクセス制限が必要です。

アクティベーションユーティリティの機能

- ドメインを作成するためのステップバイステップのガイド付きセットアップを提供します。
- バックグラウンドジョブを実行して、プロセスプロパティとデータ分離プロパティの両方を無効にしてドメインセパレーションのインストールを処理します。
- ターゲット (最初の) ドメインに表示されるプロセスレコードを保持します
- ドメイン作成プロセス中にエラーを検出、識別、およびログに記録します
- 一般的なエラーの解決策を提示します
- 監査を目的として、セットアッププロセス中に実行されたすべてのアクションをログに記録します
- 実行されたすべてのアクション、作成されたドメイン、システムに加えられた変更を含む詳細なサマリーを生成します

ドメインジョブ管理

ドメインジョブマネージャーを使用して、複数のドメインセパレーションの更新を単一のバックグラウンドジョブに順番にキューに入れます。

親の再指定や階層の変更など、ドメインレコードが変更されると、デフォルトでは、ドメインセパレーションテーブル内のすべてのレコードを修正するためのジョブが実行されます。この動作がいつ発生するかを ドメインジョブマネージャー で制御し、[ジョブの進行状況の表示] ボタンで進行中のジョブの進行状況を監視できます。

ドメインジョブマネージャーを使用して、ドメインジョブを一時停止し、複数のドメインテーブル変更をキューに入れ、すべての変更の準備ができたらキューに格納されたジョブをトリガーします。

ドメインジョブの進捗状況

ラベル	説明
ジョブタイプ	現在のジョブタイプ
ステータス	ジョブの状況については、ジョブ・マネージャーの ジョブ状況 を参照してください。
進捗状況	進行中のジョブの進捗状況

ドメインジョブマネージャー

フィールド	説明
ジョブタイプ	サポートジョブタイプ: 階層変更
ジョブ状況	• アクティブ • 一時停止 注: デフォルトでは、一時停止したジョブは 1 時間で自動的に再開されます。このオプションをオンにしてこれを無効にし、後でジョブを手動でアクティブ化します

ドメイン別に削除

ドメイン階層内の非アクティブなリーフレベルドメインを、制御および自動化されたツールを使用してクリーンアップします。

始める前に

必要なロール:ドメインセパレーションアドミン

手順

1. すべて > ドメイン管理 > クリーンアップキュー
2. [クリーンアップの準備] ボタンを選択します。
3. 削除のキューに入れるドメインを選択します。

リストには、削除対象として選択できるすべての非アクティブなリーフレベルドメインが表示されます。

4. **[Save** (保存)] を選択します。
   選択したドメインに、テーブルに **[Ready for Staging**] ステータスが反映されます。ドメインは、削除のキューに入れる前にステージングする必要があります。
5. [ステージングに移動] ボタンを選択します。
6. 削除前にステージング済みドメインを保持する日数を入力します。

   注: ドメインが [ステージング済み] および [削除計画] ステータスに移行すると、クリーンアップキューから削除することはできません。

ドメインが削除される前に、ドメインアドミニストレーターはドメインのデータフットプリントにアクセスできます。

7. [**Preview Domain Data**] ボタンを選択します。
   非アクティブなドメインでメタデータをスキャンするジョブがキューに入れられます。
8. ジョブの実行後、メタデータを表示するドメインレコードを選択します。

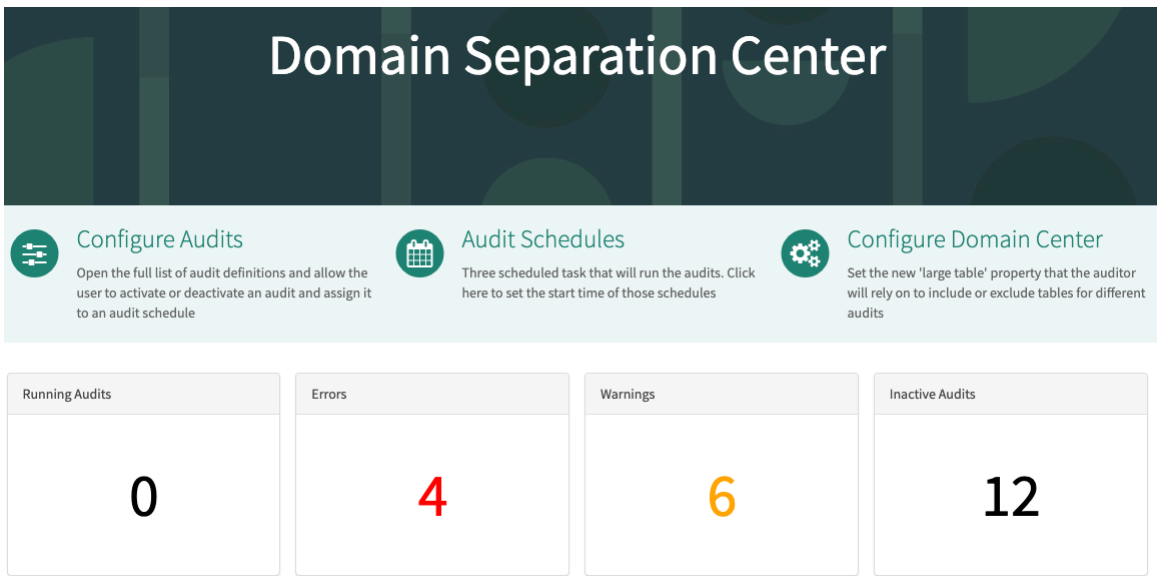
ドメインセパレーションセンター

ドメインを定期的に監査して問題を明らかにします。

概要

ドメインセパレーションセンターは、domain_audit_definition テーブルに格納されているすべてのドメインの監査をスケジュールおよび設定できるダッシュボードです。このダッシュボードを使用すると、監査結果を確認し、ドメインのエラーと警告をより深く掘り下げることができます。ドメインセパレーションセンターのダッシュボードは、<ServiceNow-instance-name>/domaincenter にあります。

ドメインセパレーションセンターは多くの監査を提供します。独自の監査を作成することはできません。ただし、実行頻度は設定できます。監査は、domain_audit_definition テーブルに格納されているすべてのドメインで実行されます。



監査を設定する

監査がアクティブかどうかと実行頻度を設定できます。

1. ドメインセパレーションセンターで、[監査の設定] をクリックします。

   [監査の設定] ページが表示されます。

2. アクティブにする各監査を設定します。監査はデフォルトで無効にされています。
   a. 監査を選択します。
   b. [アクティブ] ボックスを選択して監査をアクティブにします。
   c. [頻度] フィールドで監査の実行頻度を指定します。

      ドメインセパレーションセンターで同じ頻度にマークされているすべての監査を同時に実行します。大きなテーブルで実行する監査に [日次] を選択しないでください。

   d. アクティブにする各監査でこれらの手順を繰り返します。
3. [保存] を選択します。

監査スケジュール

1 つ以上の監査を日次、週次、または月次で実行するように設定します。スケジューラは、これらの監査を実行する日時を指定します。同じスケジュール頻度のすべての監査は、設定した時刻から順番に実行されます。

1. ドメインセパレーションセンターで、[監査スケジュール] を選択します。

   [監査スケジュール] ページが表示されます。

2. 監査スケジュールを設定します。
   a. スケジュール名を選択します。
   b. 監査を実行する時刻を指定します。

      時間単位は 24 時間形式です。つまり、14 は午後 2 時です。

   c. 週次スケジュールの場合は、監査を実行する曜日を選択します。1 は日曜日です。
   d. 月次スケジュールの場合は監査を実行する日付になります。
   e. 他のスケジュールについても、この手順を繰り返します。
3. [保存] を選択して設定の変更を保存します。
4. [今すぐ実行] を選択すると、右側のペインの上部に表示される頻度で実行するようにスケジュールされたすべての監査が実行されます。たとえば、ペインのタイトルが「ドメイン監査スケジュール - 日次」の場合、毎日実行されるすべての監査が実行されます。

実行中の監査のステータスを表示するには、ドメインセパレーションセンターで [監査の実行] ボックスの番号を選択します。

直ちに監査を実行する

監査は通常、スケジュールどおりに実行されます。ただし、コマンドですべての監査を実行できます。

1. ドメインセパレーションセンターで、[監査スケジュール] を選択します。

   [監査スケジュール] ページが表示されます。

2. 実行する監査スケジュールの名前 (日次、週次、月次など) を選択します。
3. [今すぐ実行] を選択します。

実行中の監査のステータスを表示するには、ドメインセパレーションセンターで [監査の実行] ボックスの番号を選択します。

ドメインセパレーションセンターを構成する

1. ドメインセパレーションセンターで、[ドメインセンターの構成] を選択します。
2. [ドメインセンターの構成] ページの [詳細なドメインログ記録を有効にする **(Enables detail domain logging)]** で [はい] を選択して、ドメイン関連の問題の診断に役立つ詳細なログを保存します。詳細なログ記録は、パフォーマンスの問題を引き起こす可能性があります。

   これらのログは、syslog_domain テーブルのサーバー側のログを参照します。ログの表示の詳細については、「警告と失敗を含む監査を表示する」を参照してください。
3. スラッシュバケットで、すべての大きなテーブルを選択して [選択済み] 列に移動します。

   日次監査は大きなテーブルでは実行しないでください。[選択済み] 列のグレーアウトされたテーブル名は大きいため、[利用可能] 列に移動できません。
4. [更新] を選択します。

警告と失敗を含む監査を表示する

ダッシュボードの [エラー] および [警告] は、問題が発生した監査に関する詳細情報を提供します。ドメインセパレーションに関連するログは、サーバーの syslog_domain テーブルにあります。

1. ドメインセパレーションセンターで、[エラー] または [警告] ボックスの数字を選択します。

   [エラー] または [警告] ページがそれぞれ表示されます。
2. リスト内のいずれかの監査を選択します。

   選択した監査の問題に関する詳細情報がページに表示されます。[監査結果の詳細] のメッセージは、エラーまたは警告を明らかにしたテーブル監査の syslog_domain テーブルの値を参照します。
3. 警告またはエラーの監査に関する情報を提供するサーバーのログを表示するには、次の手順を実行します。
   a. 警告またはエラーの左パネルの [詳細 **ID]** の値をコピーします。
   b. HI のフィルターナビゲーターで「*syslog_domain.do*」と入力します。

      [ドメインログ] ページが表示されます。
   c. [ソース] 検索フィールドに「=<詳細 ID>」 (等号の後にスペースを入れない) を入力します。たとえば、「=f6a00fd29a85b300a9503a81b9169678」と入力します。

      [ドメインログ] ページには、指定した詳細 ID の監査に関連するログのみが表示されます。テーブルの各行には、監査で見つかった問題が明記されます。このページの [メッセージ] フィールドは、ドメインサービスセンターの [メッセージ] 列に表示される値と一致します。メッセージの形式は監査タイプと一致します。
4. 次から選択します。

- [監査を再実行]：監査を再実行して、まだ警告またはエラーが発生するかどうかを確認します。
- [監査の非アクティブ化]：監査を非アクティブにします。
- [詳細をコピー]：監査の詳細をクリップボードにコピーします。

実行中の監査と処理待ちの監査を表示する

アクティブな監査は、スケジュールどおりに定期的に実行されるか、実行キューに格納されます。実行中のステータスを表示できます。

1. [監査の実行] ボックスの数字を選択して、現在実行中または実行処理待ちの監査を表示します。

   [監査の実行] ページが表示されます。

2. 詳細については、監査を選択してください。

非アクティブな監査を表示する

1. [非アクティブな監査] ボックスで数字を選択します。

   [非アクティブな監査] ページには、現在非アクティブ化されているすべての監査が表示されます。

2. いずれかの監査を選択すると、それに関する詳細情報が表示されます。
3. 監査をアクティブにするには、[アクティブ] ボックスを選択し、[頻度] フィールドを使用して監査の実行頻度を指定します。
4. [更新] を選択します。

ドメインセパレーションセンターを構成する

ドメイン内のどのテーブルが大きいか、および詳細なログ記録が必要かどうかを指定します。

始める前に

必要なロール：admin

このタスクについて

監査は、日次、週次、または月次ベースで実行するようにスケジュールできます。大きなテーブルで実行される監査には、かなりの時間がかかります。そのため、大きなテーブルで毎日監査を実行しないようにしてください。実行に 1 日以上かかっている新しい監査を開始すると、悪影響を与える可能性があります。

詳細なログ記録により、監査中に見つかった問題についてより詳細な情報を得ることができますが、パフォーマンスの問題が発生する場合があります。

手順

1. ドメインセパレーションセンターで、[ドメインセンターの構成] を選択します。
   [ドメインセンターの構成] ページが表示されます。
2. [詳細なドメインログ記録を有効にする **(Enables detail domain logging)**] で [はい] を選択して、ドメイン関連の問題の診断に役立つ詳細なログを保存します。
   これらのログは、ドメインログ [syslog_domain] テーブルのサーバー側のログを参照します。ログの表示の詳細については、「警告と失敗を含む監査を表示する」セクションを参照してください。
3. リストで、すべての大きなテーブルを [選択済み] 列に移動します。
   日次監査は大きなテーブルでは実行しないでください。[選択済み] 列のグレーアウトされたテーブル名は大きいため、[利用可能] 列に移動できません。

4. [更新] を選択します。
5. 監査する必要がない大きなテーブルがある場合は、*com.glide.domain.audit.big_tables.additional* システムプロパティをそれらのテーブル名のカンマ区切りリストに設定します。

監査の設定

監査がアクティブかどうか、および監査の実行頻度を設定します。

始める前に

必要なロール：admin

手順

1. ドメインセパレーションセンターで、[監査の設定] をクリックします。
   T が表示されます。
2. [監査の設定] ページで、アクティブにする各監査を設定します。
   監査はデフォルトで無効にされています。
   a. [監査] をクリックします。
   b. [アクティブ] チェックボックスをクリックして監査を有効にします。
   c. [頻度] フィールドで監査の実行頻度を指定します。
      ドメインセパレーションセンターで同じ頻度にマークされているすべての監査を同時に実行します。大きなテーブルで実行する監査に [日次] を選択しないでください。
   d. アクティブにする各監査で a ～ c の手順を繰り返します。
3. [保存] をクリックします。

監査のスケジュール

監査を実行する日時を指定します。

始める前に

必要なロール：admin

このタスクについて

1 つ以上の監査を日次、週次、または月次で実行するように設定しました。スケジューラは、これらの監査を実行する日時を指定します。同じスケジュール頻度のすべての監査は、設定した時刻から順番に実行されます。

手順

1. ドメインセパレーションセンターで、[監査スケジュール] をクリックします。
   [監査スケジュール] ページが表示されます。
2. 監査スケジュールを設定します。
   a. スケジュール名をクリックします。
   b. 監査を実行する時刻を指定します。
      時間単位は 24 時間形式 (14 は午後 2 時) です。

c. 週次スケジュールの場合は、監査を実行する曜日を選択します。1 は日曜日です。

d. 月次スケジュールの場合は監査を実行する日付になります。

e. 他のスケジュールについても、この手順を繰り返します。

3. [保存] をクリックして設定を保存します。

4. オプション: [今すぐ実行] をクリックすると、右ペインの上部に表示されている頻度で実行するようにスケジュールされたすべての監査が実行されます。
たとえば、ペインタイトルが「ドメイン監査スケジュール - 日次」の場合、[今すぐ実行] は毎日実行するようにスケジュールされているすべての監査をすぐに実行します。

次のタスク
実行中の監査のステータスを表示するには、ドメインセパレーションセンターで [監査の実行] ボックスの番号をクリックします。

直ちに監査を実行する

監査は通常、スケジュールどおりに実行されます。ただし、コマンドですべての監査を実行できます。

始める前に
必要なロール：admin

このタスクについて
個々の監査を手動で実行することはできません。ただし、同じスケジュール頻度で設定されたすべての監査を実行できます。たとえば、毎日実行するように設定されているすべての監査を実行できます。

手順

1. ドメインセパレーションセンターで、[監査スケジュール] をクリックします。

2. 実行する監査スケジュールの名前 (日次、週次、月次など) をクリックします。

3. [今すぐ実行する] をクリックします。

次のタスク
実行中の監査のステータスを表示するには、ドメインセパレーションセンターで [監査の実行] ボックスの番号をクリックします。

警告と失敗を含む監査を表示する

ドメインセパレーションセンターは、監査エラーと警告に関する詳細を提供します。

始める前に
必要なロール：admin

このタスクについて
ドメインセパレーションダッシュボードの [エラー] および [警告] は、問題が発生した監査に関する詳細情報を提供します。ドメインセパレーションに関連するログは、サーバーの syslog_domain テーブルにあります。エラーは、即時の対応が必要な問題です。警告は失敗につながることはありませんが、ドメイン名を長くしすぎないなどのベストプラクティスを示します。

手順

1. ドメインセパレーションセンターで、[エラー] または [警告] ボックスの数字をクリックします。
   [エラー] または [警告] ページが表示されます。
2. リスト内のいずれかの監査をクリックします。
   選択した監査の問題に関する詳細情報がページに表示されます。[監査結果の詳細] のメッセージは、エラーまたは警告を明らかにしたテーブル監査について、サーバー上の syslog_domain テーブルの値を参照します。
3. 警告またはエラーの監査に関する情報を提供するサーバーのログを表示するには、次の手順を実行します。

   a. 警告またはエラーの左パネルの [詳細 **ID]** の値をコピーします。

   b. ドメインセパレーションセンターを実行しているインスタンスで、[フィルターナビゲーター] に「*syslog_domain.list*」と入力します。
   [ドメインログ] ページが表示されます。

   c. [ソース] 列の検索フィールドに、「=<詳細 ID>」 (等号の後にスペースを入れない) を入力します。たとえば、「=f6a00fd29a85b300a9503a81b9169678」と入力します。
   [ドメインログ] ページには、指定した詳細 ID の監査に関連するログのみが表示されます。テーブルの各行には、監査で見つかった問題が明記されます。このページの [メッセージ] フィールドは、ドメインサービスセンターの [メッセージ] 列に表示される値と一致します。メッセージに含まれる情報は監査タイプと一致します。

4. オプション: 次のいずれかを選択します。
   - [監査を再実行]：監査を再実行して、まだ警告またはエラーが発生するかどうかを確認します。
   - [非アクティブ化]：監査を非アクティブにします。
   - [詳細をコピー]：監査の詳細をクリップボードにコピーします。

実行中の結果と処理待ちの結果を表示する

実行中の監査と処理待ちの監査を表示して、そのステータスを確認できます。

始める前に

必要なロール：admin

このタスクについて

アクティブな監査は、スケジュールどおりに定期的に実行されるか、実行キューに格納されます。実行中のステータスを表示できます。

手順

1. [監査の実行] ボックスの数字をクリックして、現在実行中または実行処理待ちの監査を表示します。
   [監査の実行] ページが表示されます。
2. 実行中の監査または処理待ちの監査をクリックすると、それに関する情報が表示されます。

非アクティブな監査を表示する

すべての非アクティブな監査を 1 つの場所で表示し、オプションでそれらをアクティブにすることができます。

始める前に
必要なロール：admin

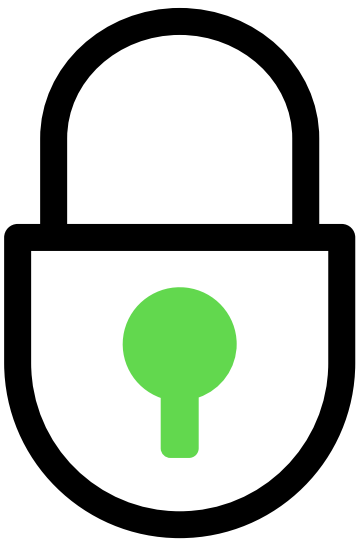
手順

1. [非アクティブな監査] ボックスの数字をクリックします。
   [非アクティブな監査] ページには、現在非アクティブ化されているすべての監査が表示されます。
2. いずれかの監査をクリックすると、それに関する詳細情報が表示されます。
3. オプション: 監査をアクティブにするには、[アクティブ] ボックスをクリックし、[頻度] フィールドに監査の実行頻度を指定します。
4. [更新] をクリックします。

認証

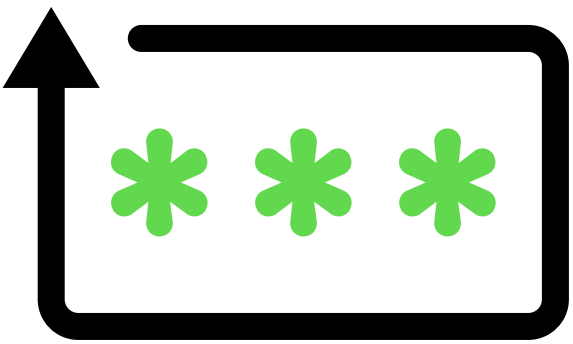
ServiceNow の認証は、インスタンスにアクセスするユーザーの ID を検証し、ユーザーのロールまたは職務に一致する機能に対してユーザーを許可します。

開始



複数プロバイダーのシングルサインオン **(SSO)**

データベース内に一致するユーザーアカウントがある ID プロバイダーで構成されたユーザー名とパスワード。



OAuth 受信と送信

OAuth ベースの認証は、認証プロトコルを使用してシステムで信頼を確立しようとするクライアントの ID を検証します。

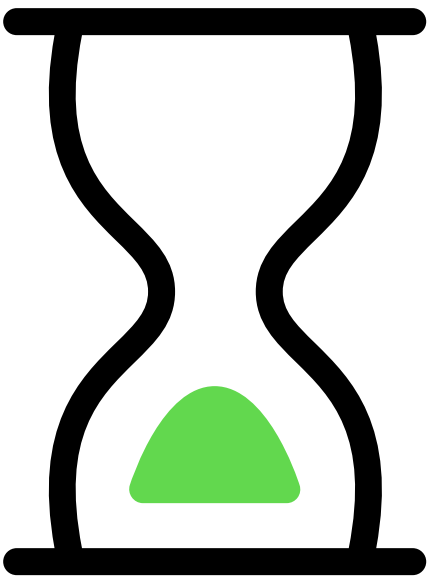
自動翻訳

API アクセスポリシー

API アクセスポリシーは、ポリシーを通じて制御できる、API に対する権限とアクセスを定義します。

ダイジェストトークン認証

テーブル内のユーザー名とシークレットは、SHA1、SHA 256、MD5 などのユーザー固有のハッシュ動作を実行します。この値は、URL サフィックスの一部として追加する必要があります。これはクエリーパラメーターで機能します。



時間制限付き認証

多要素認証 **(MFA)**

MFA を使用すると、認証アプリ、ハードウェアキー、生体認証装置、SMS、またはメールからのパスコードの使用を含む第 2 レベルの認証を提供できます。

自動翻訳

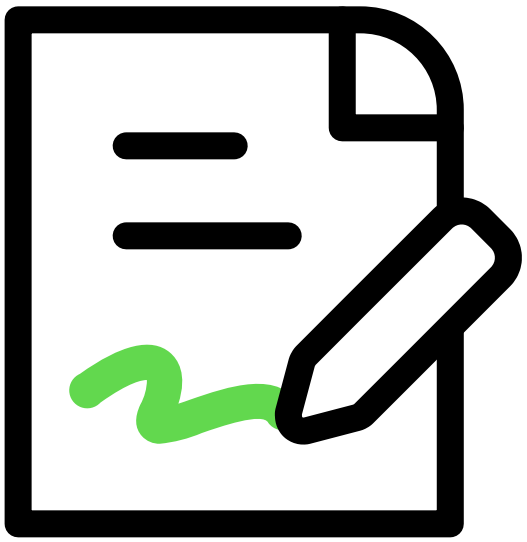
でのリンクベースの認証の構成

ServiceNow

インスタンス構成されたリンクは、メールまたは SMS を介してユーザーと共有でき、ユーザーはそれらのリンクを使用してインスタンスにログインできます。

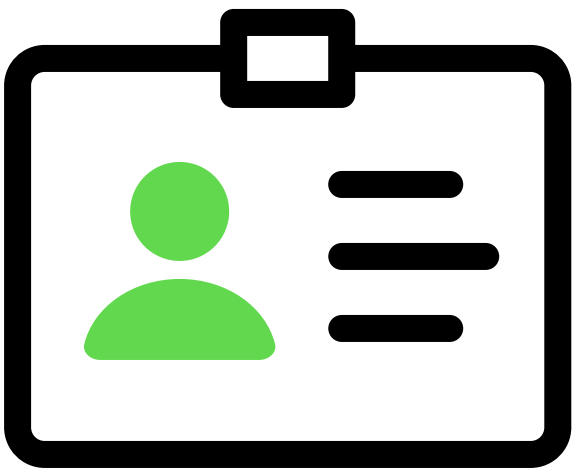
証明書ベースの認証

ユーザー名とパスワードの代わりにユーザーにマップされた一意の PEM エンコード証明書を使用した証明書ベースの認証。



自己登録

外部ユーザーの自己登録を使用して、大量の外部ユーザーをインスタンスにオンボーディングします。



LDAP

自動翻訳

ライトウェイトディレクトリアクセスプロトコル (LDAP) ディレクトリと統合して、ユーザーログインプロセスを簡素化します。

ユーザーの認証には、数種類の異なる方法を使用できます。ユーザーの認証情報は、各方法に対して保存された各種認証情報と照合されます。

注:

- Okta SSO プラグインは廃止されました。
- 認証処理に影響するセキュリティプロパティの詳細については、「インスタンスセキュリティ強化設定」の アクセス制御 を参照してください。
- Multiple Provider SSO アプリケーションを介して SAML および Digest Authentication を使用できます。

適応認証

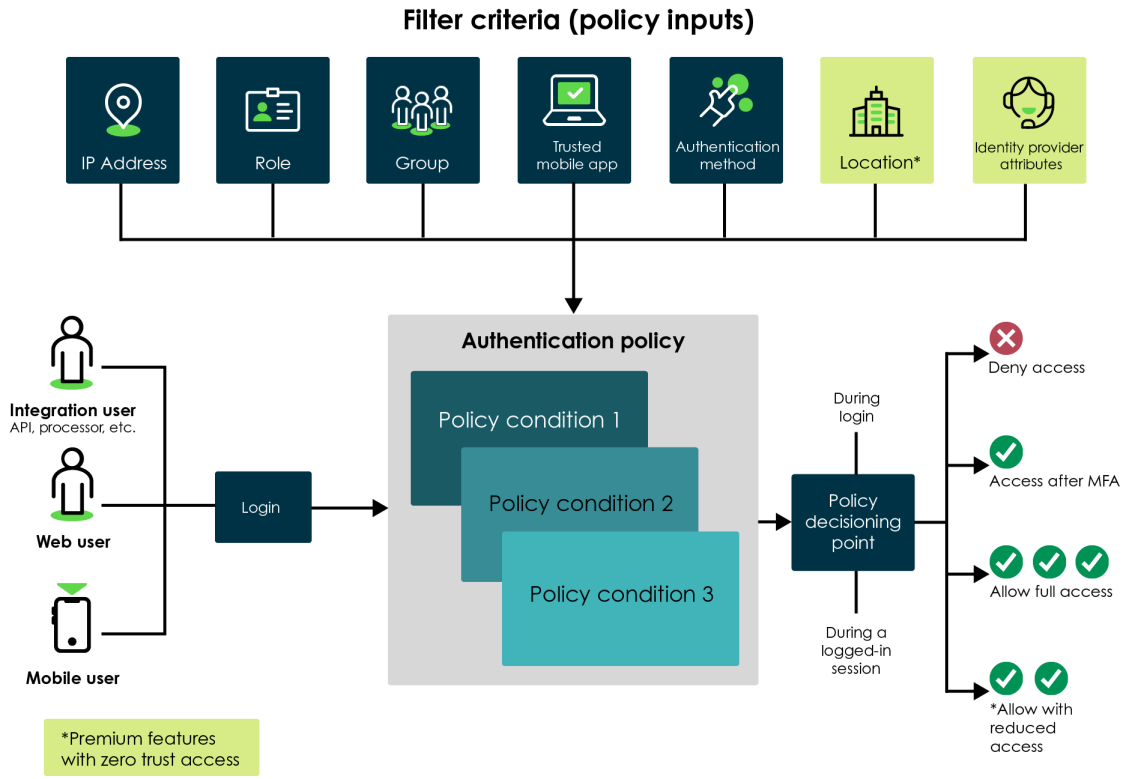
適応認証ポリシーフレームワークを使用すると、適切なタイミングで適切なユーザーにコンテキスト認証コントロールを適用できます。適応認証では、認証ポリシーを使用して認証要求を評価してから、指定されたポリシー条件に基づいてインスタンスへのアクセスを拒否または許可します。

適応認証ポリシーとコンテキストを使用して、IP アドレス、ユーザーロール、ユーザーグループなどの基準に基づいて、ユーザーおよび API のインスタンスへのアクセスを制限します。セキュリティ要件に従って、ビルトイン認証ポリシーを設定できます。

たとえば、アドミニストレーターは [*Allow Access Policy*] を使用して、信頼できる IP アドレス範囲内にあり、特定のロールのメンバーであるユーザーからのログインのみを許可するように設定できます。*Post-authentication context*に割り当てると、アクセスポリシーは信頼できない IP アドレスからのアクセスを拒否します。

インスタンスの言語でカスタムメッセージを設定するには、sys_ui_message.list にキーと値のペアを追加し、sys_ui_message レコードを更新する必要があります。誤ったパスワードでログインすると、優先言語のカスタムメッセージが表示されます。

自動翻訳



適応認証コンポーネント

認証ポリシー

認証ポリシーは、指定されたポリシー条件に基づいて認証要求を評価し、ポリシー条件評価の出力に応じてアクセスを許可または拒否します。たとえば、[アクセス許可ポリシー] で指定されたすべてのポリシー条件が true と評価された場合にのみ、アクセスが許可されます。

認証ポリシーは、フィルター基準によって提供される情報を使用してポリシーの条件と比較し、インスタンスへのアクセスを許可するかどうかを決定します。たとえば、フィルター基準はユーザーの IP アドレスを提供し、ポリシー条件はアクセスを許可する前にこのアドレスが特定の範囲内かどうかを判断します。認証ポリシーの詳細については、「認証ポリシー」を参照してください。

認証ポリシーのコンテキスト

認証ポリシーのコンテキストは、ログインプロセス中にポリシーを強制適用する方法とタイミングを定義します。事前認証コンテキストは、ユーザーにログイン画面が表示される前に実行されます。認証後コンテキストは、ユーザーが認証情報を入力した後に実行されます。ポリシーを使用するには、ポリシーコンテキストにアサインする必要があります。これらのコンテキストの詳細については、「認証ポリシーのコンテキスト」を参照してください。

フィルター基準

フィルター基準 (ポリシー入力とも呼ばれます) は、ポリシー条件の入力として使用されます。ポリシー条件では、これらの入力を使用して認証要求の要件を検証して満たします。これらの入力は、ユーザーロール、IP 範囲、ID プロバイダーなどの情報を提供します。フィルター基準の詳細については、「フィルター基準」を参照してください。

認証プロパティ

認証プロパティを使用して、インスタンスで適応認証を有効にするかどうかを制御します。プロパティを使用してデバッグを有効にし、アクセスがブロックされたときにユーザーに表示されるメッセージを定義することもできます。これらのプロパティの詳細については、「適応認証プロパティの設定」を参照してください。

REST API アクセスポリシー

適応認証フレームワークのフィルター基準を使用して、受信した ServiceNow REST API へのアクセスを制限できます。詳細については、「REST API アクセスポリシー」を参照してください。

ドメインセパレーションと適応認証

適応認証は、認証ポリシー条件レベルでドメインセパレーションされたインスタンスでサポートされています。ポリシー条件は、レコードの ドメイン [sys_domain] フィールドのドメインに影響します。グローバルドメインのポリシー条件は、すべてのドメインに影響します。

適応認証イベント

適応認証イベントテーブルを使用して、適応認証機能に固有のイベントについて知ることができます。詳細については、「適応認証イベント」を参照してください。

適応認証のアクティブ化

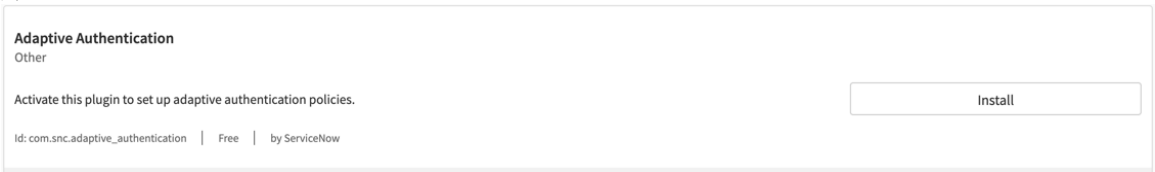
admin ロールを持っている場合は、適応認証の適応認証プラグイン (com.snc.adaptive_authentication) をアクティブ化できます。

始める前に
必要なロール：admin。

このタスクについて

手順

1. 移動先 すべて > システムアプリケーション > 利用可能なすべてのアプリケーション > すべて.
2. フィルター基準と検索バーを使用して適応認証 (com.snc.adaptive_authentication) プラグインを検索します。



名前または ID でプラグインを検索できます。プラグインが見つからない場合は、ServiceNow 担当者から要求する必要があります。

3. [インストール] を選択して、インストールプロセスを開始します。

> 注: ドメインセパレーションと代理アドミンがインスタンスで有効になっている場合、管理ユーザーはグローバルドメインに含まれている必要があります。それ以外の場合、次のエラーが表示されます： 「別の操作が実行されているため、アプリケーションのインストールは利用できません： <プラグイン名> のプラグインの有効化 (Application installation is unavailable because another operation is running: Plugin Activation for <plugin name>)」

インストールが完了するとメッセージが表示されます。プラグインとともにインストールされるコンポーネントの詳細については、「 アプリケーションとともにインストールされているコンポーネントの検索 」を参照してください。

次のタスク
認証ポリシーを構成して、インスタンスにコンテキスト認証制御を適用します。

ポリシーを設定した後、[認証ポリシーの有効化 **(Enable Authentication Policy)]** ポリシーを使用して適応認証を有効にします。適応認証プロパティの詳細については、「適応認証プロパティの設定」を参照してください。

フィルター基準

フィルター基準 (ポリシー入力とも呼ばれます) は、ポリシー条件の入力として、認証要求の要件を検証し、その要件を満たすために使用されます。

フィルター基準を使用して、ユーザーの IP アドレス、ロール、グループなどの情報認証ポリシーを指定します。ポリシーの [ポリシー条件] セクションにこれらの基準を追加します。

適応認証で使用されるフィルター基準は 7 種類あります。認証ポリシーでは、これらの基準を 1 つ以上使用して認証要求を評価できます。

注: 場所フィルターと ID プロバイダーフィルターは、Zero Trust アクセス機能で使用できます。詳細については、「ゼロトラストアクセス (ZTA)」を参照してください。

フィルター基準タイプ

タイプ	説明
IP フィルター基準	IP フィルター基準を使用して、ユーザーの IP アドレスに基づいてユーザーをフィルタリングします。IPv4 と IPv6 の両方がサポートされています。
ロールフィルター基準	ロールフィルター基準を使用して、ユーザーのロールに基づいてユーザーをフィルタリングします。
グループフィルター基準	グループフィルター基準を使用して、ユーザーが属するユーザーグループに基づいてユーザーをフィルタリングします。
場所のフィルター基準	場所のフィルター基準を使用して、ユーザーの場所に基づいてユーザーをフィルタリングします。
ID プロバイダー属性のフィルター基準	IdP からの SAML 応答から受信した ID プロバイダー属性を認証のフィルター基準として使用します。

汎用基準

前述のタイプに加えて、4 種類の汎用的なフィルター基準があります。これらの基準はフィルターナビゲーターには表示されませんが、認証ポリシーにポリシー入力を追加するときに選択できます。

汎用的なフィルター基準タイプ

タイプ	説明
認証スキーム	ユーザーの認証スキームに基づいてフィルタリングするために使用します。この基準は、次の 2 つのオプションを持つ選択肢タイプです。 • ローカルログインを示すユーザー名とパスワード☒ • マルチ SSO (SAML、OIDC、または Digest) ベースのログインを示す SSO。

汎用的なフィルター基準タイプ (続く)

タイプ	説明
	注: **Integration - Multiple Provider Single Sign-On Installer** [com.snc.integration.sso.multi.installer] プラグインがインストールされている場合のみフィルター基準を使用できます。
Identity Provider	ユーザーの ID プロバイダーに基づいてフィルタリングするために使用します。認証スキーム基準とともに使用して、ログインプロセスを細やかに制御します。この基準は、**ID** プロバイダー [sso_properties] テーブルへの参照です。 注: **Integration - Multiple Provider Single Sign-On Installer** [com.snc.integration.sso.multi.installer] プラグインがインストールされている場合のみフィルター基準を使用できます。
ロールベースの MFA	ロールベースの MFA 機能に基づいてフィルタリングするために使用します。この基準は、ユーザーに対してロールベースの MFA が有効かどうかを示すブールタイプのフィルター基準です。☒
ユーザーベースの MFA	ユーザーベースの MFA 機能に基づいてフィルタリングするために使用します。この基準は、ユーザーに対してユーザーベースの MFA が有効かどうかを示すブーリアン型のフィルター基準です。
信頼できるモバイルアプリ	モバイルアプリからのインスタンスへのアクセスを有効にするための信頼できるモバイルアプリフィルター。

IP フィルター

IP フィルター基準を使用して、ユーザーの IP アドレスに基づいてユーザーをフィルタリングします。IPv4 と IPv6 の両方がサポートされています。

IP フィルター基準を使用すると、ユーザーの IP アドレスに基づいてユーザーをフィルタリングできるようになります。特定のアドレスまたはアドレス範囲へのアクセスを許可または拒否するように認証ポリシーを設定できます。

IP フィルター基準の作成

IP フィルター基準を使用すると、ユーザーの IP アドレスに基づいてユーザーをフィルタリングできるようになります。特定のアドレスまたはアドレス範囲へのアクセスを許可または拒否するように認証ポリシーを設定できます。

始める前に

必要なロール：admin

手順

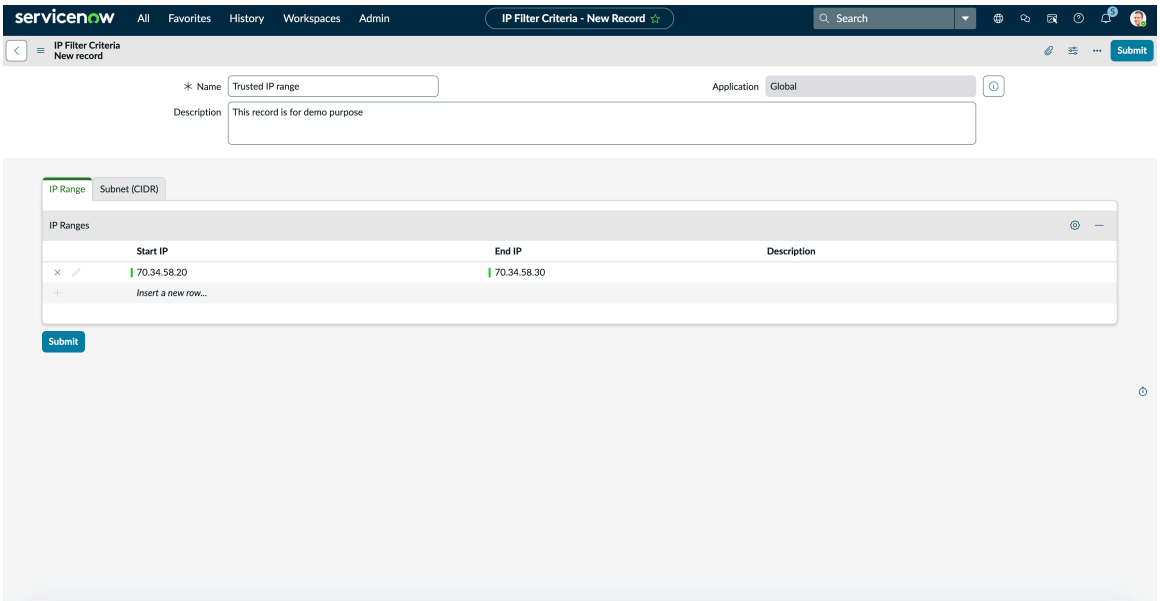
1. 移動先 すべて > 適応認証 > フィルター基準 > **IP** フィルター基準.
2. [**New**] をクリックします。
3. フォーム上で、以下のフィールドに記入します。

[IP フィルター基準] フォーム

フィールド	説明
名前	IP ネットワークを識別する名前

フィールド	説明
説明	IP ネットワークの簡単な説明
アプリケーション	アプリケーションのスコープ

IP フィルター基準レコードの例

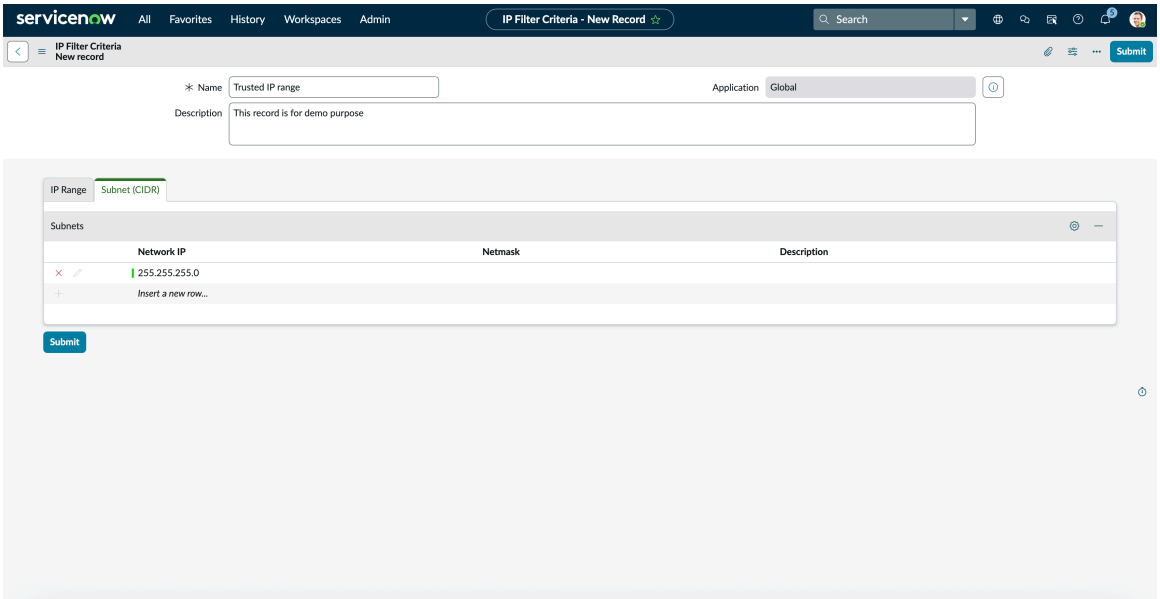


4. フォームヘッダーを右クリックし、[保存] をクリックします。
5. **[IP** 範囲] タブで、[新規行を挿入] をダブルクリックします。
   単一の IP アドレスまたは複数の IP アドレス範囲を入力できます。たとえば、IP アドレスの範囲の場合は、[開始 **IP]** 列に「 192.0.2.0」、[終了 **IP]** 列に「192.0.2.255」と入力します。

   注: 単一の IP アドレスの場合は、[開始 **IP]** 列と [終了 **IP]** 列に同じ IP アドレスを入力する必要があります。[開始 **IP]** 列に IP アドレスを入力する際に、[終了 **IP]** 列を空白にしたままでレコードを保存してください。[終了 **IP]** 列には、[開始 **IP]** と同じアドレスが自動的に入力されます。

6. [サブネット **(CIDR)]** タブで、[新規行を挿入] をダブルクリックします。

クラスレスドメイン間ルーティング (CIDR) 形式でネットワーク IP アドレスとネットマスクを入力します。たとえば、ネットワーク IP として 255.255.255.0、ネットマスクとして 25 を入力します。



ロールフィルター

ロールフィルター基準を使用して、ユーザーのロールに基づいてユーザーをフィルタリングします。

ロールフィルター基準を使用すると、ロールに基づいてユーザーをフィルタリングできるようになります。ユーザーロールのリストへのアクセスを許可または拒否するように認証ポリシーを設定できます。

ロールフィルター基準の作成

ロールフィルター基準を使用すると、ロールに基づいてユーザーをフィルタリングできるようになります。ユーザーロールのリストへのアクセスを許可または拒否するように認証ポリシーを設定できます。

始める前に

必要なロール：admin

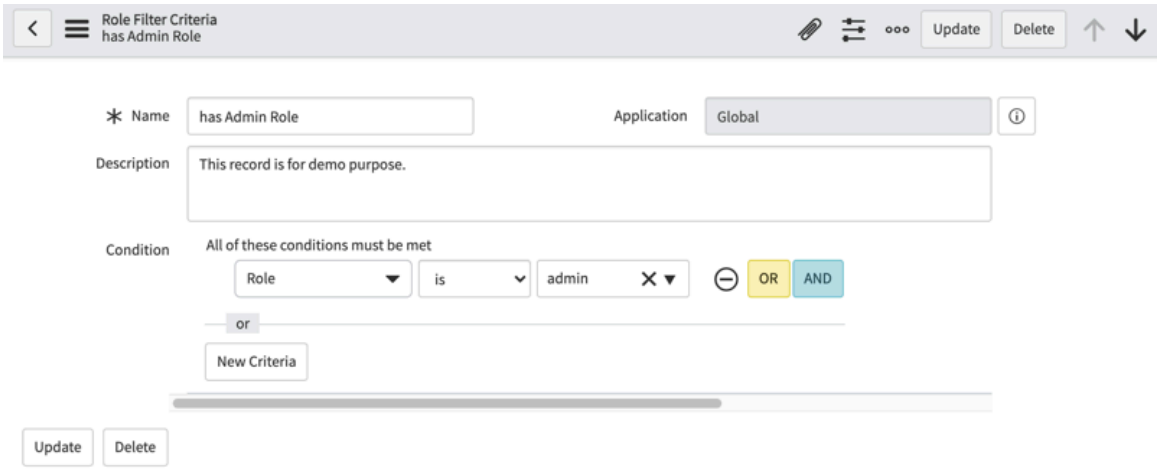
手順

1. 移動先 すべて > 適応認証 > フィルター基準 > ロールフィルター基準.
2. [**New**] をクリックします。
3. フォーム上で、以下のフィールドに記入します。

[ロールフィルター基準] フォーム

フィールド	説明
名前	ロールを識別する名前
アプリケーション	アプリケーションのスコープ
説明	ロールの簡単な説明

ロールフィルター基準レコードの例

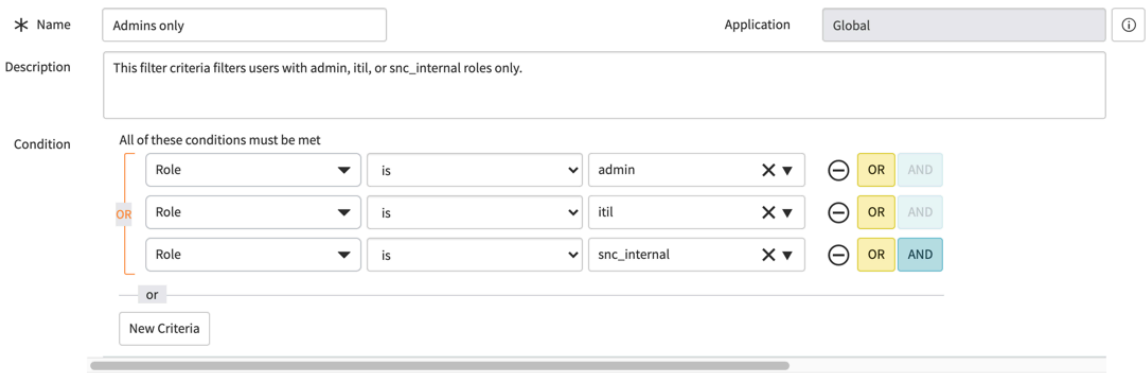


4. [基準のロール] から、[新規行を挿入] をダブルクリックします。
5. 条件ビルダーを使用して特定のロールの条件を作成します。

   たとえば、admin、itil、または snc_internal ロールを持つユーザーのみを許可する条件を作成できます。条件ビルダーの詳細については、「条件ビルダーを使用した条件ステートメントの作成」を参照してください。

   注:

   - 現在、ロールフィルター基準ではドット連結はサポートされていません。
   - ロールフィルター基準では、次の演算子はサポートされていません:
     - 次の値と異なる (!=)
     - 次の値を含まない
     - 次の値と異なる
     - (空) である
     - 次の値と同じ
     - (空) でない
     - 空の文字列である



グループフィルター

グループフィルター基準を使用して、ユーザーが属するユーザーグループに基づいてユーザーをフィルタリングします。

グループフィルター基準を使用して、ユーザーが属するユーザーグループに基づいてユーザーのアクセスを許可または拒否します。

グループフィルター基準の作成

グループフィルター基準を使用して、ユーザーが属するユーザーグループに基づいてユーザーのアクセスを許可または拒否します。

始める前に
必要なロール：admin

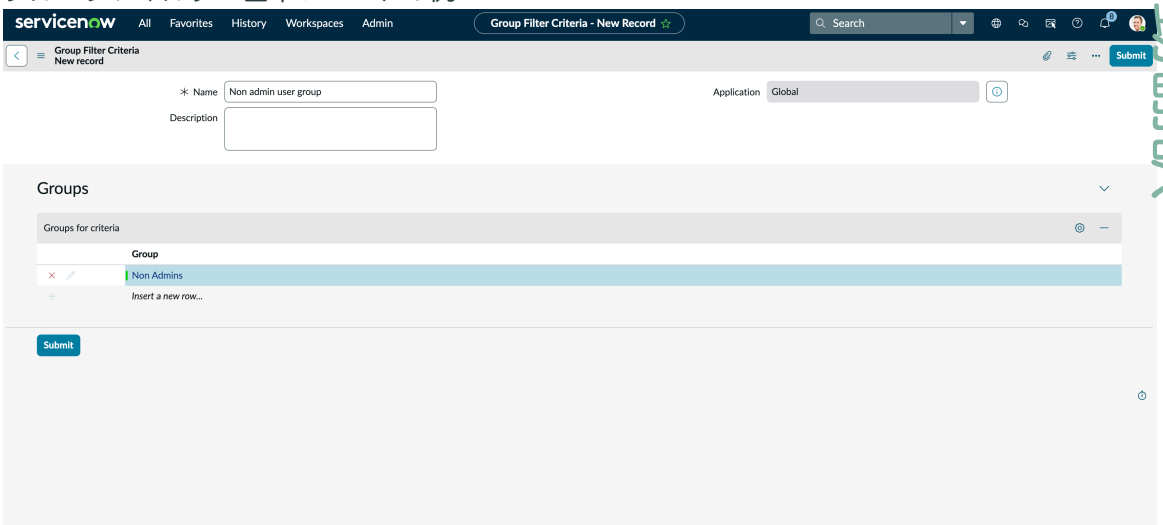
手順

1. 移動先 すべて > 適応認証 > フィルター基準 > グループフィルター条件.
2. [**New**] をクリックします。
3. フォーム上で、以下のフィールドに記入します。

グループフィルター基準

フィールド	説明
名前	グループを識別する名前
説明	グループの簡単な説明
アプリケーション	アプリケーションのスコープ

グループフィルター基準レコードの例



4. [基準のグループ] タブから、[新規行を挿入] をダブルクリックします。
5. 検索アイコン ( ) をクリックして、ユーザーグループを検索して選択します。
6. 保存アイコンをクリックします。

場所フィルター

場所フィルター基準を、ユーザーの場所に基づいたユーザーのフィルター入力として使用できます。

場所フィルターは、アドミニストレーターがデバイスの物理的な場所に基づいて認証ポリシーを作成するときに使用できるフィルター基準です。

注:

- 場所フィルターは、Zero Trust アクセス機能で使用できます。詳細については、「ゼロトラストアクセス (ZTA)」を参照してください。
- このインスタンスは ADCv2 上にある必要があります。インスタンスが ADCv2 上にない場合、ユーザーの場所情報は利用できません。

場所フィルター基準を使用して、次のことを実行できます。

- 認証の要件を検証して満たすために、ポリシー条件に対するポリシー入力として機能します。
- [国] に基づいて適応認証ポリシーを作成する機能を提供します。
- 指定された地域へのインスタンスアクセスを許可するかどうかを指定します。
- 位置情報に基づく事前認証または事後認証ポリシーを使用して、次のことを行います。
  - 会社の判断により、プライバシーが厳格に制御されている地域以外の制裁対象国からアクセスを防止します。
  - 該当するプライバシー対象地域と会社の判断による範囲内でのみ、エリアへのアクセスを許可します。
  - 認証のための国ベースの許可リストを構成します。

ユースケース

適応認証に場所フィルター基準を使用するユースケースの一部を次に示します。

- 国からのインスタンスへのアクセスをブロックします。
- 特定の国からのインスタンスへのアクセスのみを許可します。
- 国に基づいてログインするためのステップアップ認証または MFA を強制します。
- 国に基づいてユーザーのロールを削減または制限します。
- 場所フィルター基準は、MFA、Zero Trust Access (ZTA)、認証前コンテキスト、および認証後コンテキストに使用できます。

場所の識別

ユーザーの場所を識別するための位置情報サービスは、サードパーティサービス MaxMind によって提供されます。

ユーザーの場所は、VPN を介して x-forwarded-for ヘッダーから識別されます。サービスによってヘッダーが入力されていない場合は、VPN IP (場所) のみがユーザーの場所として表示されます。

注: 場所フィルターの構成後に誤った場所が表示される場合は、KB 記事 を参照してトラブルシューティングを行います。

場所ベースのアクセスのアクティブ化

アドミニストレーターがユーザーの場所に基づいて適応認証ポリシーを設定できるようにするには、**[Zero Trust - Location Based Access (Zero Trust - Location Based Access)]** (com.snc.zero_trust_location_access) を有効にします。

始める前に
必要なロール：admin

- 依存プラグイン：Adaptive authentication
- プラグインタイプ：有料でライセンスが必要です。
- このインスタンスは ADCv2 上にある必要があります。インスタンスが ADCv2 上にない場合、ユーザーの場所情報は利用できません。

手順

1. 移動先 すべて > システムアプリケーション > 利用可能なすべてのアプリケーション > すべて.
2. フィルター基準と検索バーを使用して、**Zero Trust - Location Based Access** (com.snc.zero_trust_location_access) プラグインを検索します。

   名前または ID でプラグインを検索できます。プラグインが見つからない場合は、ServiceNow 担当者から要求する必要があります。
3. [インストール] を選択して、インストールプロセスを開始します。

   注: ドメインセパレーションと代理アドミンがインスタンスで有効になっている場合、管理ユーザーはグローバルドメインに含まれている必要があります。それ以外の場合、次のエラーが表示されます： 「別の操作が実行されているため、アプリケーションのインストールは利用できません： <プラグイン名> のプラグインの有効化 (Application installation is unavailable because another operation is running: Plugin Activation for <plugin name>)」

   インストールが完了するとメッセージが表示されます。プラグインとともにインストールされるコンポーネントの詳細については、「 アプリケーションとともにインストールされているコンポーネントの検索 」を参照してください。

場所のフィルター基準の作成

場所のフィルター基準を使用して、ユーザーの場所に基づいてユーザー認証の入力をフィルタリングします。

始める前に
必要なロール：admin

必要なプラグイン：**Zero Trust - Location Based Access** (com.snc.zero_trust_location_access)

プロパティ： [適応認証] プロパティを有効にします。

注: アドミニストレーターは、場所が現在のユーザーセッションで利用可能な場合にのみ場所フィルターに基づいてポリシーを作成できます。

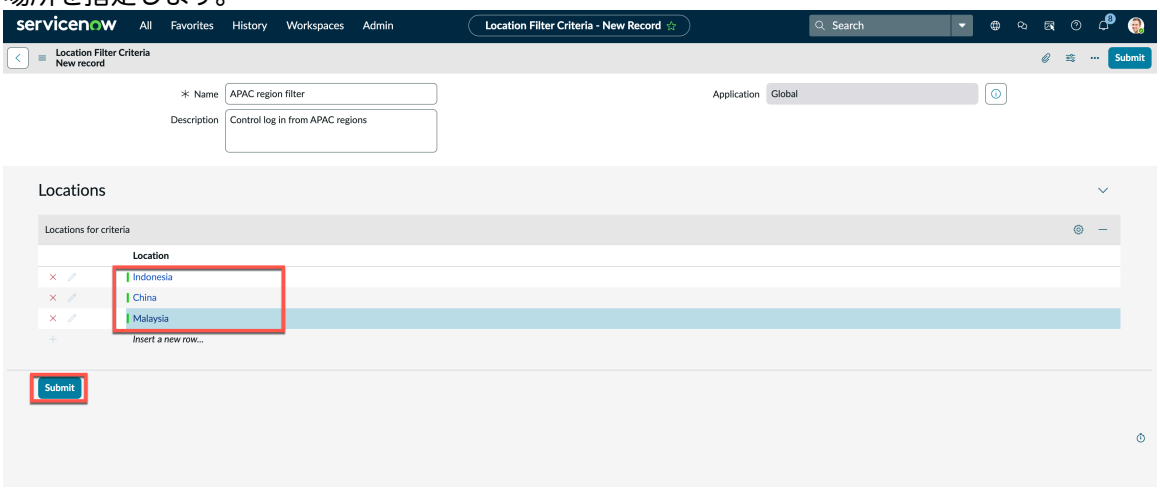
手順

1. 移動先 すべて > 適応認証 > フィルター基準 > 場所フィルター基準.
2. **[New** (新規)] を選択します。
3. フォーム上で、以下のフィールドに記入します。

場所のフィルター基準フォーム

フィールド	説明
名前	基準を識別する名前
説明	基準の簡単な説明
アプリケーション	アプリケーションのスコープ

4. [場所] セクションの [基準の場所 (Locations for criteria)] タブで、[新しい行を挿入] をダブルクリックします。
5. 場所を指定します。



6. [送信] を選択します。

チュートリアル：場所のフィルター基準を使用する

認証ポリシーで場所フィルター基準を使用し、場所に基づいてユーザーへのアクセスを制限する手順について説明します。

始める前に

必要なロール：admin

必要なプラグイン：**Zero Trust - Location Based Access** (com.snc.zero_trust_location_access)

プロパティ： [適応認証] プロパティを有効にします。

> **注:** アドミニストレーターは、場所が現在のユーザーセッションで利用可能な場合にのみ場所フィルターに基づいてポリシーを作成できます。

次の手順では、場所のフィルター基準を作成して認証ポリシーで使用する方法について説明します。

手順

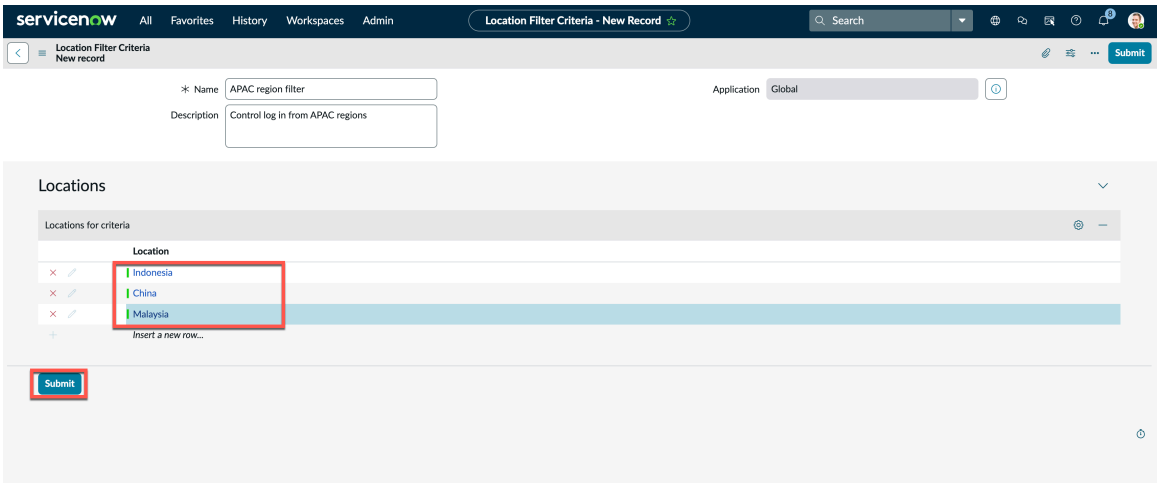
1. 移動先 すべて > 適応認証 > フィルター基準 > 場所フィルター.
2. **[New** (新規)] を選択します。
3. フォーム上で、以下のフィールドに記入します。

場所のフィルター基準フォーム

フィールド	説明
名前	基準を識別する名前
説明	基準の簡単な説明
アプリケーション	アプリケーションのスコープ

4. [場所] セクションの [基準の場所 (Locations for criteria)] タブで、[新しい行を挿入] をダブルクリックします。
5. 場所を指定します。

たとえば、一部の APAC 地域を使用して APAC からのユーザーのログインを制御します。



例で設定された基準に基づいて、インスタンスの Indonesia、China、および Malaysia からのログインを制御できます。

6. [送信] を選択します。
7. 任意の認証コンテキスト (事前、事後、MFA) およびセッションアクセスで作成されたフィルター基準を使用します。

   認証コンテキストとセッションアクセスに基づく構成の詳細については、以下を参照してください。

   - 事前承認コンテキストでの場所のフィルター
   - 認証後コンテキストでの場所のフィルター
   - MFA コンテキストでの場所のフィルター
   - セッションアクセスでの場所のフィルター

   認証ポリシーの失敗が原因でログインが失敗したときにユーザーに表示されるプロパティ ID - エラーメッセージ (glide.auth.policy.ui.error.message) を使用して、エラーメッセージをカスタマイズできます。

事前認証コンテキストで場所のフィルターを使用する

事前認証コンテキストで作成された場所のフィルター基準を使用します。

始める前に

必要なロール：admin

必要なプラグイン：**Zero Trust - Location Based Access** (com.snc.zero_trust_location_access)

場所に基づいてユーザーへのアクセスを制限する国で、場所のフィルターを作成します。詳細については、「場所のフィルター基準の作成」を参照してください。

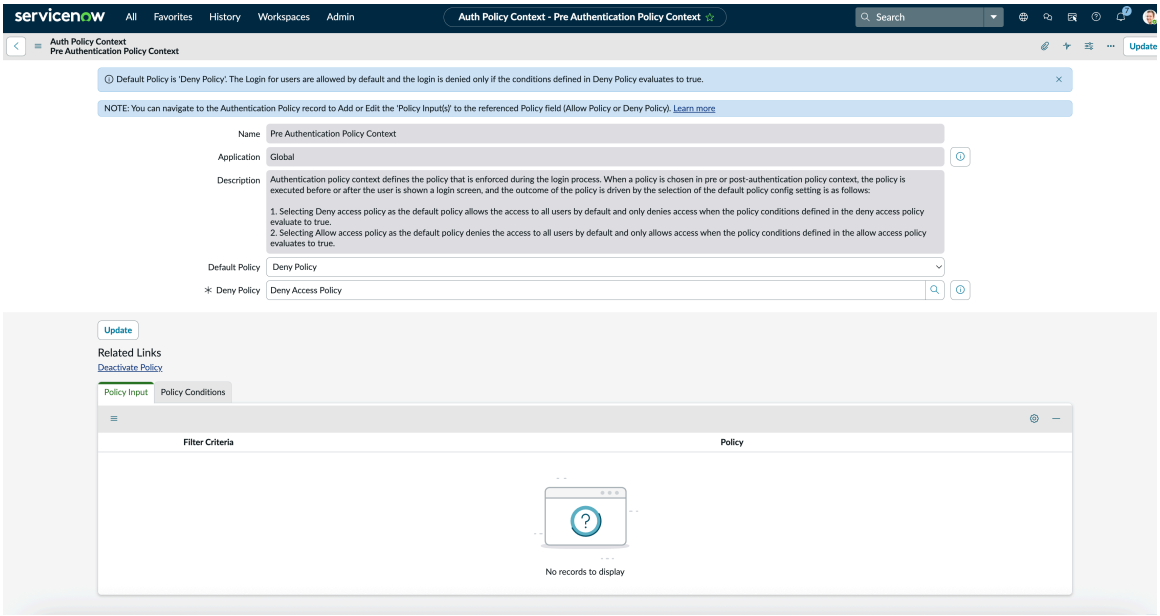
手順

1. 移動先 すべて > 適応認証 > 認証ポリシーのコンテキスト > 事前認証のコンテキスト.

   ポリシーが事前認証ポリシーコンテキストで選択された場合：

- [アクセス拒否ポリシー (Deny access policy)] をデフォルトポリシーとして選択すると、デフォルトですべてのユーザーへのアクセスが許可され、アクセス拒否ポリシーで定義されたポリシー条件が true と評価された場合にのみアクセスが拒否されます。
- [アクセス許可ポリシー (Allow access policy)] をデフォルトポリシーとして選択すると、デフォルトですべてのユーザーへのアクセスが拒否され、アクセス許可ポリシーで定義されたポリシー条件が true と評価された場合にのみアクセスが許可されます。

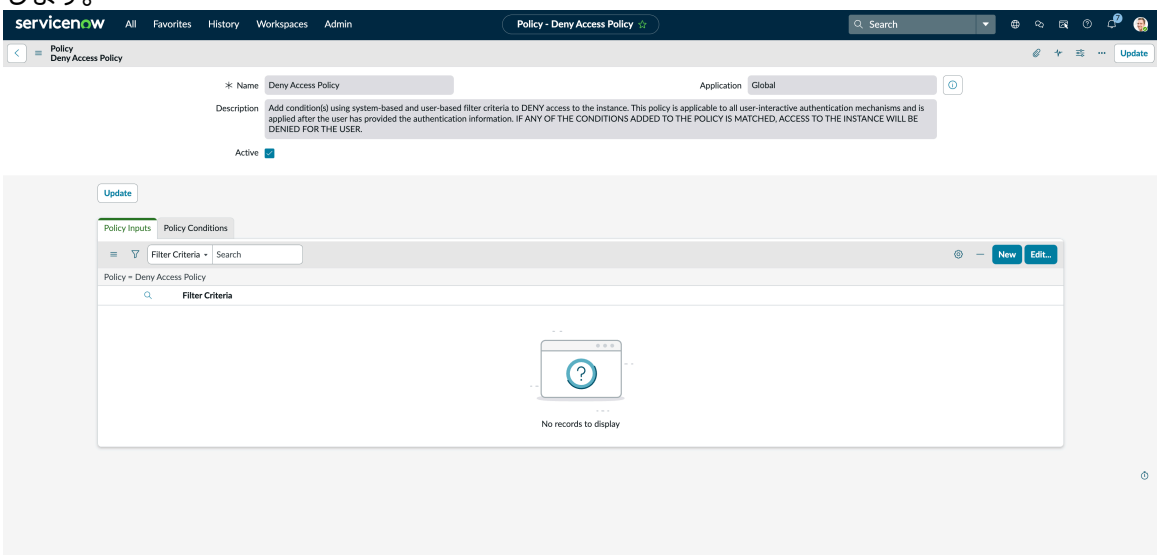
この例は、指定された場所からのログインを制限する方法を示しています。事前認証ポリシーとして [アクセスを拒否 (Deny Access)] および関連するポリシー ([アクセスを拒否 (Deny Access)]) を選択して、ポリシーの入力と条件を指定できます。



2. 情報アイコンを選択し、[レコードを開く] を選択して 拒否ポリシー レコードを開きます。

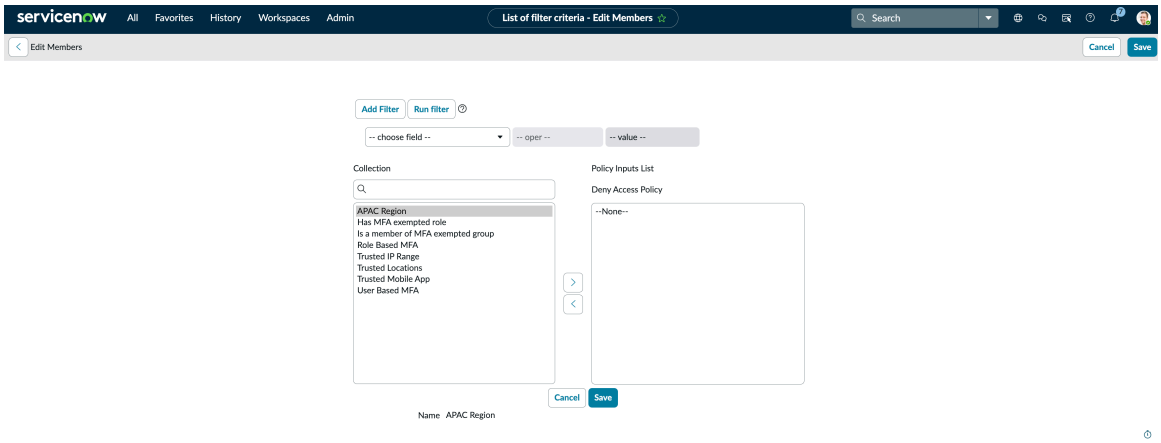
> **注:** このタスクで説明する例は [拒否ポリシー] です。[許可ポリシー] を使用して条件を適宜設定し、ログインを制御することもできます。

3. [アクセス拒否ポリシー (Deny access policy)] の [ポリシーの入力] セクションで、[新規] を選択します。



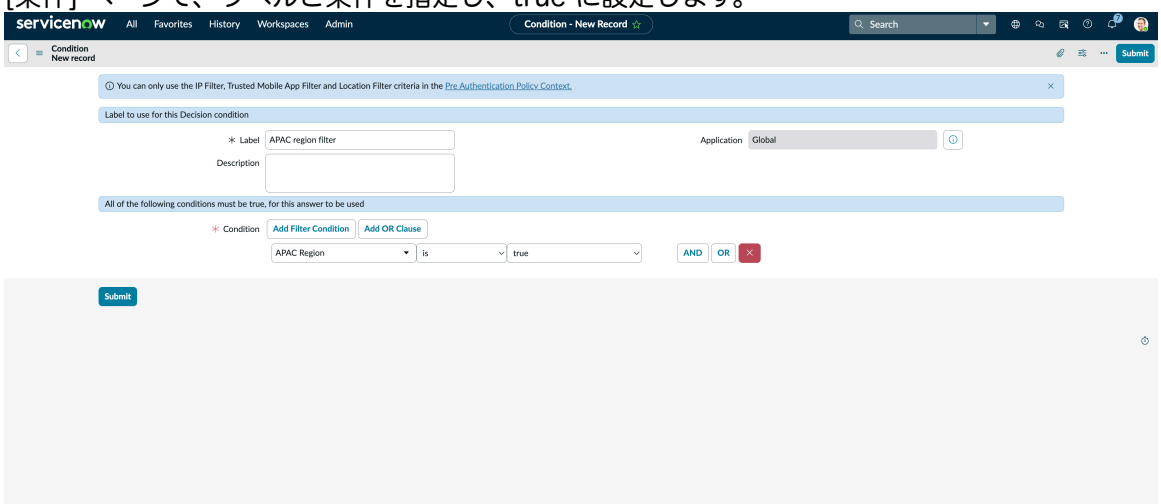
4. 場所のフィルター入力を追加して、[保存] を選択します。

例：APAC 地域



フィルターは [ポリシーの入力] として追加されます。

5. [ポリシー条件] タブを選択し、[新規] を選択します。
6. [条件] ページで、ラベルと条件を指定し、true に設定します。



注:

- この例では、条件として true を選択すると、設定された地域からログインしているユーザーがインスタンスにログインできなくなることを示しています。
- 条件が false に設定されている場合、設定された地域からのユーザーのみがインスタンスにログインでき、他のユーザーはインスタンスにログインできません。

7. [送信] を選択します。
インスタンスリンクを選択し、設定された国からログインしているユーザーには、アクセス拒否に関するエラーメッセージ (ポリシーのプロパティページでアドミニストレーターが設定したエラーメッセージ) が表示されます。

認証後コンテキストでの場所フィルターの使用

認証後コンテキストで作成された場所のフィルター基準を使用します。

始める前に
必要なロール：admin

必要なプラグイン：**Zero Trust - Location Based Access** (com.snc.zero_trust_location_access)

場所に基づいてユーザーへのアクセスを制限する国で、場所のフィルターを作成します。詳細については、「場所のフィルター基準の作成」を参照してください。

手順

1. 移動先 すべて > 適応認証 > 認証ポリシーのコンテキスト > 認証後のコンテキスト.

   ポリシーが認証後ポリシーコンテキストで選択された場合：

   - [アクセス拒否ポリシー (Deny access policy)] をデフォルトポリシーとして選択すると、デフォルトですべてのユーザーへのアクセスが許可され、アクセス拒否ポリシーで定義されたポリシー条件が true と評価された場合にのみアクセスが拒否されます。
   - [アクセス許可ポリシー (Allow access policy)] をデフォルトポリシーとして選択すると、デフォルトですべてのユーザーへのアクセスが拒否され、アクセス許可ポリシーで定義されたポリシー条件が true と評価された場合にのみアクセスが許可されます。

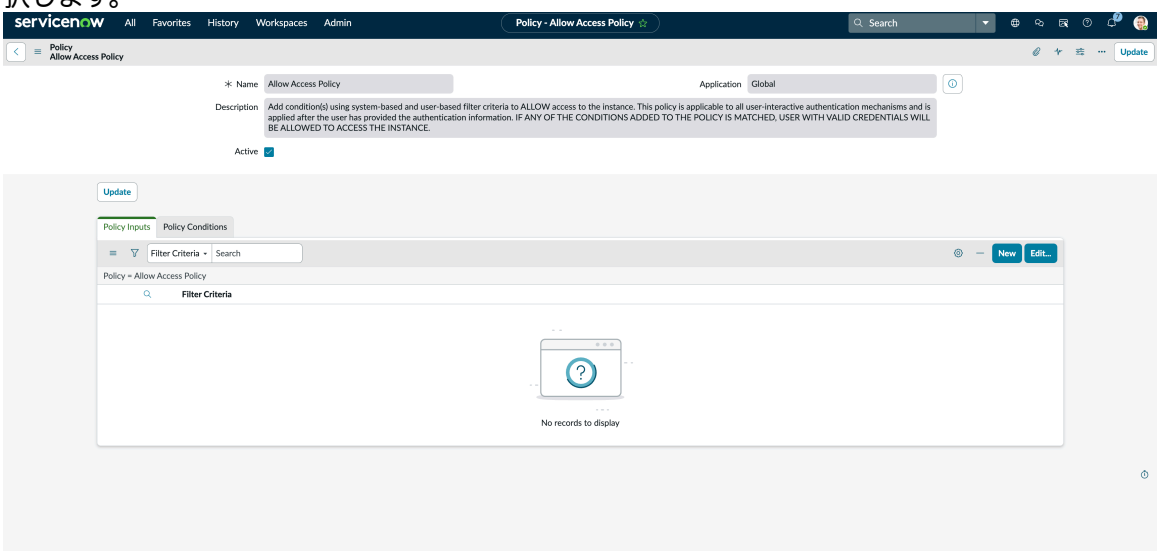
   この例は、指定された場所 (米国) からのみインスタンスにアクセスするという条件を使用して itil ユーザーを構成する方法を示しています。また、itil ユーザーは他の国からログインできません。

   ユーザーがログインページで認証情報を入力した後、認証後ポリシーとして [アクセスを許可 (Allow Access)] および関連するポリシー ([アクセスを許可 (Allow Access)]) を選択して、ポリシーの入力と条件を指定できます。

2. 情報アイコンを選択し、[レコードを開く] を選択して [許可ポリシー] レコードを開きます。

   > **注:** このタスクで説明する例は [許可ポリシー] です。拒否ポリシーを使用し、それに応じて条件を設定してログインを制御することもできます。

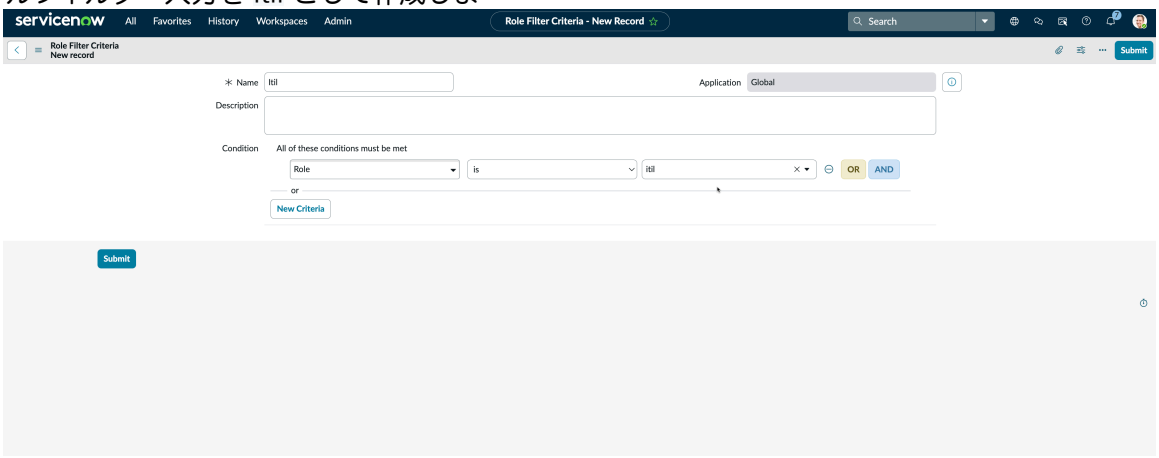
3. [アクセス許可ポリシー (Allow Access Policy)] の [ポリシーの入力] セクションで、[新規] を選択します。



4. ロールフィルター基準と場所フィルター基準を追加します。

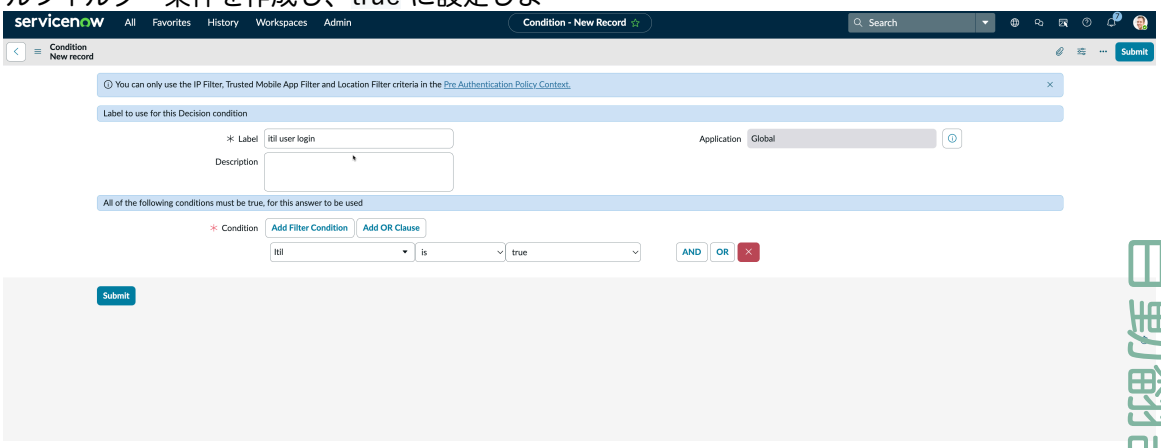
a. ロールフィルター基準を追加する場合：

- ロールフィルター入力を itil として作成しま



す。

- ロールフィルター条件を作成し、true に設定しま

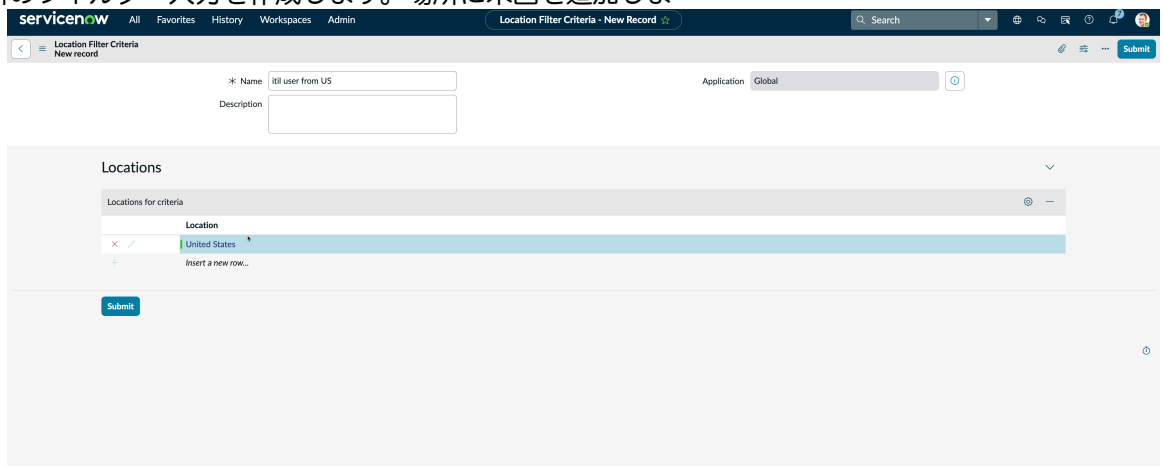


す。

ロールフィルター基準を作成する方法の詳細については、「ロールフィルター基準の作成」を参照してください。

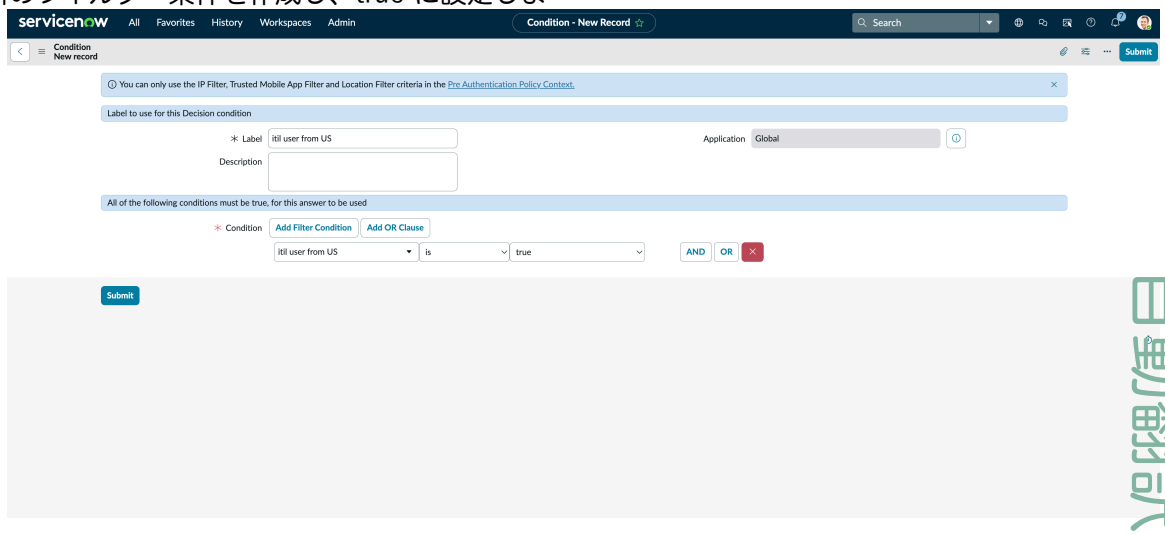
b. 場所のフィルター基準を追加する場合：

- 場所のフィルター入力を作成します。場所に米国を追加しま



す。

- 場所のフィルター条件を作成し、true に設定しま

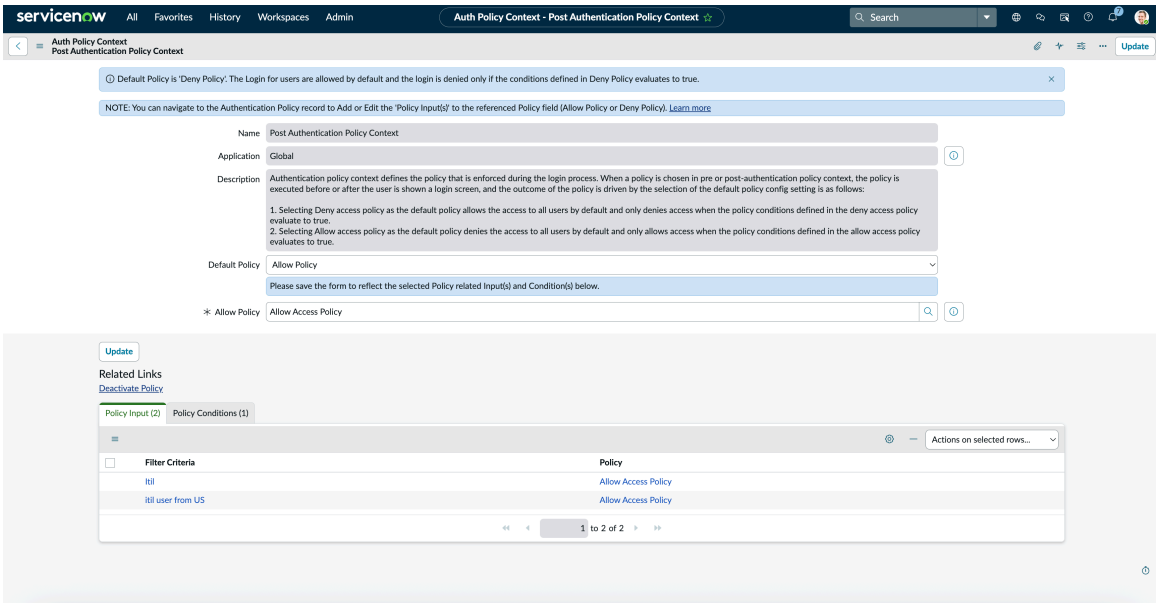


す。

ロールフィルター基準を作成する方法の詳細については、「場所のフィルター基準の作成」を参照してください。

[アクセス許可ポリシー (Allow Access Policy)] には、前のステップで作成されたポリシーの入力と条件が表示されます。

- ポリシーの入力：itil、米国の itil ユーザー
- ポリシー条件：itil ユーザーログイン、米国からの itil ユーザー条件



注:

- この例では、条件として true を選択すると、設定された国 (米国) からログインしている itil ユーザーがインスタンスにログインできることを示しています。
- 条件が false に設定されている場合、設定された国 (米国) からの itil ユーザーはインスタンスにログインできず、他のユーザーもインスタンスにログインできません。

5. [送信] または [更新] を選択して、認証後のコンテキストを更新します。
インスタンスリンクを選択し、認証情報を指定してから設定された国 (米国) 以外からログインしている itil ユーザーには、アクセス拒否に関するエラーメッセージ (ポリシーのプロパティページでアドミニストレーターが設定したエラーメッセージ) が表示されます。

MFA コンテキストで場所のフィルターを使用する

MFA コンテキストで作成された場所のフィルター基準を使用します。

始める前に
必要なロール：admin

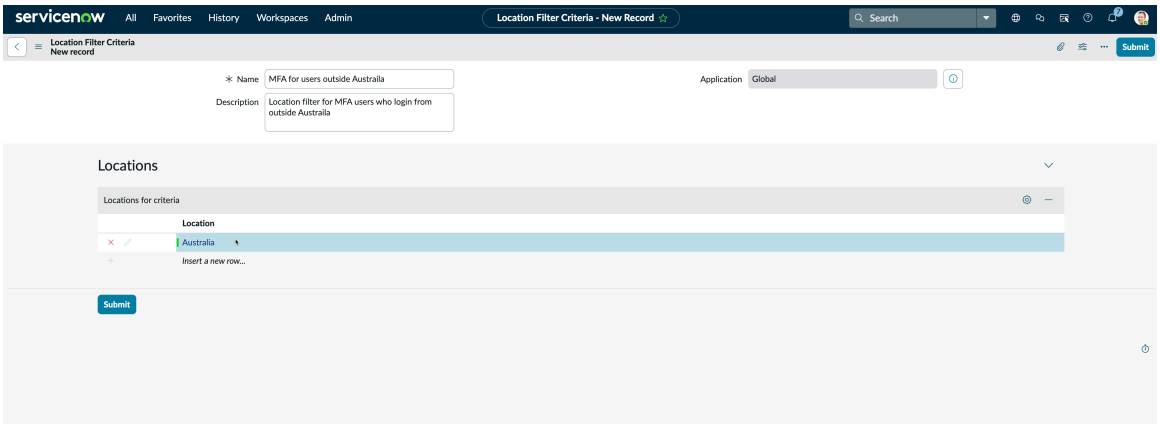
必要なプラグイン：**Zero Trust - Location Based Access** (com.snc.zero_trust_location_access)

次の手順では、場所に基づいてユーザーを認証する場合に、MFA を認証の第 2 要素として構成する国で場所のフィルターを作成する手順について説明します。

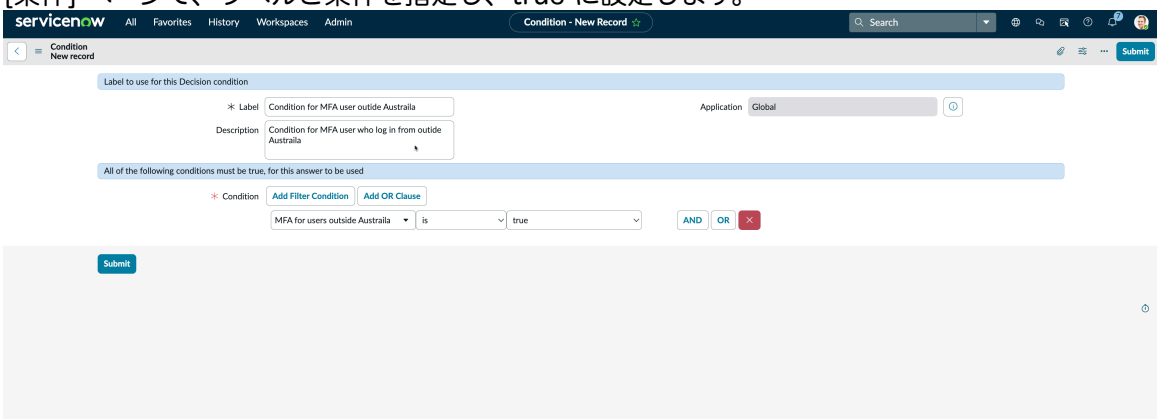
手順

1. 移動先 すべて > 適応認証 > 認証ポリシーのコンテキスト > **MFA** コンテキスト.
2. ステップアップ **MFA** ポリシー情報アイコンを選択し、[レコードを開く] を選択して **MFA** コンテキストを開きます。
3. [ステップアップ MFA ポリシー] ページの [ポリシーの入力] タブで、[新規] を選択します。
4. 場所のフィルター入力を追加して、[送信] を選択します。

   たとえば、オーストラリア国外でインスタンスにログインしているユーザーに MFA を表示するとします。



5. [ステップアップ MFA ポリシー] ページの [ポリシー条件] タブで、[新規] を選択します。
6. [条件] ページで、ラベルと条件を指定し、true に設定します。



7. [送信] を選択します。
8. [ステップアップ MFA ポリシー] ページで、MFA ポリシーが [非アクティブ化済み] の場合はアクティブ化します。
9. [更新] を選択して構成を更新します。

   インスタンスリンクを選択し、認証情報を指定する設定された国 (オーストラリア) 以外のユーザーには、インスタンスにログインするための第 2 要素の認証情報を提供するための MFA 画面が表示されます。

セッションアクセスに対する場所のフィルターの使用

セッションアクセスで作成された場所のフィルター基準を使用して、ユーザーの場所に基づいてロールを削減します。

始める前に

必要なロール：admin

必要なプラグイン：**Zero Trust - Location Based Access** (com.snc.zero_trust_location_access)

次の手順では、場所に基づいてロールを削除または制限する国で場所のフィルターを作成する手順について説明します。

手順

1. 移動先 すべて > **Zero Trust** アクセス > セッションアクセスロール構成.
2. セッションアクセスロールの構成を作成するには、[新規] を選択します。
3. フォームの各フィールドに入力します。

セッションアクセスのロール構成

フィールド	説明
名前	構成の名前
説明	構成の簡単な説明
ポリシー	アクセスポリシーを選択します。ルックアップアイコンを使用して、ポリシーのリストを表示します。 注: ポリシーレコードを開いて、場所のフィルター入力と条件を追加する必要があります。
アクション	[ロールを削除] または [ロールに制限 **(Limit to Roles)**] ○ ロールを削除：構成されたユーザーがログインすると、ロールリストまたはグループリストで提供されたロールのリストがログインセッションに対して削除されます。 ○ ロールに制限 **(Limit to Roles)**：構成されたユーザーがログインすると、選択されたロールのみがユーザーに提供され、他のすべてのロールはログインセッションから削除されます。
ロールリスト	ロールリストからロールを選択します。
グループリスト	グループリストからロールを選択します。

4. [送信] を選択します。

   構成された国に基づいたユーザーのログインは次のとおりです。

   ○ [ロールを削除] の場合、場所のフィルターで構成された国のユーザーがセッションに対して構成されたロールとともに削除されます。

   ○ [ロールに制限 **(Limit to Roles)**] の場合、場所のフィルターで構成された国のユーザーはセッションに対して構成されたロールのみを持ちます。

   セッションに対してロールを削除または制限する方法の詳細については、「チュートリアル：Zero Trust アクセスを使用する」を参照してください。

ID プロバイダー属性フィルター

セキュリティアサーションマークアップ言語 (SAML) 応答から受信した ID プロバイダー属性と、ID プロバイダー (IdP) からの OpenID Connect (OIDC) を認証のフィルター基準として使用します。

SAML および OIDC 応答を介して IdP からすべての属性をフェッチするには、IdP とのテスト接続を実行する必要があります。テスト接続に成功すると、属性が ID プロバイダー構成ページの新しいタブに追加されます。

詳細は、以下のトピックを参照してください。

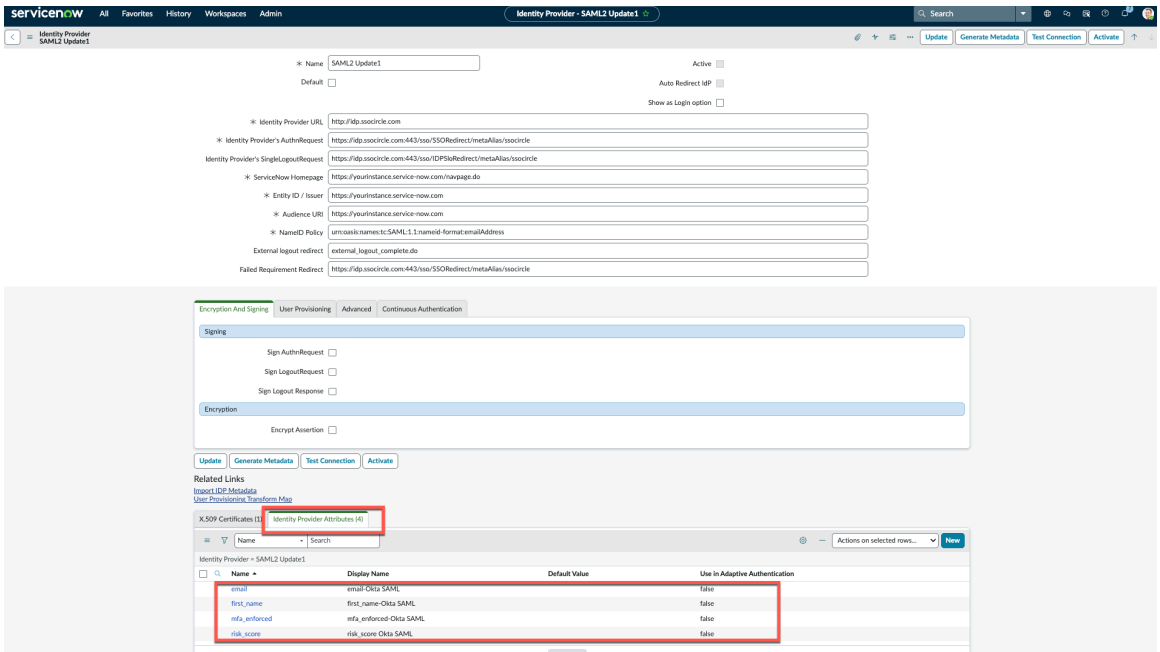
- セキュリティアサーションマークアップ言語の ID プロバイダー属性
- OpenID Connect の ID プロバイダー属性

セキュリティアサーションマークアップ言語の **ID** プロバイダー属性

セキュリティアサーションマークアップ言語 (SAML) 応答から受信した ID プロバイダー属性と、ID プロバイダー (IdP) からの OpenID Connect (OIDC) を認証のフィルター基準として使用します。

SAML 応答を介して IdP からすべての属性をフェッチするには、IdP とのテスト接続を実行する必要があります。テスト接続に成功すると、属性が ID プロバイダー構成ページの新しいタブに追加されます。

- ID プロバイダーフィルターは、Zero Trust アクセス機能で使用できます。詳細については、「ゼロトラストアクセス (ZTA)」を参照してください。
- IdP 属性フィルター基準は、 認証後コンテキスト、 ゼロトラストアクセス (ZTA) セッション降格、および マルチファクター認証コンテキストで使用できます。



ID プロバイダー属性セクションから [新規] を選択して属性を追加し、`true` に設定してそれらの属性を適応認証に使用することもできます。

[ID プロバイダー属性 (Identity Provider Attributes)] が次の詳細とともに表示されます。

場所のフィルター基準フォーム

フィールド	説明
名前	ID プロバイダーによって提供される属性名
表示名	表示名はフィルター基準に使用される詳細な名前です。 **注:** ID プロバイダーによって指定された表示名が長くて判読できないことがあるため、表示名として判読可能な名前を指定できます。

場所のフィルター基準フォーム (続く)

フィールド	説明
デフォルト値	デフォルト値は、SAML 応答に属性がない場合にフィルター基準の評価に使用されます。
適応認証で使用	適応認証で属性を使用するオプション。

注: Azure IdP から入力される属性は、属性の名前の長さにより、名前と表示名が文字数に制限されています。

[ID プロバイダー属性 **(Identity Provider Attributes)]** セクションで [新規] を選択して、新しい属性を追加することもできます。

[適応認証で使用 (Use in Adaptive Authentication)] が true に設定されている場合、選択した属性が汎用的なフィルター基準のフィルター基準として追加されます。たとえば、**risk_score** を true に設定します。汎用的なフィルター基準ページに新しいフィルターが作成されます。

SAML のフィルター基準として **ID** プロバイダー属性を使用

セキュリティアサーションマークアップ言語 (SAML) 応答からの ID プロバイダー (IDP) 属性を認証ポリシーのフィルター基準として使用します。

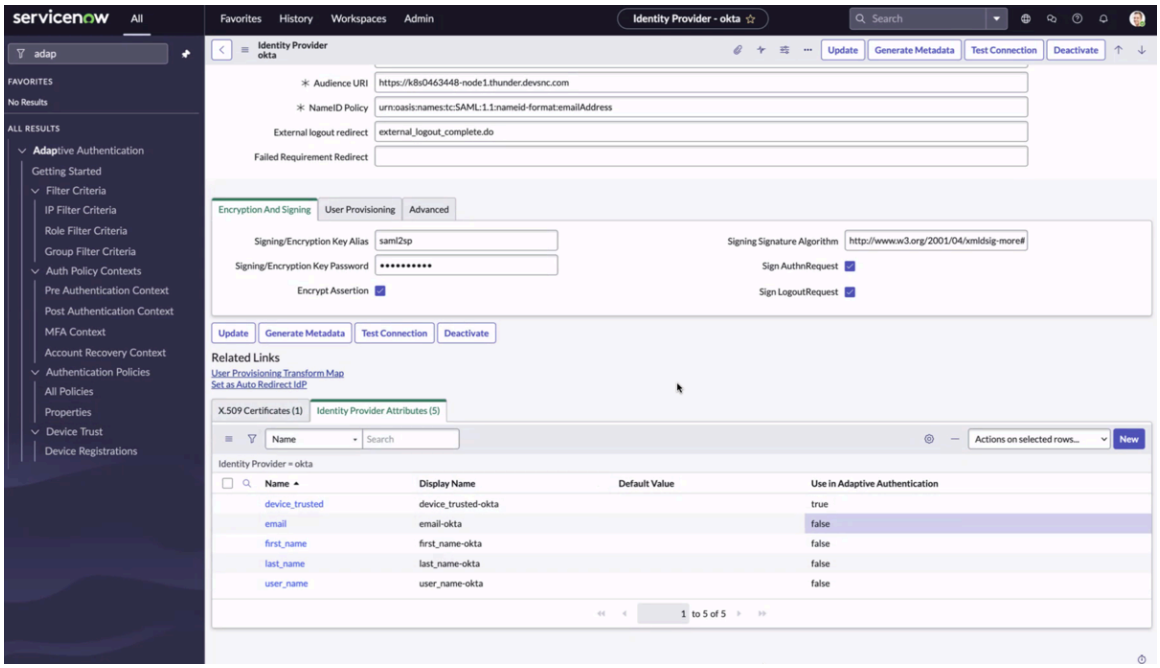
始める前に

必要なロール：admin

ポリシーコンテキスト (事前認証、認証後、マルチファクター認証) とフィルター基準 (ロール、グループ、IP、場所) をポリシーの入力と条件とともに使用して、セッションアクセスポリシーを作成できます。

次の手順は、SAML 応答の IdP 属性をポリシーの入力として設定し、[認証後のコンテキスト]、[マルチファクター認証 **(MFA)** コンテキスト]、および **[Zero Trust - Policy Based Session Access]** で認証を制御する手順を示しています。

Okta IDP 属性は、次のスクリーンショットに表示されているとおりです。[認証後のコンテキスト]、[マルチファクター認証 **(MFA)** コンテキスト]、および **[Zero Trust - Policy Based Session Access]** ポリシーで使用するには、[適応認証で使用 (Use in Adaptive Authentication)] を true に設定する必要があります。



注: 認証後、MFA、 Zero Trust - Policy Based Session Access のポリシーは、ユーザーが認証情報または SSO 応答を入力した後に実行されます。

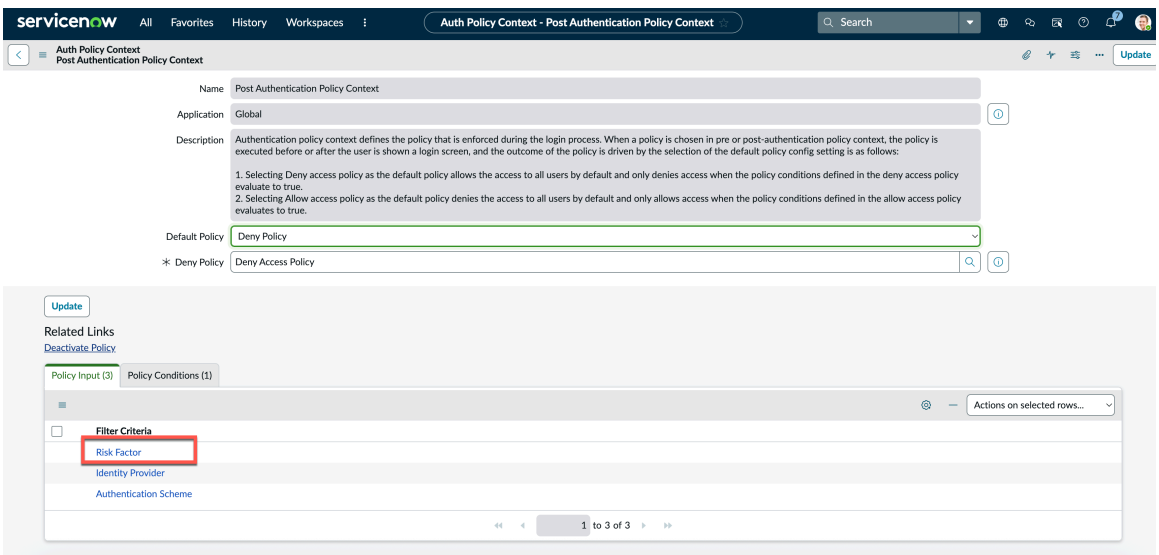
手順

1. 認証後ポリシーのコンテキストでの IDP 属性の使用
   例：デバイスが信頼できる場合に、Okta IDP 属性からのログインを有効にするように設定します。

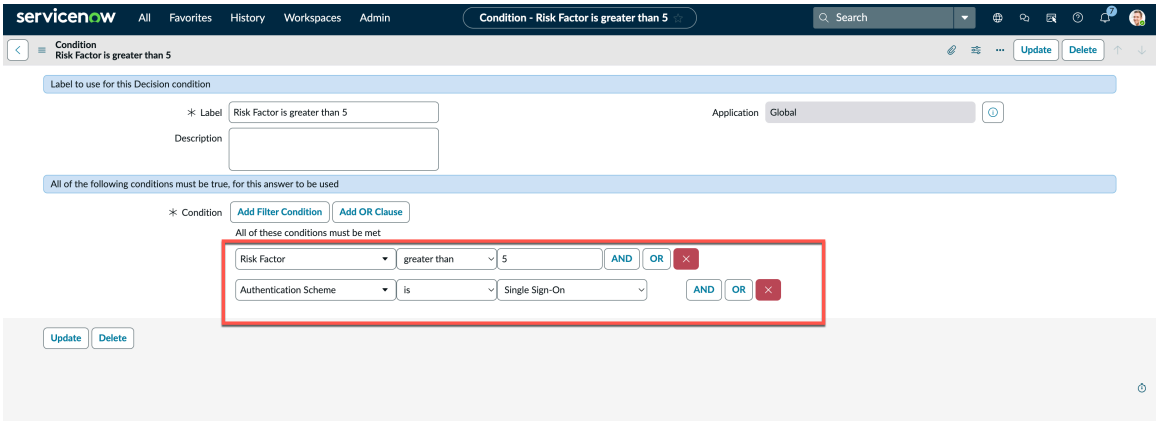
   a. 移動先 すべて > 適応認証 > 認証ポリシーのコンテキスト > 認証後ポリシーのコンテキスト。.

   b. [許可ポリシー] を選択し、ポリシーレコードを開きます。

   c. ポリシー入力で、ポリシーの入力と条件を作成します。
      - ポリシー入力：**device_trusted-okta** を追加します。



- ポリシー条件：**[device_trusted-okta]** は trusted であり、**[ID** プロバイダー] は Okta です。



この構成に基づいて、デバイスが Okta (IdP) から信頼されている場合、ユーザーはインスタンスに対して認証されます。

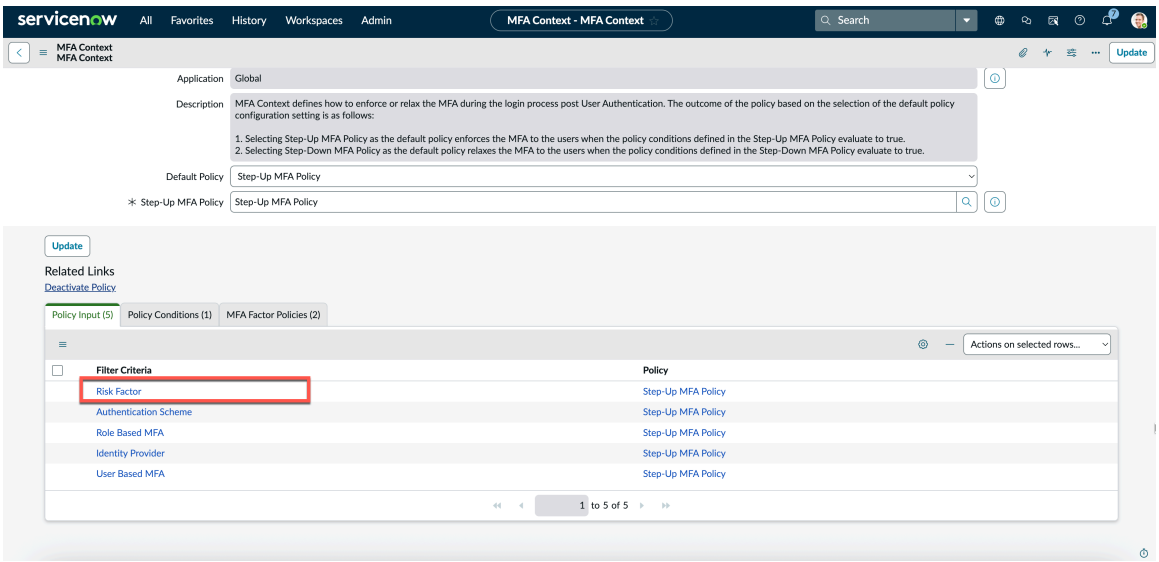
ポリシーと条件で認証後のコンテキストを作成する方法の詳細については、「認証後コンテキスト」を参照してください。

2. MFA ポリシーのコンテキストでの IDP 属性の使用
例：デバイスが信頼できない場合に、Okta IDP 属性からの MFA を有効にするように設定します。

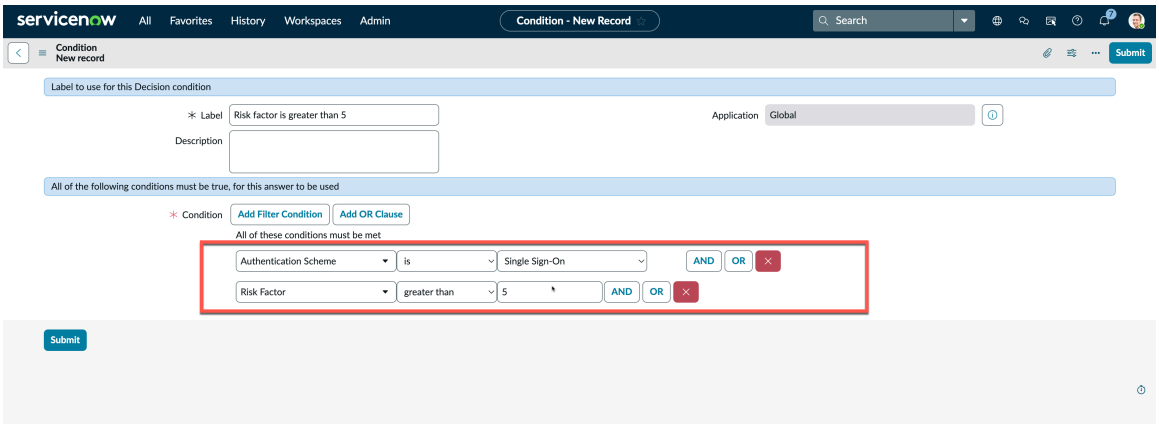
a. 移動先 すべて > 適応認証 > 認証ポリシーのコンテキスト > **MFA** 認証ポリシーのコンテキスト。.

b. ポリシー入力で、ポリシーの入力と条件を作成します。

- ポリシー入力：**device_trusted-okta** を追加します。



- ポリシー条件：**[device_trusted-okta]** は not_trusted であり、**[ID プロバイダー]** は Okta です。



この構成に基づいて、デバイスが Okta (IdP) から信頼されていない場合、ユーザーにはインスタンスにログインするための第 2 要素認証が表示されます。

ポリシーと条件で MFA コンテキストを作成する方法の詳細については、「マルチファクター認証コンテキスト」を参照してください。

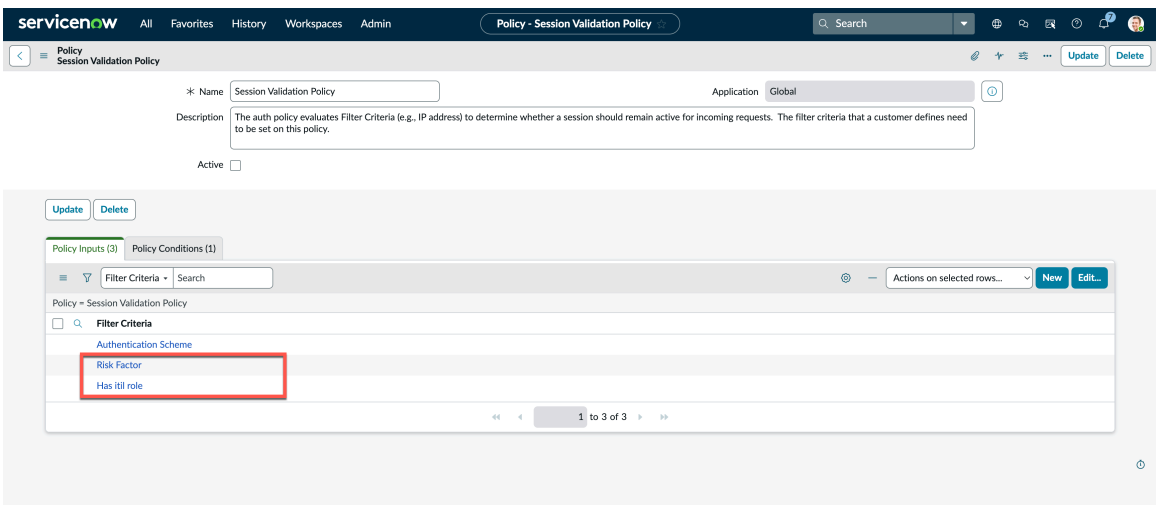
3. Zero Trust - Policy Based Session Access での IDP 属性の使用
例：デバイスが信頼できない場合に、Okta IDP 属性からの Itil ロールの特権を減らすように設定します。

   a. 移動先 すべて > **Zero Trust** アクセス > セッションアクセスロール構成.

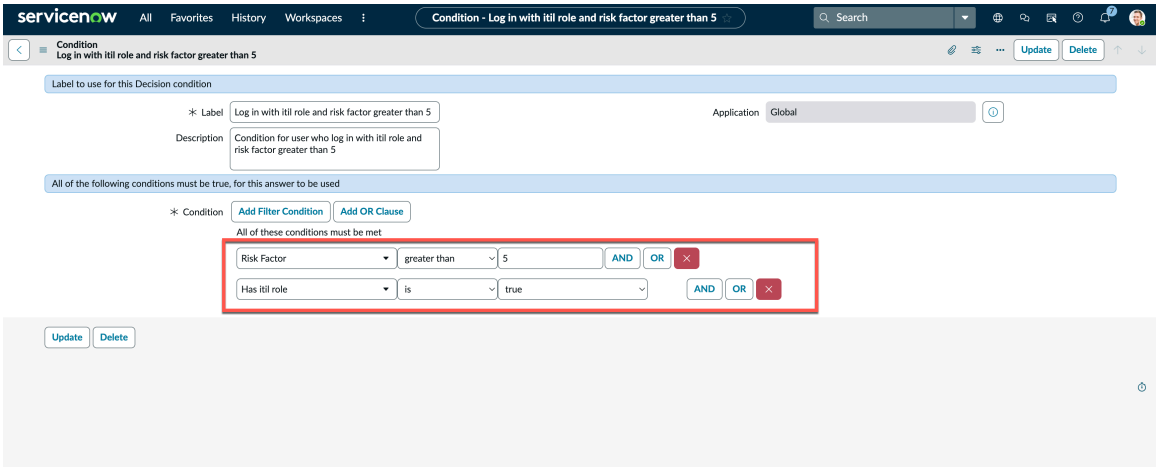
   b. セッションアクセスのロール構成を作成します。

   c. ポリシー入力で、ポリシーの入力と条件を作成します。

   - ポリシー入力：**device_trusted-okta** と **has itil role** を追加します。



- ポリシー条件：**[device_trusted-okta]** は not_trusted であり、**[ID プロバイダー]** は Okta であり、**[has itil role]** は true です。



この構成に基づいて、Okta (IdP) から信頼されていないデバイスを itil ユーザーが使用すると、ログインセッションに対するユーザーの権限が減らされます。

ポリシーと条件で Zero Trust - Policy Based Session Access を作成する方法の詳細については、「ゼロトラストアクセス (ZTA)」を参照してください。

OpenID Connect の ID プロバイダー属性

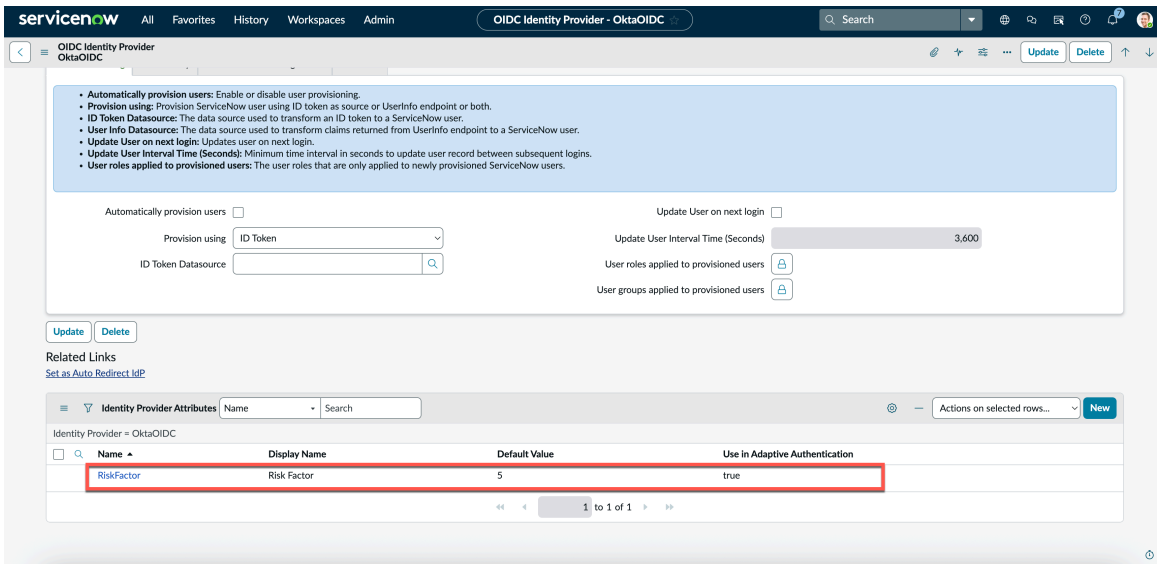
ID プロバイダー (IdP) の OpenID Connect (OIDC) から受信した ID プロバイダー属性を認証のフィルター基準として使用します。

ID トークンの一部として受け取った要求に基づいて、IdP 属性を手動で作成できます。

注:

- ID プロバイダーフィルターは、Zero Trust アクセス機能で使用できます。詳細については、「ゼロトラストアクセス (ZTA)」を参照してください。
- IdP 属性フィルター基準は、 認証後コンテキスト、 ゼロトラストアクセス (ZTA) セッション降格、および マルチファクター認証コンテキストで使用できます。

[ID プロバイダー属性] セクションから [新規] を選択して IdP 属性を追加することから構成を開始し、それらの属性を true に設定して適応認証に使用します。



自動翻訳

ID プロバイダー属性の OIDC 構成で定義された **RiskFactor** は、ID トークン要求から取得されます。この値は、IdP 側で構成された既存の要求またはカスタム要求にすることができます。この要求をさまざまな認証コンテキストで使用して、ユーザーのログイン動作をカスタマイズおよび制御します。

[ID プロバイダー属性 (Identity Provider Attributes)] が次の詳細とともに表示されます。

場所のフィルター基準フォーム

フィールド	説明
名前	ID プロバイダーによって提供される属性名
表示名	表示名はフィルター基準に使用される詳細な名前です。 注: ID プロバイダーによって指定された表示名が長くて判読できないことがあるため、表示名として判読可能な名前を指定できます。
デフォルト値	デフォルト値は、SAML 応答に属性がない場合にフィルター基準の評価に使用されます。
適応認証で使用	適応認証で属性を使用するオプション。

注: Azure IdP から入力される属性は、属性の名前の長さにより、名前と表示名が文字数に制限されています。

[ID プロバイダー属性 (Identity Provider Attributes)] セクションで [新規] を選択して、新しい属性を追加することもできます。

[適応認証で使用 (Use in Adaptive Authentication)] が true に設定されている場合、選択した属性が汎用的なフィルター基準のフィルター基準として追加されます。たとえば、**risk_score** を true に設定します。汎用的なフィルター基準ページに新しいフィルターが作成されます。

ID プロバイダー属性を OIDC のフィルター基準として使用する

OpenID Connect (OIDC) 応答の ID プロバイダー (IDP) 属性を認証ポリシーのフィルター基準として使用します。

始める前に

必要なロール：admin

ポリシーコンテキスト (事前認証、認証後、マルチファクター認証) とフィルター基準 (ロール、グループ、IP、場所) をポリシーの入力と条件とともに使用して、セッションアクセスポリシーを作成できます。

次の手順は、SAML 応答の IdP 属性をポリシーの入力として設定し、[認証後のコンテキスト]、[マルチファクター認証 **(MFA)** コンテキスト]、および **[Zero Trust - Policy Based Session Access]** で認証を制御する手順を示しています。

Okta IDP 属性を次のスクリーンショットに示します。[認証後のコンテキスト]、[マルチファクター認証 **(MFA)** コンテキスト]、および **[Zero Trust - Policy Based Session Access]** ポリシーで使用するには、[適応認証で使用 (Use in Adaptive Authentication)] を true に設定する必要があります。

注: 認証後、MFA、 Zero Trust - Policy Based Session Access のポリシーは、ユーザーが認証情報または SSO 応答を入力した後に実行されます。

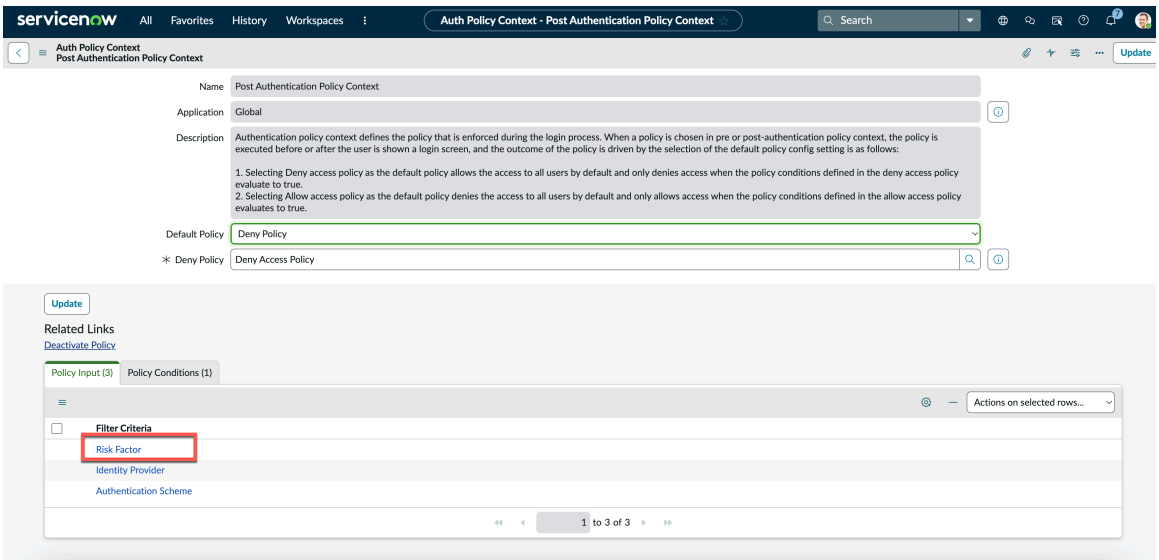
手順

1. 認証後ポリシーのコンテキストでの IDP 属性の使用
   例:デバイスが信頼できる場合に Okta IDP 属性からのログインを有効にするように設定します。

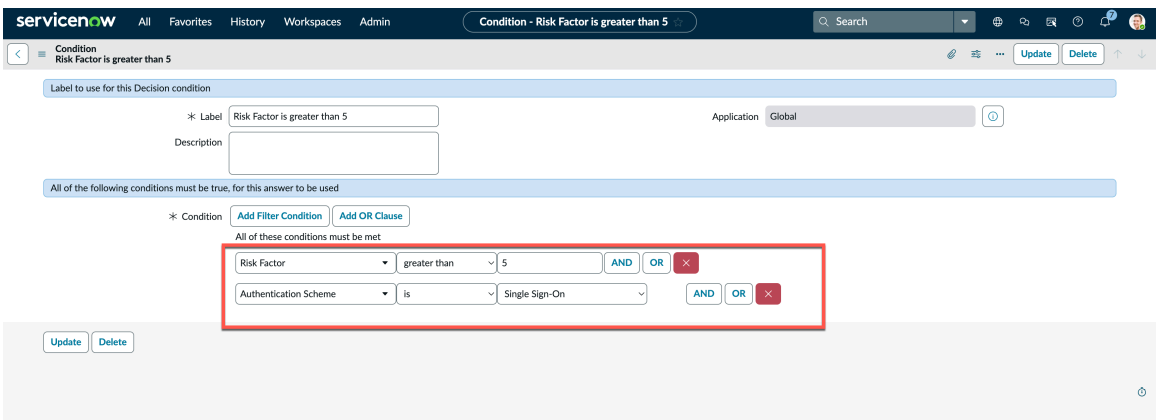
   a. 移動先 すべて > 適応認証 > 認証ポリシーのコンテキスト > 認証後ポリシーのコンテキスト。.

   b. [許可ポリシー] を選択し、ポリシーレコードを開きます。

   c. ポリシー入力で、ポリシーの入力と条件を作成します。
      - ポリシー入力: リスク要因を追加します。



      - ポリシー条件: リスクファクター が 5 より大きく、 認証スキーム が シングルサインオン です。



   この構成に基づいて、デバイスが Okta (IdP) から信頼されている場合、ユーザーはインスタンスに対して認証されます。

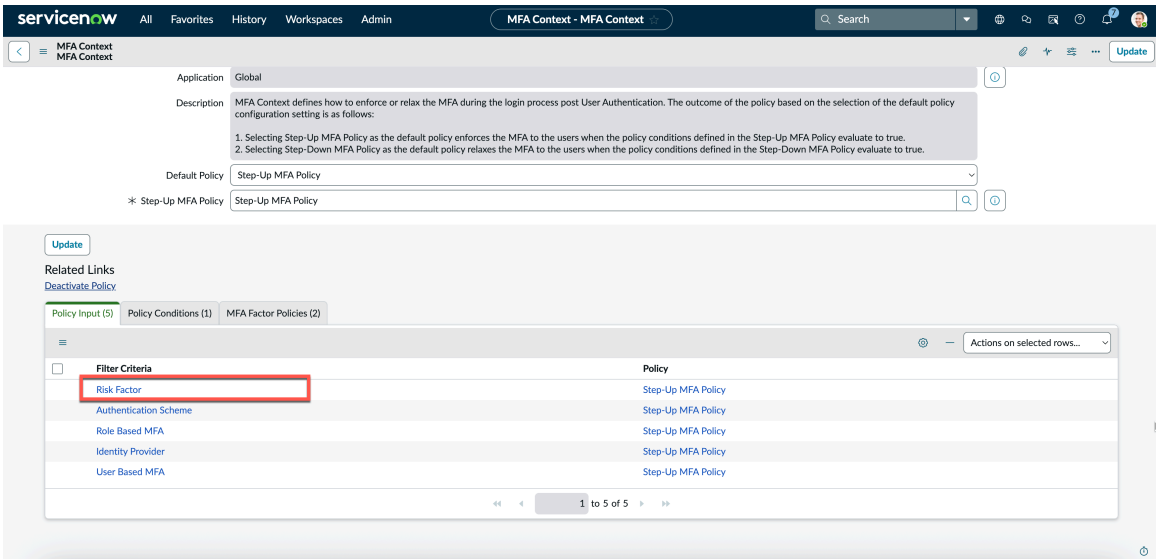
   ポリシーと条件で認証後のコンテキストを作成する方法の詳細については、「認証後コンテキスト」を参照してください。

2. MFA ポリシーのコンテキストでの IDP 属性の使用
   例：デバイスが信頼できない場合に、Okta IDP 属性からの MFA を有効にするように設定します。

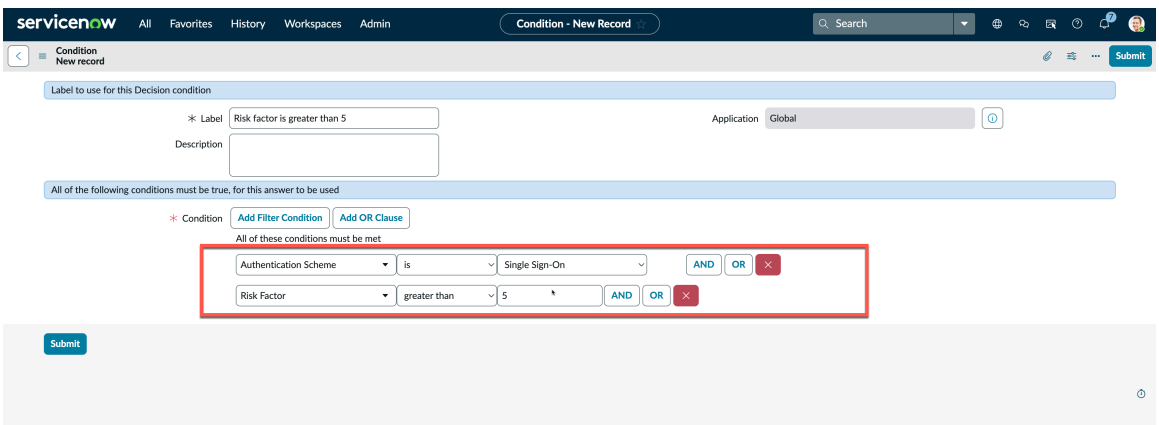
a. 移動先 すべて > 適応認証 > 認証ポリシーのコンテキスト > **MFA** 認証ポリシーのコンテキスト。.

b. ポリシー入力で、ポリシーの入力と条件を作成します。

- ポリシー入力: リスク要因を追加します。



- ポリシー条件: リスクファクター が 5 より大きく、 認証スキーム が シングルサインオン です。



この構成に基づいて、デバイスが Okta (IdP) から信頼されていない場合、ユーザーにはインスタンスにログインするための第 2 要素認証が表示されます。

ポリシーと条件で MFA コンテキストを作成する方法の詳細については、「マルチファクター認証コンテキスト」を参照してください。

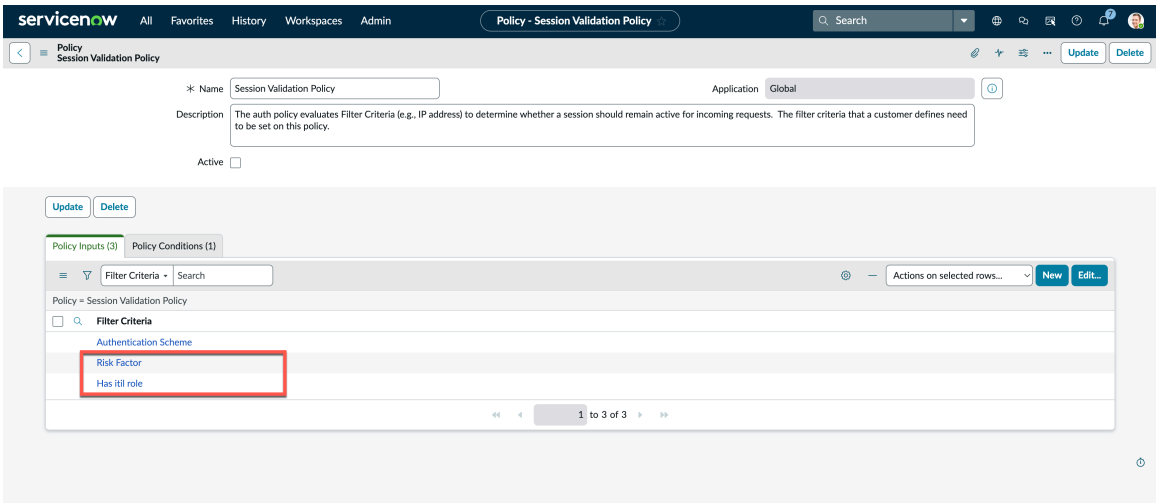
3. Zero Trust - Policy Based Session Access での IDP 属性の使用
例：デバイスが信頼できない場合に、Okta IDP 属性からの Itil ロールの特権を減らすように設定します。

a. 移動先 すべて > **Zero Trust** アクセス > セッションアクセスロール構成.

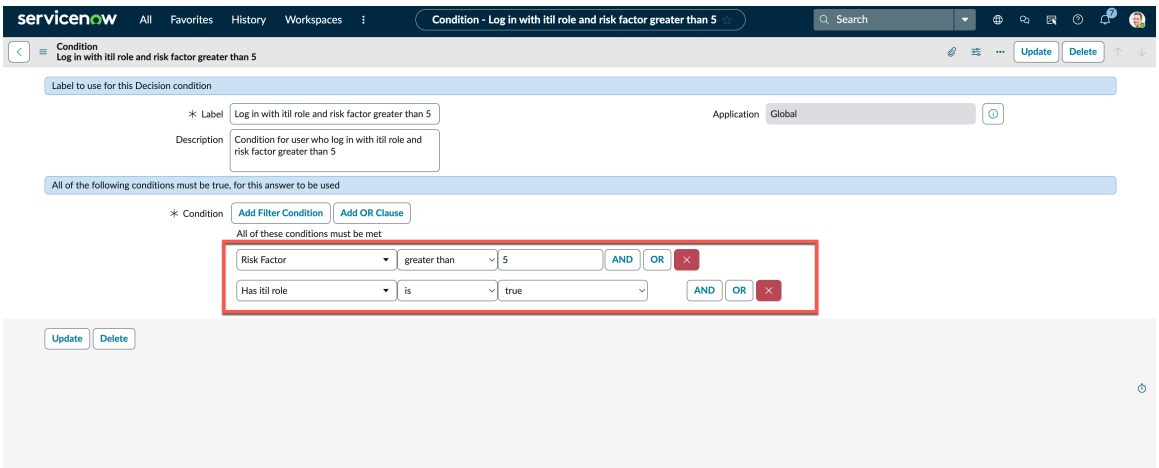
b. セッションアクセスのロール構成を作成します。

c. ポリシー入力で、ポリシーの入力と条件を作成します。

- ポリシー入力:[リスク要因を追加 (Add Risk Factor)] と **[itil** ロールがある **(Has itil role)]**



- ポリシー条件: リスクファクター が 5 より大きく、 認証スキーム が シングルサインオン です。



この構成に基づいて、Okta (IdP) から信頼されていないデバイスを itil ユーザーが使用すると、ログインセッションに対するユーザーの権限が減らされます。

ポリシーと条件で Zero Trust - Policy Based Session Access を作成する方法の詳細については、「ゼロトラストアクセス (ZTA)」を参照してください。

認証ポリシーのコンテキスト

認証ポリシーコンテキストを使用して、インスタンスで認証ポリシーを適用する方法とタイミングを決定します。

認証コンテキストは、ログインプロセス中にポリシーを適用する方法とタイミングを定義します。ポリシーコンテキストにポリシーを割り当て、インスタンスが認証を処理する方法に関する入力と条件を定義します。

事前承認コンテキスト

事前承認コンテキストのポリシーは、ユーザーによるインスタンスへの初回アクセス時に、ログイン画面を表示する前に実行されます。事前承認コンテキストを使用して、選択したポリシーに基づいて

ユーザーに対してログイン認証情報が要求される前に、アクセスを許可または拒否できます。これらのポリシーはユーザーが情報を入力する前に評価されるため、ユーザーのロールやグループなどの基準を考慮することはできません。

このコンテキストの詳細については、「事前認証コンテキスト」を参照してください。

認証後コンテキスト

承認後コンテキストのポリシーは、ユーザーが認証情報または SSO 応答を入力した後に実行されます。インスタンスは、選択したポリシーに基づいてアクセスを許可または拒否します。ユーザーはログイン認証情報を使用して本人確認を行ったため、ポリシーでは、ユーザー情報を使用してアクセスを許可するかどうかを決定できます。

このコンテキストの詳細については、「認証後コンテキスト」を参照してください。

MFA (マルチファクター認証) コンテキスト

MFA コンテキストに割り当てられたポリシーは、ログインプロセス中に MFA を適用するかどうかを定義します。インスタンスで MFA が適用されるかどうかは、このコンテキストのポリシーの構成によって決まります。このコンテキストの詳細については、「マルチファクター認証コンテキスト」を参照してください。

アカウント復旧のコンテキスト

アドミニストレーターは、アカウント復旧 (ACR) を設定して、SSO の設定ミスや期限切れの証明書への対処などの復旧アクティビティを実行できます。アカウント復旧を使用するには、少なくとも 1 つのアドミンアカウントをアカウント復旧ユーザーとして登録する必要があります。少なくとも 1 つのアカウントが設定されるまで、インスタンスでシングルサインオンを有効にすることはできません。設定可能なコンテキストの詳細については、「アカウント復旧のコンテキスト」を参照してください。

セッション検証コンテキスト

セッション検証コンテキストは、適応認証ポリシーフレームワークで使用できます。このフレームワークでは、認証ポリシーを使用して認証要求 (セッション) を評価してから、ポリシー条件に基づいてアクセスを拒否または許可します。詳細については、「セッション検証コンテキスト」を参照してください。

デフォルトポリシー

ポリシーコンテキスト内で、デフォルトポリシーを [デフォルトポリシー] フィールドに定義できます。このデフォルトは、インスタンスがポリシーの結果にどのように応答するかを定義します。使用可能なデフォルトのポリシーオプションは、使用しているコンテキストによって決まります。これらのオプションの詳細については、個々のコンテキストの説明書を参照してください。

事前認証コンテキスト

事前認証ポリシーコンテキストは、ログインプロセス中にポリシーを適用する方法とタイミングを定義します。このコンテキストで使用されるポリシーは、ユーザーにログイン画面が表示される前に実行されます。

事前認証コンテキストレコード

事前認証コンテキストのポリシーは、ユーザーが初めてインスタンスにアクセスしたときに、ログイン画面が表示される前に実行されます。

事前認証コンテキストを使用して、選択したポリシーに基づいてユーザーがログイン認証情報を要求する前に、アクセスを許可または拒否できます。これらのポリシーはユーザーが情報を入力する前に評価されるため、ユーザーのロールやグループなどの基準を考慮することはできません。

事前認証ポリシーコンテキストレコードのフィールドを使用して、インスタンスでのポリシーの使用方法を定義します。

事前認証コンテキストフォーム

フィールド	説明
名前	ポリシーコンテキストの名前このフィールドは静的であり、変更することはできません。
説明	コンテキストの説明
デフォルトポリシー	ポリシーを評価するときの、このコンテキストのデフォルト動作を定義します。次のオプションのいずれかを選択します。 許可ポリシー デフォルトですべてのユーザーへのアクセスを拒否し、 [許可ポリシー] フィールドでポリシーが選択した条件が true と評価された場合にのみアクセスを許可します。 拒否ポリシー デフォルトですべてのユーザーへのアクセスを許可し、 [拒否ポリシー] フィールドでポリシーが選択した条件が true と評価された場合にのみアクセスを拒否します。
許可ポリシー	このコンテキストで使用されるポリシー。このフィールドは、[デフォルトポリシー] フィールドが [許可ポリシー] に設定されている場合にのみ表示されます。
拒否ポリシー	このコンテキストで使用されるポリシー。このフィールドは、[**Default Policy** (デフォルトポリシー)] フィールドが [**Deny Policy** (拒否ポリシー)] に設定されている場合にのみ表示されます。

注:

事前認証ポリシーコンテキストでは、IP フィルター、信頼できるモバイルアプリフィルター、および場所フィルターの基準のみを使用できます。

ポリシーの入力と条件

[ポリシー入力] タブと [ポリシー条件] タブには、[許可ポリシー] または [拒否ポリシー] フィールドで選択されたポリシーの入力と条件が表示されます。これらのタブは参照できますが、ポリシーの入力や条件を変更するために使用することはできません。ポリシーを変更するには、[許可ポリシー] または [拒否ポリシー] フィールドの横にある参照アイコン ( ) を使用してポリシーに移動します。

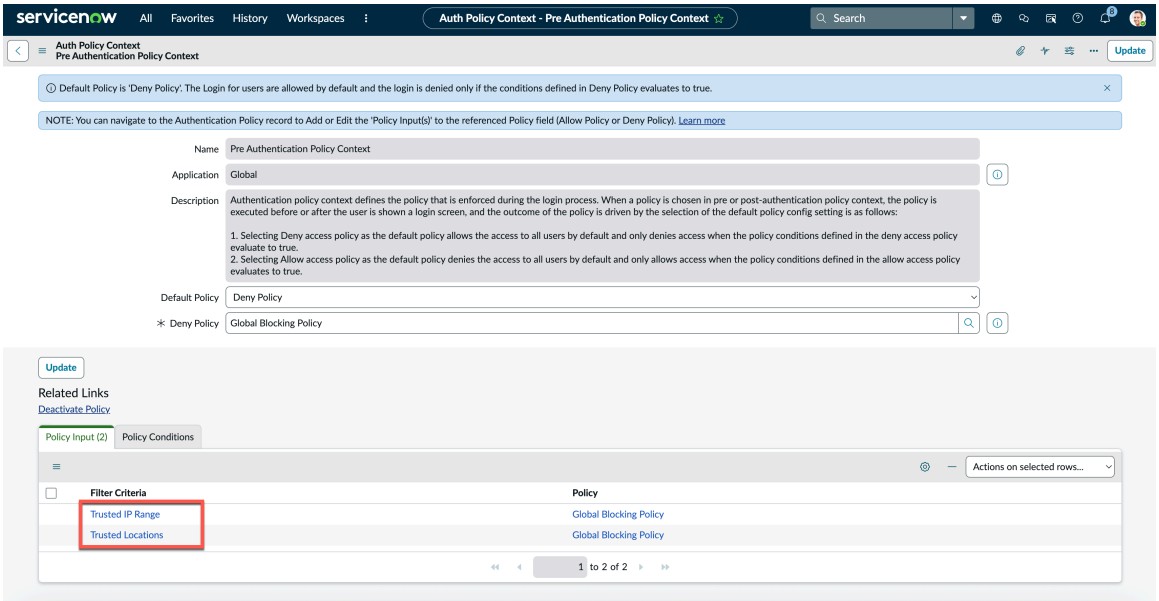
この例は、デフォルトでアクセスを拒否するように設定された事前認証ポリシーコンテキストレコードを示しています。コンテキストでは、[アクセスポリシーの拒否] と呼ばれるポリシーを使用します。そのポリシーには、[ポリシー入力] タブと [ポリシー条件] タブに表示される一連の入力と条件があります。

注:

- 事前認証ポリシーコンテキストで使用できるのは、IP ベースのフィルター、場所ベースのフィルター、または信頼できるモバイルアプリフィルターのみです。
- 非絶対条件またはフィルター基準が設定された事前認証がある場合は常に、ポリシーまたはコンテキストを構成できないことを示すエラーメッセージが表示されます。事前認証コンテキストのすべての入力を検証してから、インスタンスに対して実行することをお勧めします。

  たとえば、アドミニストレーターが信頼できるネットワーク外にいて、IP 範囲を使用して事前認証コンテキストを構成している場合、IP 範囲がアドミンの現在のセッションと一致しない場合、アドミニストレーターはブロックされます。

[事前認証ポリシーのコンテキスト] フォーム



認証後コンテキスト

認証後ポリシーのコンテキストは、ログインプロセス中にポリシーを適用する方法とタイミングを定義します。このコンテキストで使用されるポリシーは、ユーザーにログイン画面が表示された後に実行されます。

認証後コンテキストレコード

承認後コンテキストのポリシーは、ユーザーが認証情報または SSO 応答を入力した後に実行されます。インスタンスは、選択したポリシーに基づいてアクセスを許可または拒否します。ユーザーはログイン認証情報を使用して本人確認を行うため、ポリシーでは、ロールやグループなどのユーザー情報を使用してアクセスを許可するかどうかを決定できます。

認証後ポリシーコンテキストレコードのフィールドを使用して、インスタンスでのポリシーの使用方法を定義します。

認証後コンテキストフォーム

フィールド	説明
名前	ポリシーコンテキストの名前このフィールドは静的であり、変更することはできません。
説明	コンテキストの説明

認証後コンテキストフォーム (続く)

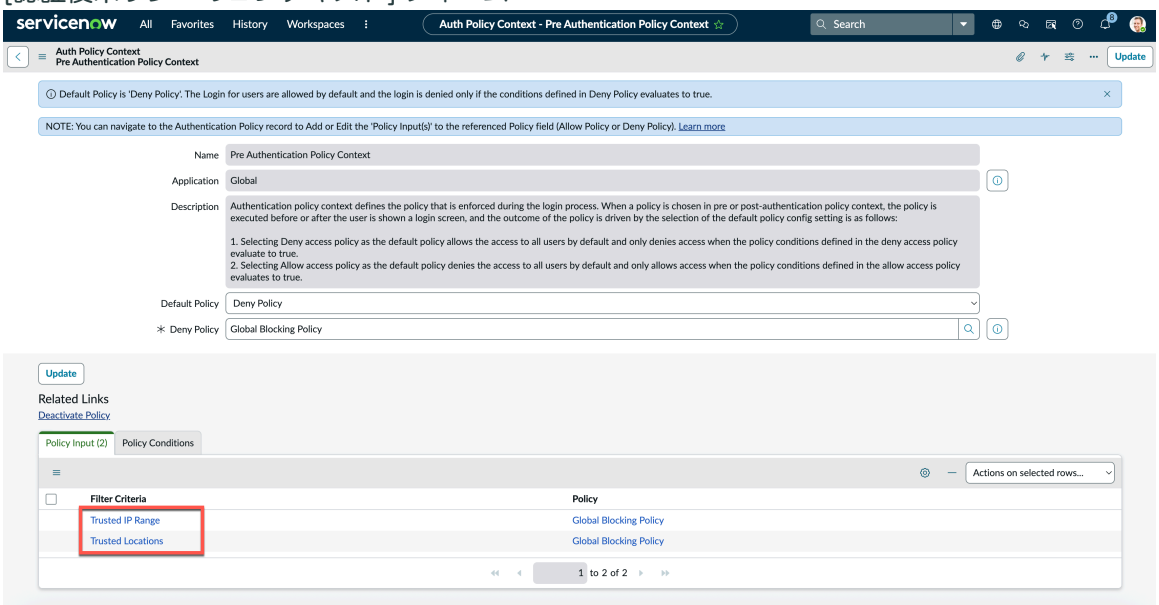
フィールド	説明
デフォルトポリシー	ポリシーを評価するときの、このコンテキストのデフォルト動作を定義します。次のオプションのいずれかを選択します。 許可ポリシー デフォルトですべてのユーザーへのアクセスを拒否し、 [許可ポリシー] フィールドでポリシーが選択した条件が true と評価された場合にのみアクセスを許可します。 拒否ポリシー デフォルトですべてのユーザーへのアクセスを許可し、 [拒否ポリシー] フィールドでポリシーが選択した条件が true と評価された場合にのみアクセスを拒否します。
許可ポリシー	このコンテキストで使用されるポリシー。このフィールドは、[デフォルトポリシー] フィールドが [許可ポリシー] に設定されている場合にのみ表示されます。
拒否ポリシー	このコンテキストで使用されるポリシー。このフィールドは、[デフォルトポリシー] フィールドが [拒否ポリシー] に設定されている場合にのみ表示されます。

ポリシーの入力と条件

[ポリシー入力] タブと [ポリシー条件] タブには、[許可ポリシー] または [拒否ポリシー] フィールドで選択されたポリシーの入力と条件が表示されます。これらのタブは参照できますが、ポリシーの入力や条件を変更するために使用することはできません。ポリシー設定を変更するには、[許可ポリシー] または [拒否ポリシー] フィールドの横にある参照アイコン (ⓘ) を使用してポリシーに移動します。

この例は、デフォルトでアクセスを拒否するように設定された認証後ポリシーコンテキストレコードを示しています。コンテキストでは、[アクセスポリシーの拒否] と呼ばれるポリシーを使用します。そのポリシーには、[ポリシー入力] タブと [ポリシー条件] タブに表示される一連の入力と条件があります。

[認証後ポリシーのコンテキスト] フォーム



マルチファクター認証コンテキスト

マルチファクター認証 (MFA) ポリシーコンテキストでは、ポリシーを使用して、ログインプロセス中に MFA を適用する方法とタイミングを定義します。

MFA コンテキストレコード

MFA ポリシーコンテキストは、ユーザーがログイン時に 2 番目の認証フォームを提供する必要があるかどうかを定義します。このコンテキストは、認証後および認証前のポリシーとしてインスタンスへのアクセスを拒否するわけではありません。このコンテキストで選択したポリシーは、マルチファクター認証のユーザーまたはロールベースの構成よりも優先されます。

MFA コンテキストにアクセスするには、 すべて > 多要素認証 > **MFA** コンテキスト.

認証後ポリシーコンテキストレコードのフィールドを使用して、インスタンスでのポリシーの使用方法を定義します。

注:

- デフォルトのポリシーがステップアップ **MFA** ポリシーの場合、ステップアップ **MFA** ポリシーで構成されたポリシーが true と評価されると、ユーザーにはマルチファクター認証が表示されます。ポリシーは、ユーザーまたはロールベースの構成よりも優先されます。
- SSO ログインを使用する MFA は、glide.authenticate.mfa.with.multisso.enabled プロパティが true に設定されている場合にのみ使用できます。
- [認証ポリシー] レコードに移動して、参照されているポリシーフィールド ([ステップアップ **MFA** ポリシー] または [ステップダウン **MFA** ポリシー]) に「ポリシー入力」を追加または編集します。
- MFA コンテキストポリシーは、ユーザーのログインにのみ適用されます。API 認証、基本認証、および OAuth リソース所有者のパスワード認証情報の付与には適用されません。

MFA コンテキストフォーム

フィールド	説明
名前	ポリシーコンテキストの名前このフィールドは静的であり、変更することはできません。
説明	コンテキストの説明
デフォルトポリシー	ポリシーを評価するときの、このコンテキストのデフォルト動作を定義します。次のオプションのいずれかを選択します。 ステップアップ MFA ポリシー [ステップアップ **MFA** ポリシー] フィールドで定義されたポリシー条件が true と評価された場合に MFA がユーザーに強制されます。 ステップダウン MFA ポリシー デフォルトで MFA を適用します。[ステップダウン **MFA** ポリシー] フィールドで定義されたポリシー条件が true と評価された場合にかぎり、MFA はユーザーに強制されません。
ステップアップ MFA ポリシー	このコンテキストで使用されるポリシー。このフィールドは、[デフォルトポリシー] フィールドが [ステップアップ **MFA** ポリシー] に設定されている場合にのみ表示されます。

MFA コンテキストフォーム (続く)

フィールド	説明
ステップダウン MFA ポリシー	このコンテキストで使用されるポリシー。このフィールドは、[デフォルトポリシー] フィールドが [ステップダウン **MFA** ポリシー] に設定されている場合にのみ表示されます。

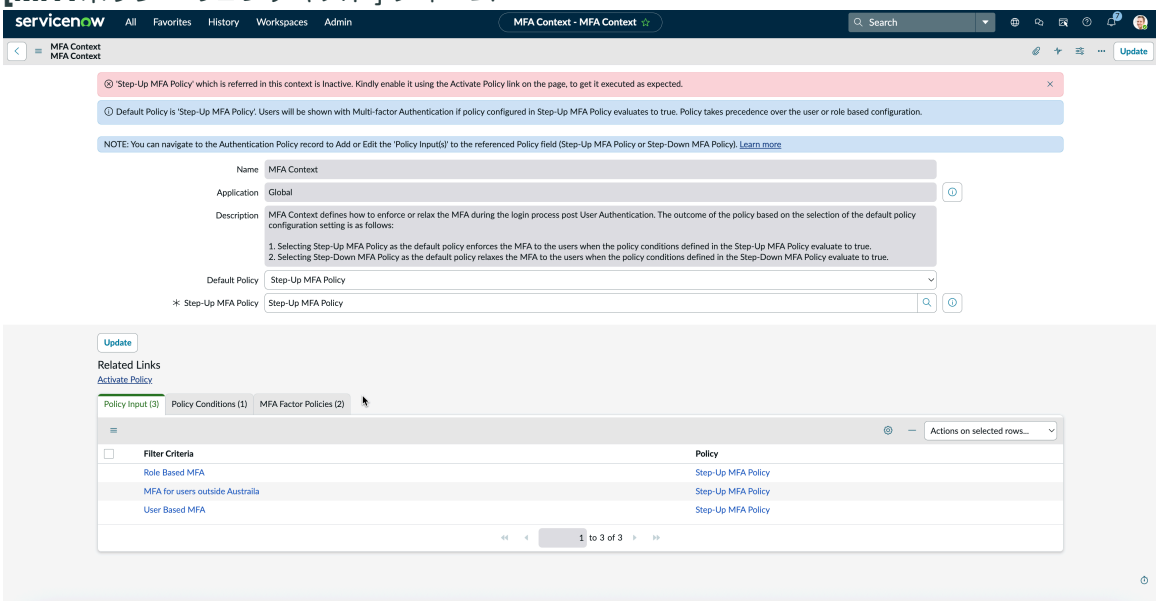
ポリシーの入力と条件

[**Policy Input** (ポリシー入力)] タブと [**Policy Conditions** (ポリシー条件)] タブには、[**Step-Up MFA Policy** (ステップアップ **MFA** ポリシー)] または [**Step-Down MFA Policy** (ステップダウン **MFA** ポリシー)] フィールドで選択されたポリシーの入力と条件が表示されます。これらのタブは参照できますが、ポリシーの入力や条件を変更するために使用することはできません。ポリシー設定を変更するには、[ステップアップ **MFA** ポリシー] または [ステップダウン **MFA** ポリシー] フィールドの横にある参照アイコン (ⓘ) を使用してポリシーに移動します。

注: ポリシー条件はここから作成できますが、ポリシーページから新しいポリシー条件を追加することをお勧めします。

この例は、ステップアップ MFA ポリシーを使用して構成された MFA コンテキストレコードを示しています。このデフォルトポリシーは、ポリシーで定義されている条件が true と評価された場合にのみ MFA が適用されることを意味します。コンテキストは、[ステップアップ **MFA** ポリシー] と呼ばれるポリシーを使用します。そのポリシーには、[ポリシー入力] タブと [ポリシー条件] タブに表示される一連の入力と条件があります。

[MFA ポリシーのコンテキスト] フォーム



MFA 要素ポリシー

MFA 要素ポリシーは、組織のセキュリティ体制の重要なコンポーネントであり、パスワード以外の追加の検証手順を適用できます。これらのポリシーは、ユーザーがアクセスするために採用する必要がある認証方法を定義し、柔軟でカスタマイズ可能な認証アプローチを提供します。詳細については、「マルチファクター認証要素ポリシー」を参照してください。

アカウント復旧のコンテキスト

アカウント復旧コンテキストでは、ポリシーを使用して、アカウント復旧を確立する方法とタイミングを定義します。

アドミニストレーターは、次の場所に移動して、このコンテキストとそれに関連するポリシーを表示および変更できます マルチプロバイダー **SSO** > アカウント復旧 > アカウント復旧のコンテキスト.

注: デフォルトでは、ポリシーは [許可ポリシー] です。ユーザーのログインはデフォルトで制限されており、[許可ポリシー] で定義された条件が true と評価された場合にのみログインが許可されます。

アカウント復旧コンテキストレコードのフィールドを使用して、インスタンスでのポリシーの使用方法を定義します。

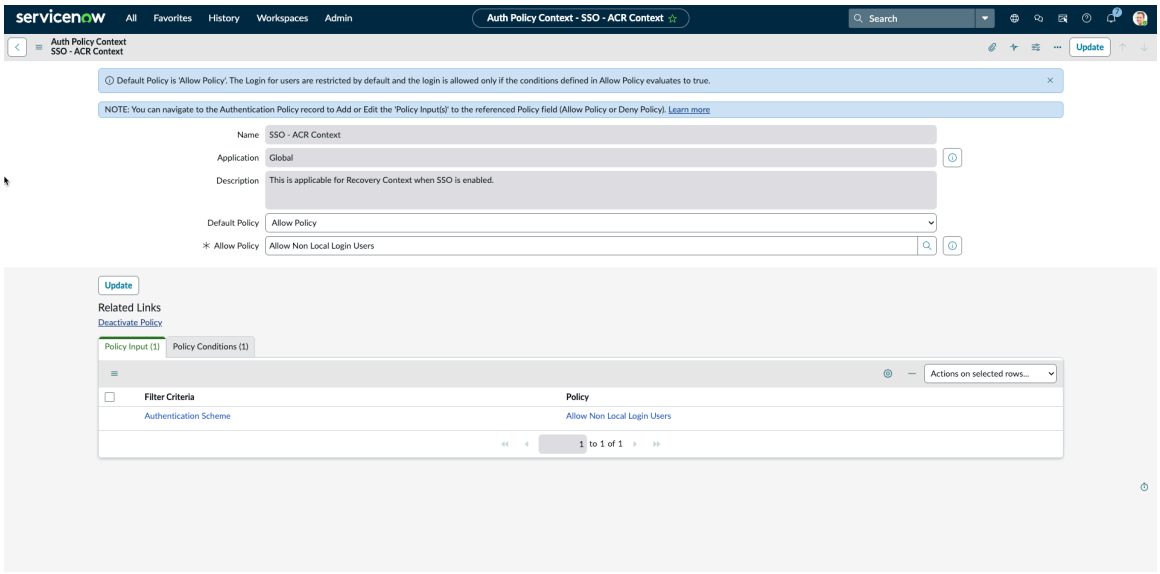
アカウント復旧のコンテキストフォーム

フィールド	説明
名前	ポリシーコンテキストの名前このフィールドは静的であり、変更することはできません。
説明	コンテキストの説明
デフォルトポリシー	ポリシーを評価するときの、このコンテキストのデフォルト動作を定義します。次のオプションのいずれかを選択します。 • 許可ポリシー • 拒否ポリシー
許可ポリシー	このフィールドは、[デフォルトポリシー] フィールドが [許可ポリシー] に設定されている場合にのみ表示されます。
拒否ポリシー	このフィールドは、[デフォルトポリシー] フィールドが [拒否ポリシー] に設定されている場合にのみ表示されます。

ポリシーの入力と条件

[ポリシー入力] タブと [ポリシー条件] タブには、[許可ポリシー] または [拒否ポリシー] フィールドで選択されたポリシーの入力と条件が表示されます。これらのタブは参照できますが、ポリシーの入力や条件を変更するために使用することはできません。ポリシー設定を変更するには、[許可ポリシー] または [拒否ポリシー] フィールドの横にある情報アイコンを使用してポリシーに移動します。

注: ポリシー条件はここから作成できますが、ポリシーページから新しいポリシー条件を追加することをお勧めします。



セッション検証コンテキスト

セッション検証コンテキストは、セッションまたは Cookie のハイジャックに対する追加の保護レイヤーとして使用します。

セッション検証コンテキストは、適応認証ポリシーフレームワークで使用できます。このフレームワークでは、認証ポリシーを使用して認証要求を評価してから、ポリシーの入力と条件に基づいてアクセスを拒否または許可します。

セッション検証コンテキストポリシーは、認証後ポリシーと組み合わせて使用することができ、admin はログインセッション中に特定またはすべてのユーザーに IP 制限を適用できます。

セッション検証コンテキスト機能は、設定した条件に基づく IP アドレスを評価し、セッション内のインスタンスへのアクセスを許可します。セッション検証コンテキストの結果は次のとおりです。

- 拒否ポリシー：デフォルトのポリシーとして [アクセス拒否ポリシー] を選択すると、ユーザーはデフォルトでセッションを続行できます。セッションは、アクセス拒否ポリシーで定義されているポリシー条件の 1 つが true と評価された場合にのみ終了します。
- 許可ポリシー：デフォルトのポリシーとして [アクセス許可ポリシー] を選択すると、アクセス許可ポリシーで定義されているポリシー条件の 1 つが true と評価されない限り、ユーザーセッションがすぐに終了します。

注:

- 認証ポリシーのセッション検証コンテキストは、デフォルトで許可ポリシーに設定されています。
- セッション検証コンテキストは、許可ポリシーによって実装されます。コンテキストを拒否ポリシーに設定することはお勧めしません。

セッション検証コンテキストは、次のメカニズムに基づいて機能します。

- ユーザー要求からのセッション作成時にユーザーの IP アドレスをキャプチャし、セッションとデータベースに保存します。
- 要求の IP アドレスがセッション内の IP アドレスと異なる場合、または顧客が定義した有効な IP 範囲の範囲外である場合に、要求を拒否します。

注: セッション検証コンテキストは以下のとおりです。

- 認証ユーザーのみ利用可能。
- ゲストユーザーセッションまたはネイティブモバイルアプリには適用されません。
- オプションであり、構成できるという要件に基づいています。
- ログイン後の要求に対してのみ実行されます。

セッション検証の利点

セッション検証コンテキストには、次の利点があります。

- ハイジャッカーがユーザーのセッション Cookie をあるデバイスから別のデバイスにコピーしてセッションを代理操作するとき、ServiceNow へのアクセスを制限します。
- ユーザーが安全でないネットワークを使用している場合、ユーザーのセッションアクセスを制限します。
- ユーザーログインのユーザーグループまたはロール別にさまざまなルールと IP 範囲を設定します。

セッション検証コンテキストレコード

セッション検証コンテキストのポリシーは、ログイン後の要求を実行します。

セッション検証ポリシーコンテキストレコードのフィールドを使用して、インスタンスでのポリシーの使用方法を定義します。

セッション検証コンテキストフォーム

フィールド	説明
名前	ポリシーコンテキストの名前このフィールドは静的であり、変更することはできません。
説明	コンテキストの説明。
デフォルトポリシー	ポリシーを評価するときの、このコンテキストのデフォルト動作を定義します。次のオプションのいずれかを選択します。 許可ポリシー デフォルトですべてのユーザーへのアクセスを拒否し、 [許可ポリシー] フィールドの条件が true と評価された場合にのみアクセスを許可します。 拒否ポリシー デフォルトですべてのユーザーへのアクセスを許可し、 [拒否ポリシー] フィールドの条件が true と評価された場合にのみアクセスを拒否します。
許可ポリシー	このコンテキストで使用されるポリシー。このフィールドは、[デフォルトポリシー] フィールドが [許可ポリシー] に設定されている場合にのみ表示されます。
拒否ポリシー	このコンテキストで使用されるポリシー。このフィールドは、[デフォルトポリシー] フィールドが [拒否ポリシー] に設定されている場合にのみ表示されます。

ポリシー入力とポリシー条件に基づいて、[セッション検証ポリシー]を [許可ポリシー] または [拒否ポリシー] として選択できます。

注:

セッション検証ポリシーには、IP、ロール、およびグループのフィルター基準のみを使用できます。

ポリシーの入力と条件

[ポリシー入力] タブと [ポリシー条件] タブには、[許可ポリシー] または [拒否ポリシー] フィールドで選択されたポリシーの入力と条件が表示されます。これらのタブは参照できますが、ポリシーの入力や条件を変更するために使用することはできません。ポリシーを変更するには、[許可ポリシー] または [拒否ポリシー] フィールドの横にある参照アイコン ( ) を使用してポリシーに移動します。

セッション検証コンテキストのアクティブ化

セッション検証コンテキストを使用して、ハイジャッカーがあるデバイスから別のデバイスにユーザーのセッション Cookie をコピーしてセッションを代理操作するときに ServiceNow へのアクセスを制限したり、安全でないネットワークを使用している場合はユーザーのセッションアクセスを制限したりします。

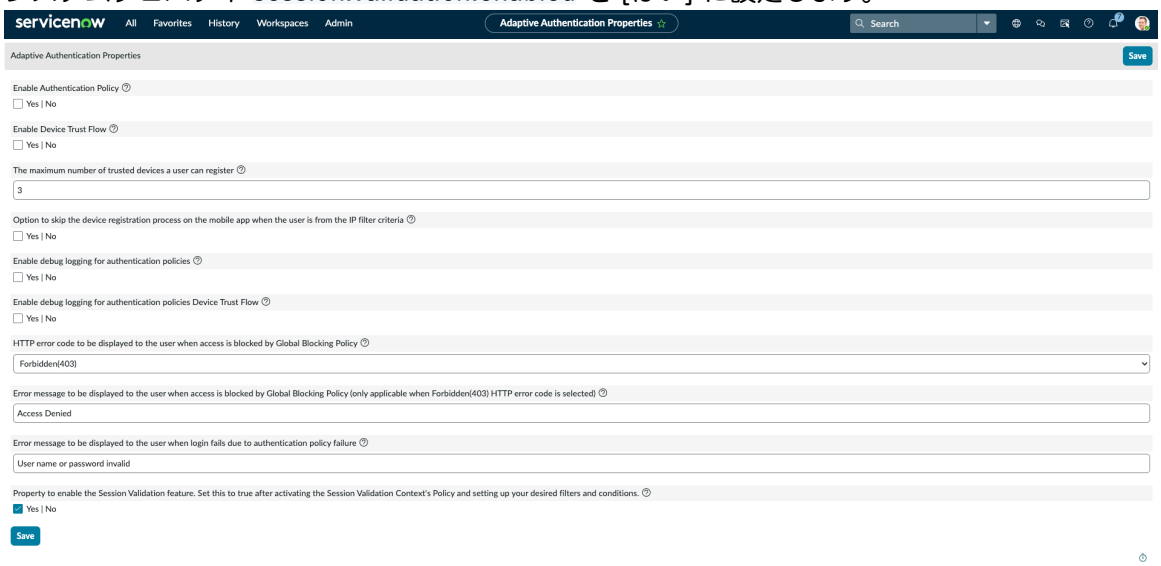
始める前に
必要なロール：admin

セッション検証を使用するには、次の手順を実行する必要があります。

手順

1. 移動先 すべて > 適応認証 > 認証ポリシー > すべてのポリシー.
2. [ポリシー (sys_authentication_policy_list.do)] ページでセッション検証ポリシーを選択します。
3. [ポリシー入力] と [ポリシー条件] を指定します。
4. 追加したフィルターの条件を **true** または **false** に設定します。
5. ポリシー入力と条件で [セッション検証ポリシー] を設定した後、[アクティブ] チェックボックスをオンにしてポリシーを有効にします。
6. インスタンスに SSO が設定されている場合は、sso_properties テーブルのデフォルトの ID プロバイダーの **[AuthnRequest** の強制] チェックボックスをオンにします。
7. 移動先 すべて > 適応認証 > 認証ポリシー > プロパティ.

8. システムプロパティ session.validation.enabled を [はい] に設定します。



結果

セッション検証機能が有効になってます。ポリシーのポリシー入力と条件を設定して、この機能を使用できます。詳細については、「チュートリアル：セッション検証の設定」を参照してください。

チュートリアル：セッション検証の設定

適応認証フレームワーク内でセッション検証を設定し、セッションまたは Cookie のハイジャックに対する保護の追加レイヤーとして提供します。

始める前に

必要なロール：admin

必要なプラグイン：適応認証 (com.snc.adaptive_authentication)

セッション検証を設定する際は、次の手順を実行する必要があります。

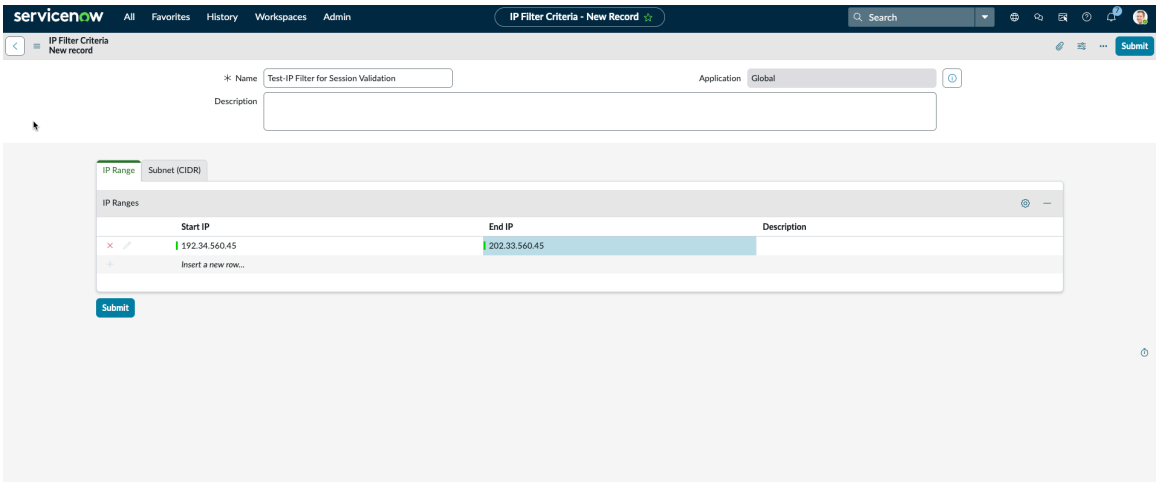
手順

1. 移動先 すべて > 適応認証 > 認証ポリシー > すべてのポリシー.
2. [ポリシー (sys_authentication_policy_list.do)] ページでセッション検証ポリシーを選択します。
3. [ポリシー入力] を選択します。

   a. [新規] または [編集] を選択します。

   b. 作成するポリシー入力 (フィルター基準) の種類を選択します。
   使用可能なオプションは、[IP]、[ロール]、および [グループフィルター基準] です。**[IP** フィルター基準] を選択します。

   c. フォームにフィルターの詳細を入力し、**IP** 範囲を入力します。



IP フィルタの作成方法の詳細については、「IP フィルター基準の作成」を参照してください 。

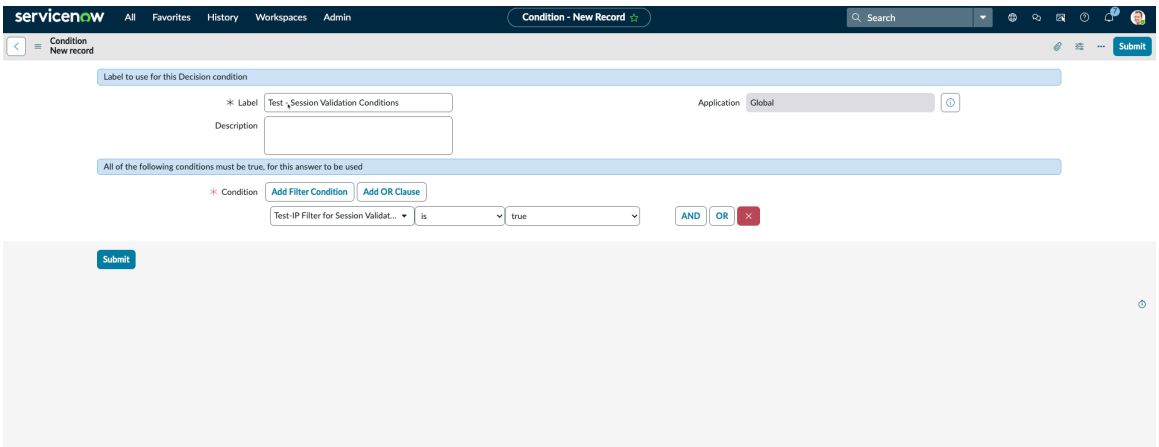
d. [送信] を選択します。

4. [セッション検証ポリシー] ページで [ポリシー条件] を選択します。

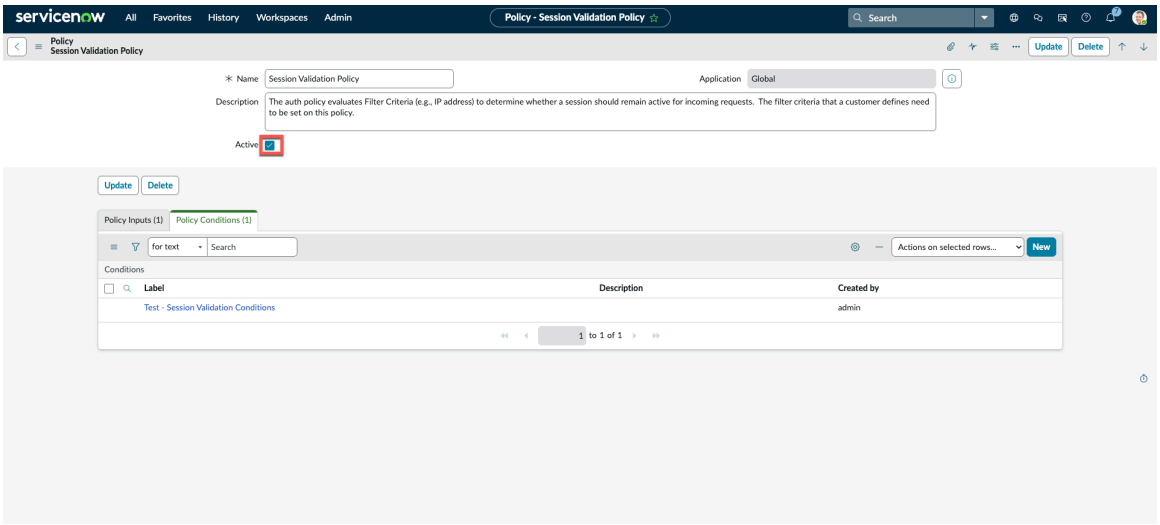
a. [新規] を選択します。

b. フォームに入力し、ポリシー入力の条件を設定します。

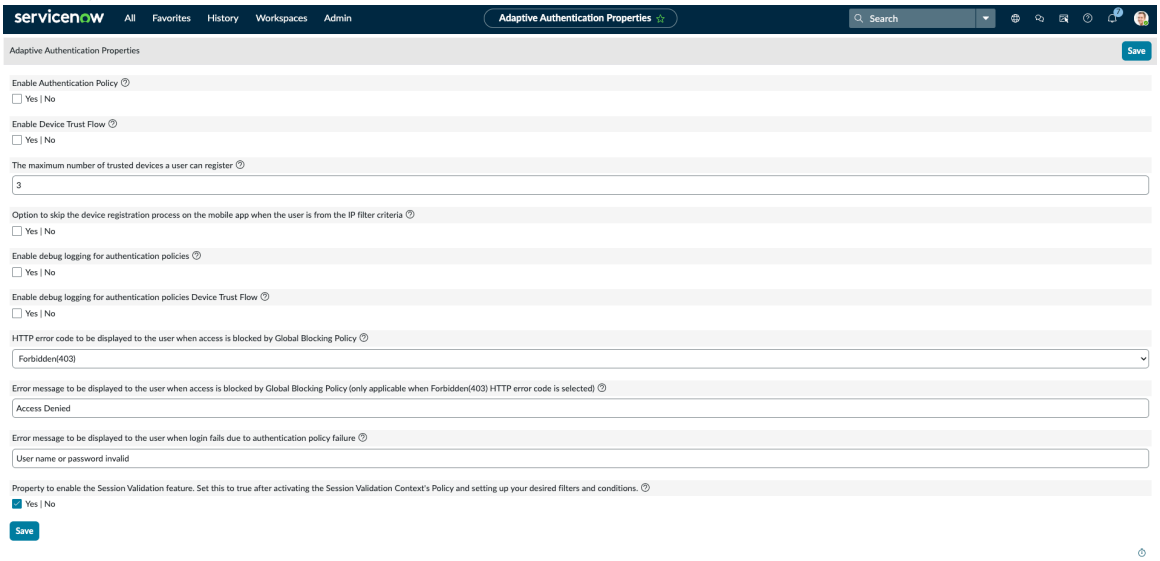
> **注:** ポリシー入力の設定に基づいて、条件を true または false に設定できます。この例では true に設定されています。この場合、条件を true に設定すると、設定された IP アドレスを持つユーザーのみがログインできます。



5. ポリシー入力と条件で [セッション検証ポリシー] を設定した後、[アクティブ] チェックボックスをオンにしてポリシーを有効にします。



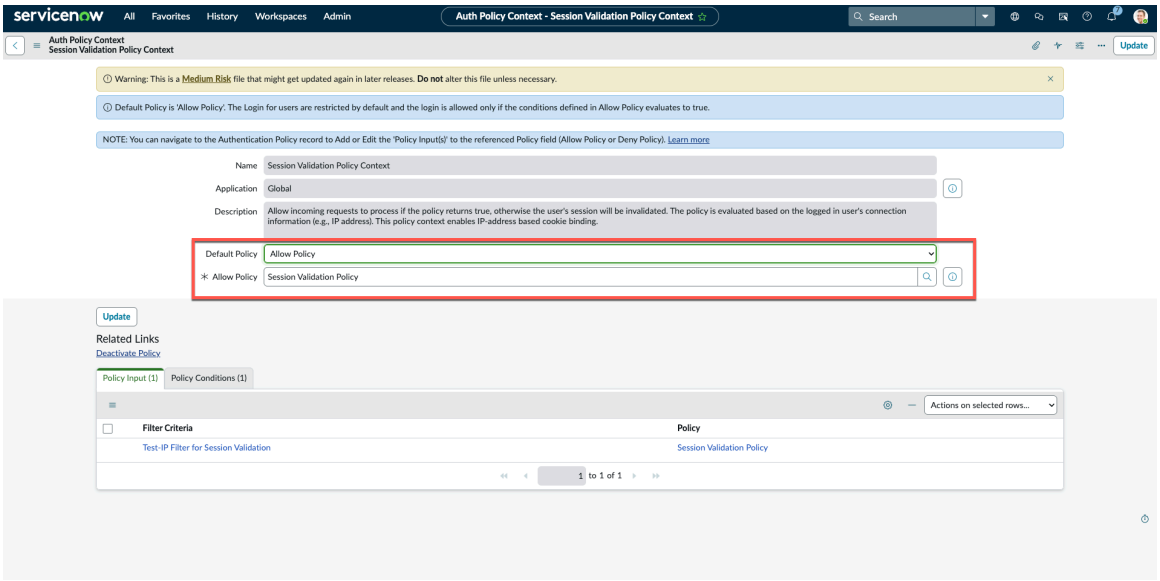
6. 移動先 すべて > 適応認証 > 認証ポリシー > プロパティ をクリックし、[セッション検証] プロパティを有効にします。



7. 移動先 すべて > 適応認証 > 認証ポリシーのコンテキスト > セッション検証コンテキスト.
8. デフォルトポリシーを [許可ポリシー] または [拒否ポリシー] に設定して、ポリシー入力とポリシー条件に従ってセッション検証コンテキストを設定します。

注: デフォルトでは、アクセス権限は以下のように設定されます。

- セッション検証コンテキストは [許可ポリシー] に設定されます。
- 許可ポリシーがセッション検証ポリシーとして選択されています。



結果
この設定では、以下に基づいてログインセッションが評価されます。

- ハイジャッカーがユーザーのセッション Cookie をあるデバイスから別のデバイスにコピーしてセッションを代理操作する際の ServiceNow インスタンスへのアクセスを制限します。
- ユーザーが安全でないネットワークを使用している場合、ユーザーのセッションアクセスを制限します。

認証ポリシー

認証ポリシーは、指定されたポリシー条件に基づいて認証要求を評価し、ポリシー条件評価の出力に応じてアクセスを許可または拒否します。たとえば、[アクセス許可ポリシー] で指定されたすべてのポリシー条件が true と評価された場合にのみ、アクセスが許可されます。

ビルトイン認証ポリシーを使用するか、セキュリティ要件に従って認証ポリシーを作成します。インスタンスのポリシーを確認するには、次に移動します: 適応認証 > 認証ポリシー > すべてのポリシー.

注: どの時点でも、[アクセスポリシーを許可] または [アクセスポリシーを拒否] のいずれかを実行できますが、両方を実行することはできません。

ポリシー	説明
アクセスポリシーを許可	デフォルトですべての認証要求を拒否します。指定されたポリシー条件に一致する認証要求のみを許可します。
アクセス事前認証ポリシーを許可	アクセス事前認証ポリシーを許可します。
ローカルログイン以外のユーザーを許可 (Allow non local login users)	ローカルログイン以外のユーザーを許可するには、このポリシーを選択します。SSO 復旧フローのコンテキストで使用されます。

ポリシー	説明
アクセスポリシーの拒否	デフォルトですべての認証要求を許可します。指定されたポリシー条件に一致する認証要求のみを拒否します。
グローバルブロックポリシー	認証前にユーザーと API のアクセス要求を拒否します。このポリシーは、IP アドレスアクセス制御の代わりに使用できます。
セッション検証ポリシー	認証ポリシーは、フィルター基準 (IP アドレスなど) を評価して、受信要求に対してセッションをアクティブのままにするかどうかを決定します。このポリシーのフィルター基準を設定する必要があります。
ローカルログイン拒否ポリシー (Local login deny policy)	すべてのローカルログインをブロックするには、このポリシーを選択します。SSO 復旧フローのコンテキストで使用されます。
ステップダウン MFA ポリシー	ユーザーが MFA 認証を必要としない場合は、デフォルトのポリシーとして [ステップダウン **MFA** ポリシー] を選択します。ステップアップ MFA ポリシーで定義されているポリシー条件が true と評価される場合、ユーザーは MFA を使用してログインする必要はありません。
ステップアップ MFA ポリシー	ステップアップ MFA ポリシーで定義されたポリシー条件が true と評価される場合、MFA 認証を要求するデフォルトポリシーとして [ステップアップ **MFA** ポリシー] を選択します。

認証ポリシーの構成

認証ポリシーを構成して、インスタンスへのアクセスを許可したり、マルチファクター認証を適用したりするために使用する入力と条件を定義します。

始める前に

必要なロール：admin

手順

1. 移動先 すべて > 適応認証 > 認証ポリシー > すべてのポリシー.

> **注:** 完成したポリシーの例として、次のポリシーをインスタンスで確認できます。
>
> - デモ POLICY - アドミンのローカルログインを信頼できる IP 範囲からのみ許可
> - デモ POLICY - アドミンのローカルログインのみ許可
> - デモ POLICY - ユーザー名およびパスワードベースの認証を特定のユーザーに制限

2. [新規] ボタンをクリックして、新しいポリシーレコードを作成します。

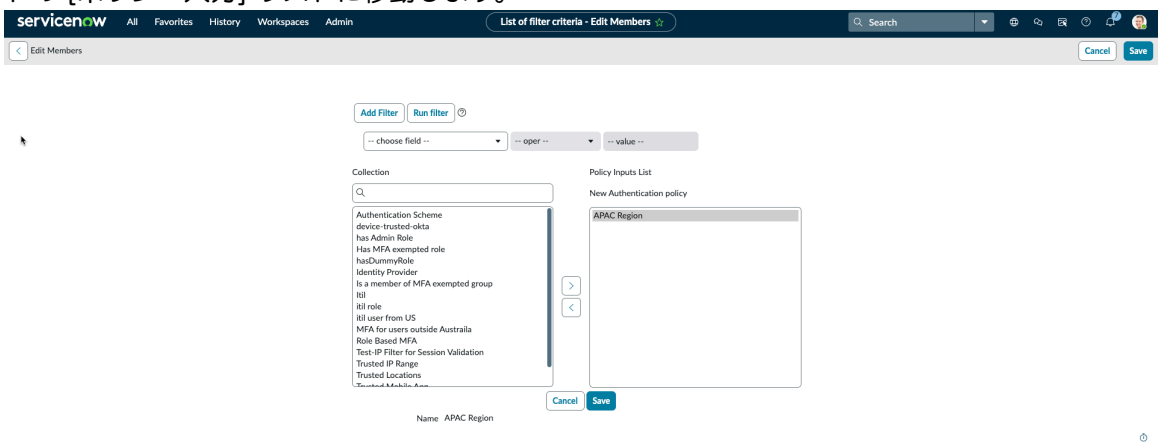
3. [ポリシー] フォームの各フィールドに入力します。

ポリシーフォーム

フィールド	説明
名前	ポリシーの名前。

フィールド	説明
アプリケーション	ポリシーのスコープ対象のアプリケーション。このフィールドには現在のアプリケーションが自動的に入力されます。
説明	ポリシーの説明
アクティブ	ポリシーがアクティブかどうか。

4. [ポリシーの入力] タブで、[編集] をクリックします。
5. [コレクション] リストから 1 つ以上のフィルター基準を選択し、[アクセスポリシーを許可] リストの [ポリシー入力] リストに移動します。



> **注:** このセクションで使用する独自のフィルター基準を作成する方法については、「フィルター基準」を参照してください。

6. [ポリシー条件] タブで、[新規] をクリックします。
7. フォームのフィールドに入力します。

[条件] フォーム

フィールド	説明
ラベル	条件を識別する名前
説明	条件の説明
条件	認証要求を評価するために使用される複数のポリシー入力 (フィルター基準) の論理的な組み合わせ。たとえば、信頼できる IP アドレスのリストから請負会社のみを許可するような条件を作成できます。

8. オプション: 追加のポリシー条件を作成するには、手順 7 を繰り返します。

> **注:** 複数のポリシー条件を作成する場合、アクセスポリシーの最終的な出力は、すべてのポリシー条件の論理 OR 出力によって決まります。これは、ポリシー条件のいずれかが満たされると、ポリシーが true と評価されることを意味します。

9. [保存] をクリックします。

認証ポリシーを認証ポリシーコンテキストに追加

認証ポリシーを、認証ポリシーコンテキストのいずれかに追加します。認証コンテキストは、ポリシー入力と条件を使用して、ユーザーによるインスタンスへのアクセスが許可されているかどうか、またはユーザーに MFA が適用されているかどうかを判断します。

始める前に

必要なロール：admin

手順

1. 移動先 すべて > 適応認証 > 認証ポリシーのコンテキストをクリックし、必要に応じていずれかのエントリを選択します。

   事前認証のコンテキスト

   事前承認コンテキストを使用して、ユーザーにログイン画面が表示される前にポリシーを評価します。ユーザーは、この評価に基づいてアクセスを許可または拒否されます。

   認証後のコンテキスト

   ユーザーがログイン認証情報を入力した後、承認後のコンテキストを使用してポリシーを評価します。ユーザーは、この評価に基づいてアクセスを許可または拒否されます。この評価はインスタンスがユーザーを識別した後に行われるため、このコンテキストのポリシーは、ロールやグループなどのユーザーデータに基づいて評価を行うことができます。

   MFA コンテキスト

   MFA コンテキストを使用して、ユーザーがログイン時にマルチファクター認証を使用する必要があるかどうかを判断します。

   セッション検証コンテキスト

   セッション検証コンテキストを使用して、フィルターとして定義された条件に基づいて設定された IP アドレスを評価し、セッション内のインスタンスへのアクセスを許可します。

2. [デフォルトポリシー] フィールドで、値を選択します。
   このフィールドの値は、ポリシーの条件の結果をコンテキストでどのように使用するのかを決定します。

   このフィールドで使用できるオプションは、選択したコンテキストによって異なります。これらのコンテキストの詳細については、「認証ポリシーのコンテキスト」を参照してください。

3. コンテキストにポリシーをアサインします。
   フィールドの名前は、選択したコンテキストによって異なります。

   認証前および認証後のコンテキスト

   これらのコンテキストには、[デフォルトポリシー] フィールドの選択に応じて [拒否ポリシー] または [許可ポリシー] フィールドがあります。

   MFA コンテキスト

   このコンテキストには、[デフォルトポリシー] フィールドの選択に応じて [ステップアップ **MFA** ポリシー] または [ステップダウン **MFA** ポリシー] フィールドがあります。

   セッション検証コンテキスト

   このコンテキストには、[デフォルトポリシー] フィールドの選択に応じて [許可ポリシー] または [拒否ポリシー] フィールドがあります。

4. [更新] をクリックします。
   レコードを更新した後、[ポリシー入力] タブと [ポリシー条件] タブでポリシーの入力と条件を確認できます。

適応認証イベント

適応認証イベントテーブルを使用して、イベントについて確認できます。

以下に適応認証イベントテーブルで追跡できるログの一部を示します。

- glide.adaptive.auth.log.success.event：認証後、API ログインの success イベントを確認します。
- glide.adaptive.auth.log.mfa.relax.event：MFA コンテキストの mfa_relaxation イベントを確認します。

注:

- 認証前コンテキストの場合、つまりタイプ「pre_auth」の REST API アクセスポリシー：デフォルトでは、Failure イベントがログに記録されます。Success イベントはログに記録できません。
- 認証後コンテキストの場合、つまり「pre_auth」以外のタイプの REST API アクセスポリシー：デフォルトでは Failure イベントがログに記録されます。Success イベントをログに記録するには、glide.adaptive.auth.log.success.event プロパティを有効にする必要があります。
- MFA の場合：デフォルトでは、MFA Enforcement イベントがログに記録されます。
- MFA Relaxation イベントをログに記録するには、glide.adaptive.auth.log.mfa.relax.event プロパティを有効にする必要があります。MFA Relaxation がログに記録するイベントは、MFA Enforcement と同じですが、Result フィールドに False が記録されます。

適応認証プロパティの設定

適応認証をアクティブ化した後、セキュリティ要件に従って適応認証プロパティを設定します。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > 適応認証 > 認証ポリシー > プロパティ.
2. 次のプロパティを設定します。

適応認証プロパティ

プロパティ	説明	値
認証ポリシーの有効化 (glide.authenticate.auth.policy.enabled)	認証ポリシーを有効にするオプション	◯| #
認証ポリシーのデバッグログ記録の有効化 (glide.authenticate.policy.debug)	認証ポリシーのデバッグログ記録を有効にするオプション	◯| #
グローバルブロックポリシーによってアクセスがブロックされたときにユーザーに表示される HTTP エラーコード (glide.authenticate.global.blocking_policy.error_code)	グローバルブロックポリシーによってユーザーのログインがブロックされている場合にユーザーに表示される HTTP エラーコード (HTTP	次から選択します。 ◦ Forbidden(403) ◦ Not Found(404)

プロパティ	説明	値
	error code that displays during login when the Global Blocking Policy blocks a user login.)	
グローバルブロックポリシーによってアクセスがブロックされたときにユーザーに表示されるエラーメッセージ (Forbidden (403) HTTP エラーコードが選択されている場合にのみ適用されます) (glide.authenticate.global.blocking_policy.error_message)	グローバルブロックポリシーがアクセスをブロックするときに表示されるエラーメッセージ。	テキストフィールド
デバイス信頼フローの有効化 (glide.authenticate.preauth.allow.trusted.device)	信頼できるデバイスフローを有効にするオプション。	◯| #
ユーザーが登録可能な信頼できるデバイスの最大数 (glide.trusted.device.max.count)	ユーザーが登録可能な信頼できるデバイスの最大数です。	テキストフィールド
ユーザーが信頼できるネットワークからアクセスしている場合、デバイス信頼フローでのデバイス登録をスキップ (glide.authenticate.preauth.skip.user.registration)	ユーザーが信頼できるネットワークから登録しようとしている場合に登録をスキップするオプション	◯| #
セッション検証機能を有効にするプロパティ。セッション検証機能を有効にするプロパティ。セッション検証コンテキストのポリシーをアクティブにし、必要なフィルターと条件を設定した後に、これを true に設定します (session.validation.enabled)。	セッション検証機能を有効にするオプション。セッション検証機能を有効にするプロパティ。セッション検証コンテキストのポリシーをアクティブにし、必要なフィルターと条件を設定した後に、これを true に設定します。	◯| #

チュートリアル：適応認証の設定

次の手順例を使用して、インスタンスで適応認証を設定します。

このチュートリアルを使用するには、適応認証がアクティブ化されているインスタンスが必要です。このプロセスの詳細については、「適応認証のアクティブ化」を参照してください。

この例では、新しいポリシーを作成してインスタンスに適用する手順を示します。このチュートリアルでは、次のことを実行します。

フィルター基準レコードの作成

ポリシーの入力として使用するグループフィルター基準レコードを作成します。このレコードにより、ポリシーはユーザーのグループに基づいてアクセスを決定できます。これらの手順では、ポリシーがアクセスを決定するために使用するグループを定義します。

ポリシーの作成

ユーザーがインスタンスにアクセスできるかどうかを決定するポリシーを作成します。このポリシーでは、作成するグループフィルター基準レコードを入力として使用します。これらの手順では、ポリシーがポリシー入力を使用してユーザーアクセスを決定する方法を定義するポリシー条件も定義します。

ポリシーコンテキストの設定

新しいポリシーを使用する認証後ポリシーのコンテキストを設定します。設定すると、インスタンスは、フィルター基準レコードで定義されたグループ内のユーザーのアクセスを拒否します。

フィルター基準レコードの作成

適応認証ポリシーのポリシー入力として使用する基準レコードの作成方法を説明します。

始める前に

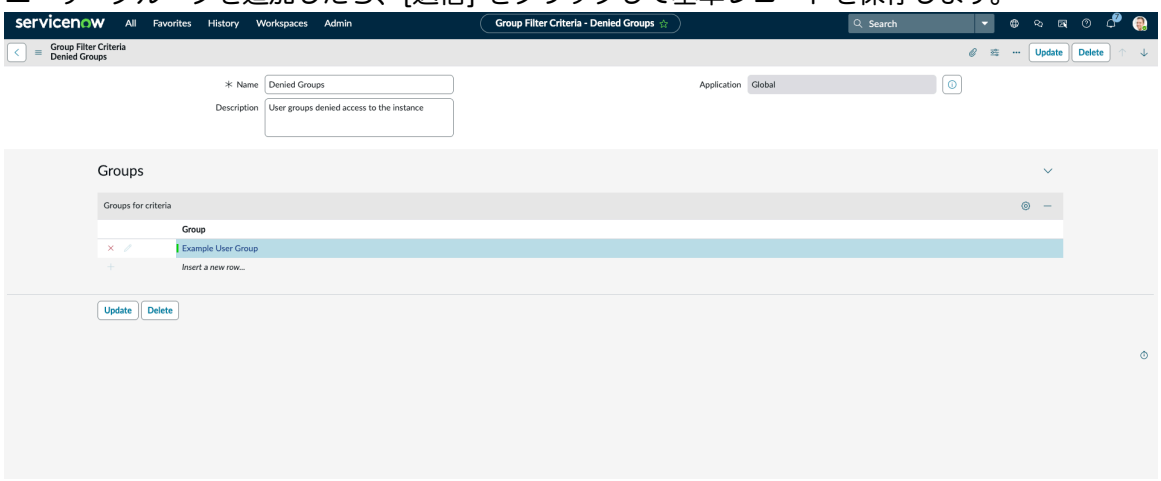
必要なロール：admin

ユーザーグループに基づいてインスタンスへのアクセスを拒否するには、グループフィルター基準レコードを作成する必要があります。このレコードは、ポリシーでアクセスを許可または拒否できるユーザーグループまたはユーザーグループセットを定義します。この例では、単一のユーザーグループのグループフィルター基準レコードを作成します。

ユーザーグループとインスタンスでの使用方法の詳細については、「ユーザー管理の詳細」を参照してください。

手順

1. 移動先 すべて > 適応認証 > フィルター基準 > グループクライテリア.
2. [新規] をクリックして、新しいレコードを作成します。
3. [名前] フィールドに、レコードの名前を入力します。
   例：拒否グループ。
4. [説明]フィールドに、簡単な説明文を入力します。
   例：インスタンスへのアクセスを拒否するユーザーグループ。
5. [基準のグループ] リストで、[新規行を挿入...] をダブルクリックします。
6. ユーザーグループの名前を入力するか、参照アイコン ( ) をクリックしてリストからグループを選択します。
   フィルター基準の新しいユーザーグループを作成する場合は、参照アイコン ( ) をクリックしてから、[新規] ボタンをクリックします。ユーザーグループの作成の詳細については、「ユーザーグループの作成」を参照してください。
7. ユーザーグループを追加したら、[送信] をクリックして基準レコードを保存します。



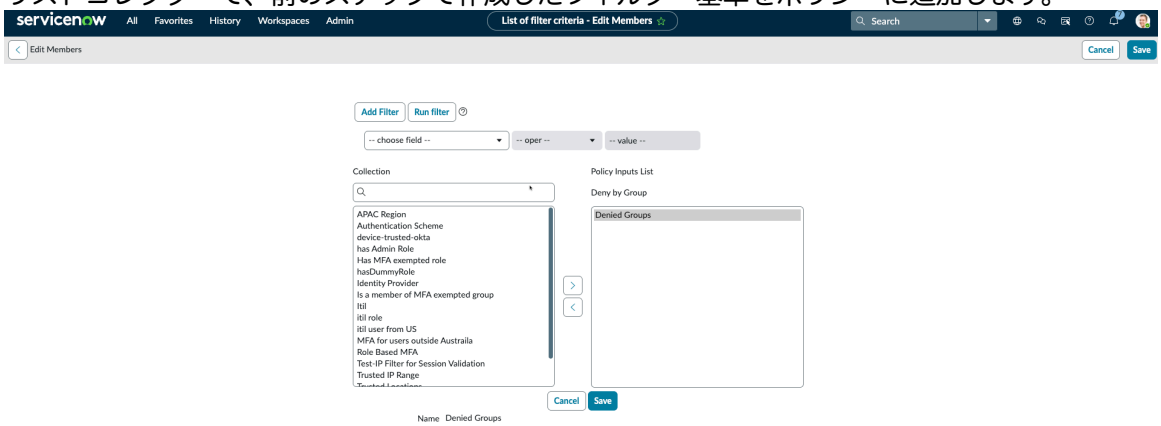
ポリシーの作成

グループフィルター基準で定義されたユーザーグループへのアクセスを拒否するポリシーを作成する方法について説明します。

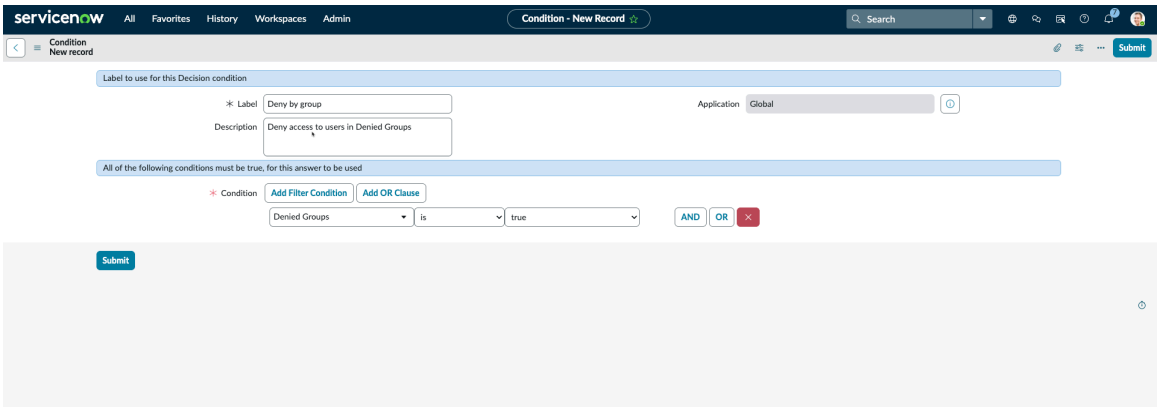
始める前に
必要なロール：admin

手順

1. 移動先 すべて > 適応認証 > 認証ポリシー > すべてのポリシー.
2. [ポリシー] リストで、[新規] をクリックします。
3. [名前] フィールドに、レコードの名前を入力します。
   例：拒否グループ。
4. [説明]フィールドに、簡単な説明文を入力します。
   例：インスタンスへのアクセスを拒否するユーザーグループ。
5. フォームヘッダーを右クリックし、[保存] をクリックします。
   保存した後、[ポリシー入力] および [ポリシー条件] リストがフォームに表示されます。
6. [ポリシーの入力] リストで、[編集] をクリックします。
7. リストコレクターで、前のステップで作成したフィルター基準をポリシーに追加します。



8. [保存] をクリックします。
   インスタンスはポリシー入力を保存し、ポリシーレコードを再度表示します。
9. [ポリシー条件] 関連リストで、[新規] をクリックします。
10. [ラベル] フィールドで、この条件のラベルを作成します。例：拒否グループ。
11. [説明]フィールドに、簡単な説明文を入力します。
    例：拒否グループのユーザーへのアクセスを拒否。
12. [条件] フィールドで、前の手順で作成したフィルター基準を選択し、[次の値に等しい **(=)**] と **[true]** を選択して条件を完成させます。



13. [保存] をクリックします。

ポリシーコンテキストの設定

新しいポリシーを使用する認証後ポリシーのコンテキストを設定します。設定すると、インスタンスでは、フィルター基準レコードで定義されたグループにいるユーザーへのアクセスを拒否します。

始める前に
必要なロール：admin

手順

1. 適応認証 > 認証ポリシーのコンテキスト > 認証後のコンテキスト.
   このチュートリアルで使用するポリシーはユーザーがどのグループに属しているかを評価する必要があるため、認証後コンテキストを使用する必要があります。この情報は、認証情報を入力した後にのみ使用できます。
2. [デフォルトポリシー] フィールドで、[拒否ポリシー] を選択します。
   ここでは、デフォルトでユーザーにアクセス権が付与され、ポリシー条件が true と評価された場合にのみアクセスを拒否します。
3. [拒否ポリシー] フィールドで、これまでのステップで作成した通知を選択します。
4. [更新] をクリックして、ポリシーコンテキストレコードを保存します。

信頼できるモバイルアプリ向け適応認証 (Adaptive Authentication for Trusted Mobile Apps)

信頼できないネットワークから ServiceNow にアクセスする際は、Now Mobile アプリを使用します。

信頼できるモバイルアプリ向け適応認証 (Adaptive Authentication for Trusted Mobile Apps)

信頼できるモバイルアプリ向け適応認証により、ユーザーは Now Mobile アプリ を使用して ServiceNow インスタンスにアクセスできます。インスタンスは、信頼できる IP ネットワークの境界の内側で保護されます。

ネットワーク外にいるときにインスタンスにアクセスする必要があるシナリオとしては、次のようなものがあります。

従業員がネットワーク外から Now Mobile アプリ および 従業員サービスセンター (ESC) (ポータル) にアクセスする必要がある場合。信頼できるモバイルアプリのフィルターを使用すると、ユーザーアカウントにリンクされている信頼できる Now Mobile アプリ からの受信要求を識別できます。

アドミンは、ポリシーを設定できます。このポリシーによって、ユーザーが自分のモバイルデバイスを登録し、信頼できるネットワークからインスタンスにアクセスすることが可能になります。

ユーザーは、インスタンスに表示されている QR をスキャンしてモバイルデバイスを登録できます。登録後、認証情報を使用して Now Mobile アプリからインスタンスにログインできます。

> **注:** モバイルデバイスを登録するには、信頼できるネットワークに接続している必要があります。登録を済ませれば、他のネットワークからもインスタンスにログインできます。

特徴

アドミンは、次の機能を使用できます。

- 信頼できるアプリのフィルター基準を使用してポリシーを設定する。
- 信頼できるモバイルアプリの新しいフィルター基準を使用してポリシー条件を作成する。
- ユーザーの信頼できるモバイルアプリにおけるすべてのログイン方法 (ローカルログイン、SAML、OIDC、および MFA) をサポートする。
- 信頼できるデバイスを取り消す。
- デバイス登録や取り消しのすべてのアクションに対するセキュリティイベントを追加する。追加したイベントを使用して、ユーザーに通知できます。
- 署名の失敗、Cookie の検証の失敗、無効なトークン、無効なヘッダー、またはクエリパラメーターに関するセキュリティイベントに対応する。
- 登録されたデバイスの最大数を制御する。
- ユーザーが既存のペアリングされたデバイスを削除しない限り、新しいデバイスを登録できないようにする。
- デバイスが最後に使用された時間をキャプチャする。

信頼できるモバイルアプリのアクティブ化

認証ポリシーとフィルター条件を使用して、信頼できるモバイルアプリを使用した適応認証を有効にします。

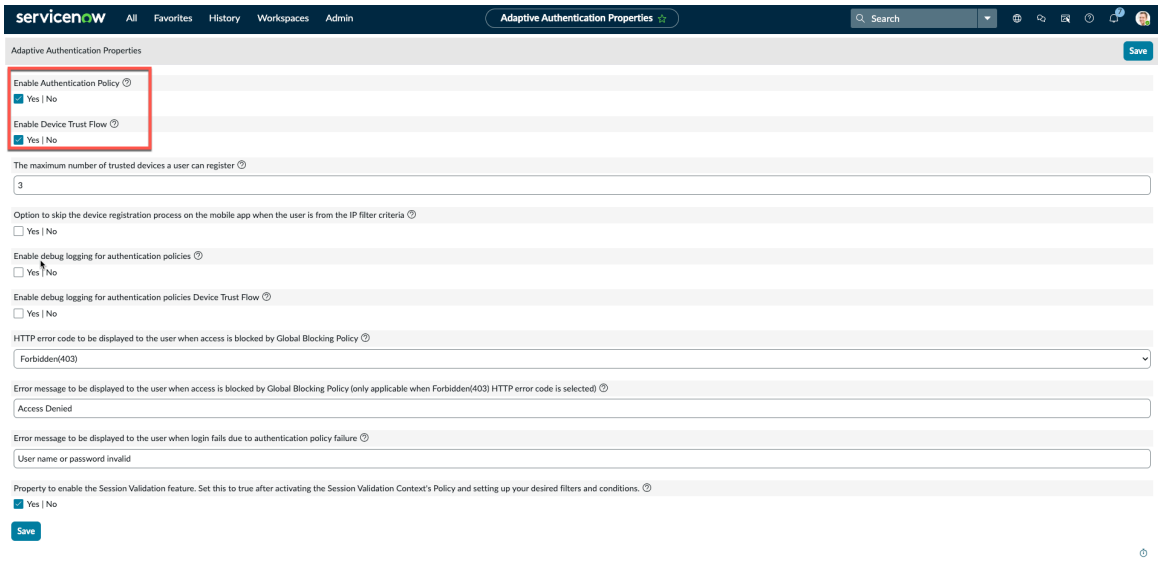
始める前に

適応認証 (*com.snc.adaptive_authentication*) プラグインがインストールされていることを確認します。

必要なロール：admin

手順

1. 移動先 すべて > 適応認証 > 認証ポリシー > プロパティ.
2. [適応認証プロパティ] ページで、次のプロパティを有効にします。
   - 認証ポリシーの有効化 (Enable Authentication Policy) (*glide.authenticate.auth.policy.enabled*)
   - デバイス信頼フローの有効化 (*glide.authenticate.preauth.allow.trusted.device*)



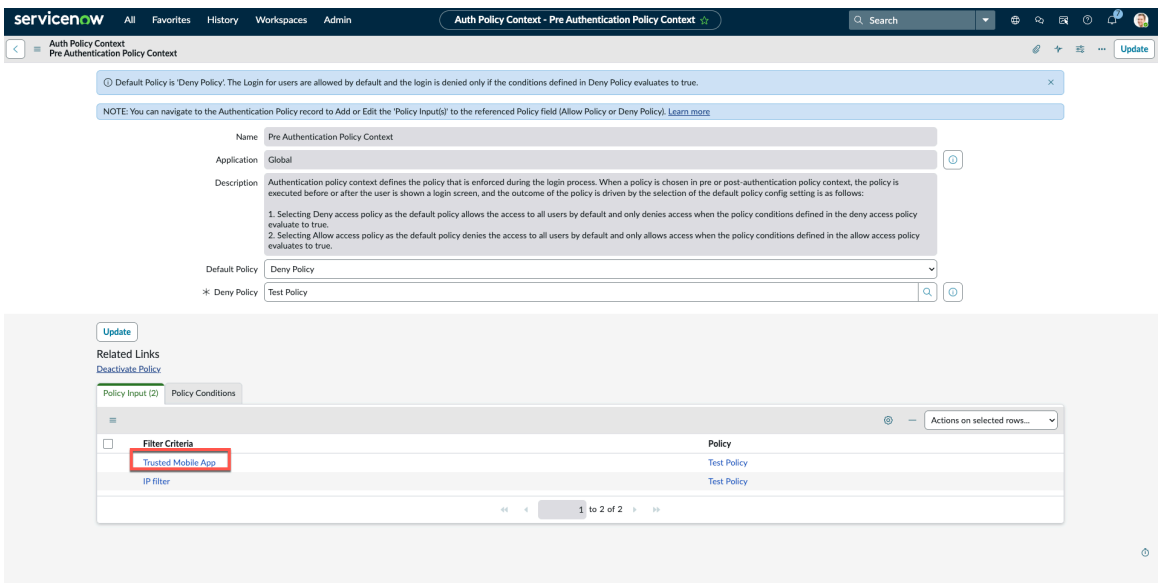
注: デバイス信頼フローのプロパティを無効にするには、信頼できるモバイルフィルターを使用して条件を削除する必要があります。そうしないと、条件の削除を要求するエラーメッセージが表示されます。

3. 移動先 すべて > 適応認証 > 認証ポリシーのコンテキスト > 事前認証のコンテキスト.
4. 事前認証のコンテキストで条件を定義します。
   詳細については、「事前認証コンテキスト」を参照してください。

   注: デフォルトでは、ポリシー条件は [拒否ポリシー]です。[許可ポリシー **(Allow Policy)**] に変更できます。この 2 つのポリシーは正反対です。

   - [許可ポリシー **(Allow Policy)**] では、デフォルトですべてのユーザーがアクセスを拒否され、アクセス許可ポリシー条件が true の場合のみアクセスが許可されます。
   - [拒否ポリシー] を使用すると、デフォルトですべてのユーザーにアクセスが許可され、アクセス拒否ポリシー条件が true の場合のみアクセスが拒否されます。

   [ポリシー入力] では、[信頼できるモバイルアプリ] が、信頼できるモバイルアプリのポリシー入力となります。

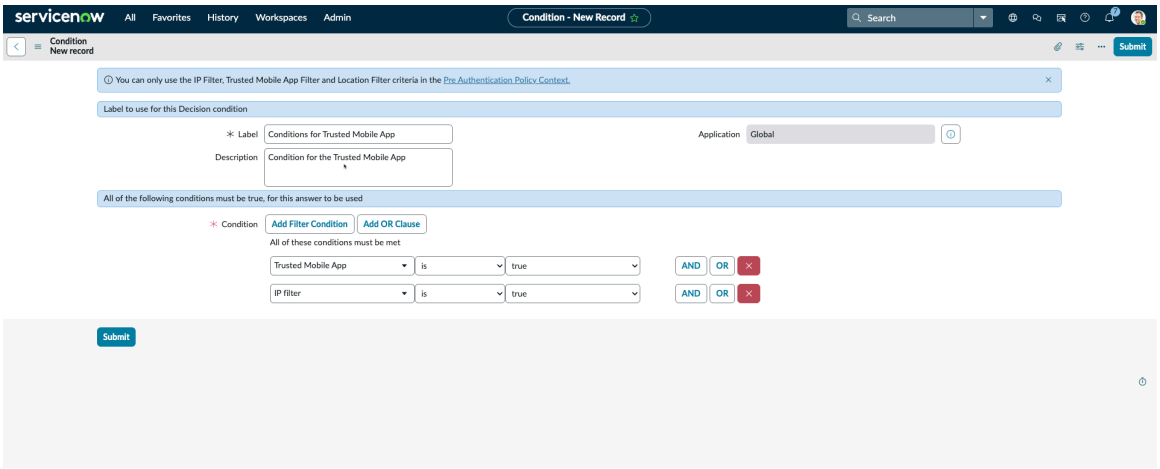


5. [ポリシー条件] で、[新規] をクリックして条件を作成します。

6. フォームの各フィールドに入力します。

[フィルター条件] フォーム

フィールド	説明
ラベル	条件の名前
説明	条件の説明
アプリケーション	このレコードのアプリケーションスコープ。
条件	AND および OR に基づく条件。認証ポリシーが [許可ポリシー **(Allow Policy)**] であるため、この画像の例では [信頼できるモバイルアプリ] の条件が true に設定されています。



7. [送信] をクリックします。

結果
[信頼できるデバイス (Trusted Device)] 機能に対して、ポリシー入力とフィルター条件が作成されます。ユーザーは [信頼できるデバイス (Trusted Device)] 機能を使用して、信頼できないネットワークから Now Mobile アプリ を使用して ServiceNow インスタンスにアクセスできます。詳細については、「信頼できるデバイスを登録する」を参照してください。

信頼できるデバイスを登録する

ネットワーク外から ServiceNow インスタンスにアクセスできるように、信頼できるデバイスを登録します。

始める前に
信頼できるデバイスの登録を実行するには、信頼できるネットワークに入っている必要があります。

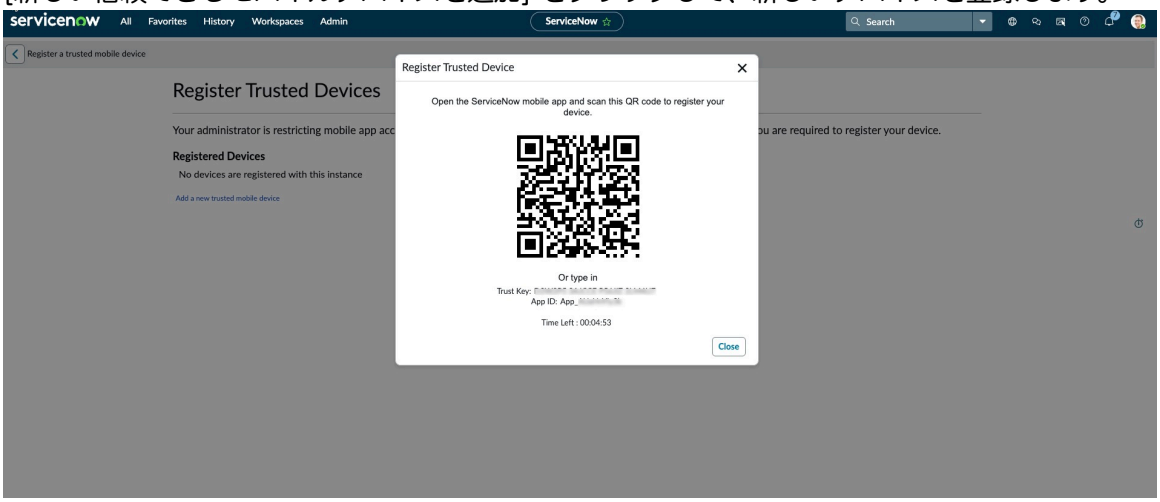
必要なロール：なし

手順

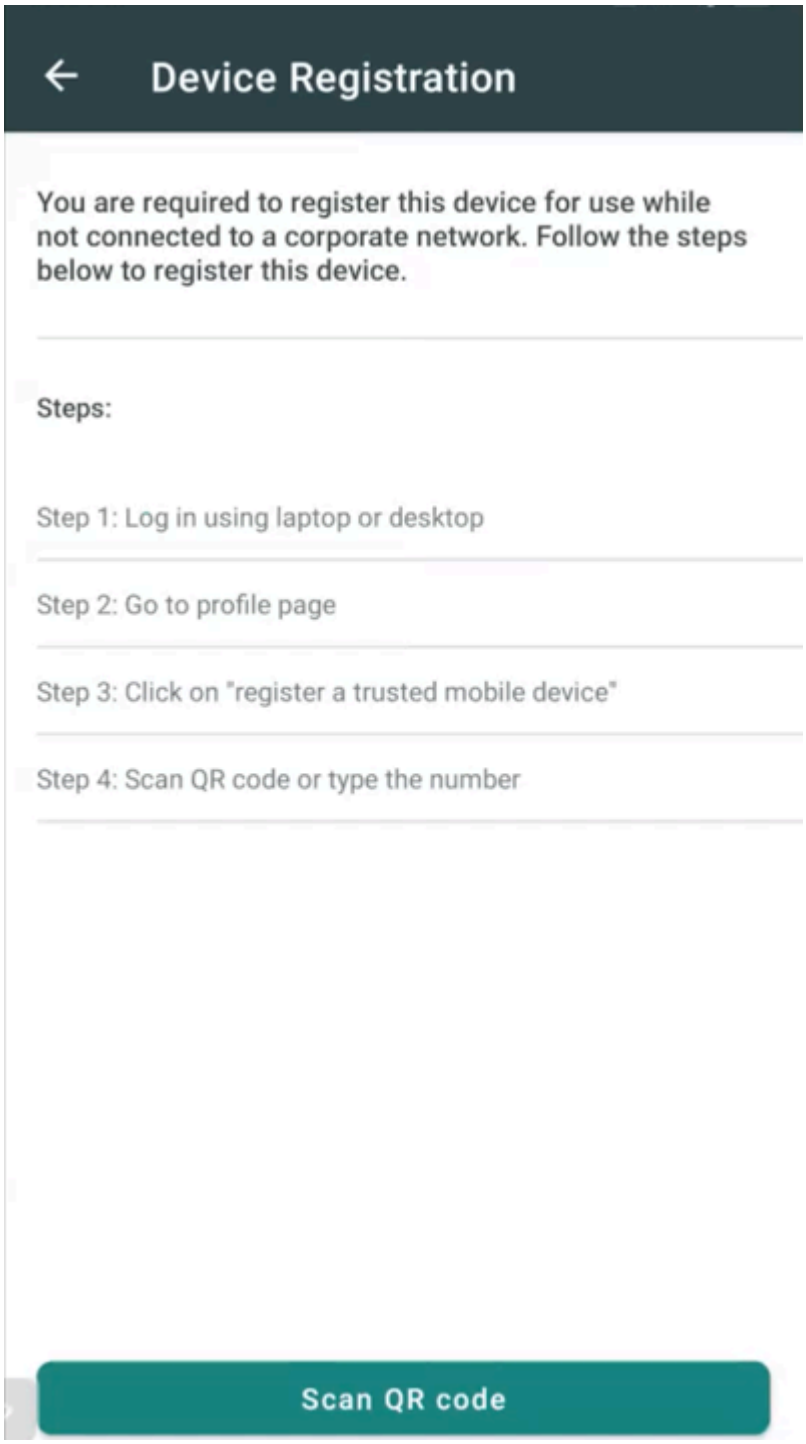
1. 次のいずれかのメニューオプションに移動します。
   - ServiceNow AI Platformで、 すべて > セルフサービス > プロファイル.

     注: インスタンスヘッダーで自分のユーザー名をクリックして、プロファイルにアクセスすることもできます。

   - Now Supportでプロファイルをクリックし、[信頼できるデバイス **(Trusted Device)]** を選択します。
2. ユーザープロファイルで、[関連リンク] セクションの [信頼できるモバイルデバイスを登録] をクリックします。
   [信頼できるデバイスを登録] ページが表示されます。
3. [新しい信頼できるモバイルデバイスを追加] をクリックして、新しいデバイスを登録します。



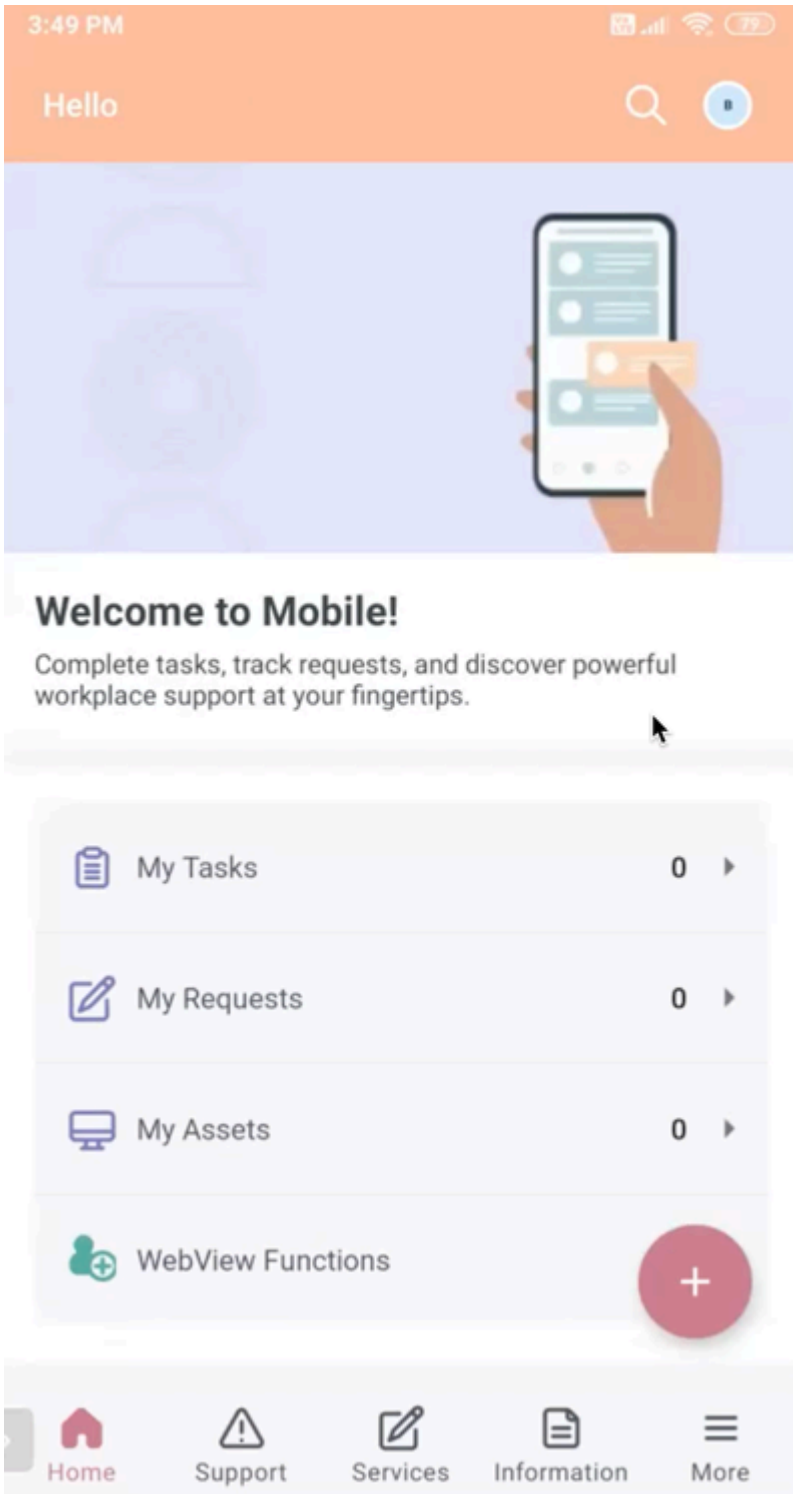
4. ServiceNow モバイルアプリの [デバイス登録 (Device Registration)] 画面で、**[QR** コード をスキャン] ボタンをタップし、ラップトップまたはデスクトップに表示されている QR コードをスキャンします。



登録プロセスが完了し、認証を完了するためのログインページが表示されます。

5. 認証情報を指定して Now Mobile アプリ にログインします。

モバイルホーム画面が表示されます。



結果
ラップトップまたはデスクトップの [信頼できるデバイスを登録] ページに、登録されたデバイスが表示されます。 アイコンを使用すると、登録されたデバイスをページから削除できます。

次のタスク
移動先 すべて > 適応認証 > デバイス信頼 > デバイス登録 をクリックして、登録されたデバイスのすべての詳細を表示します。

信頼できるデバイスの管理

信頼できるデバイス登録ページから、信頼できるデバイスを管理します。

始める前に
必要なロール：なし

手順

1. 次のいずれかのメニューオプションに移動します。
   - ServiceNow AI Platformで、 すべて > セルフサービス > プロファイル.

     注: インスタンスヘッダーで自分のユーザー名をクリックして、プロファイルにアクセスすることもできます。

   - Now Supportでプロファイルをクリックし、[信頼できるデバイス **(Trusted Device)]** を選択します。
2. ユーザープロファイルで、[関連リンク] セクションの [信頼できるデバイスを登録] をクリックします。
   [信頼できるデバイスを登録] ページが表示されます。
3. [信頼できるデバイスを登録] ページで、 アイコンをクリックしてデバイスを削除します。

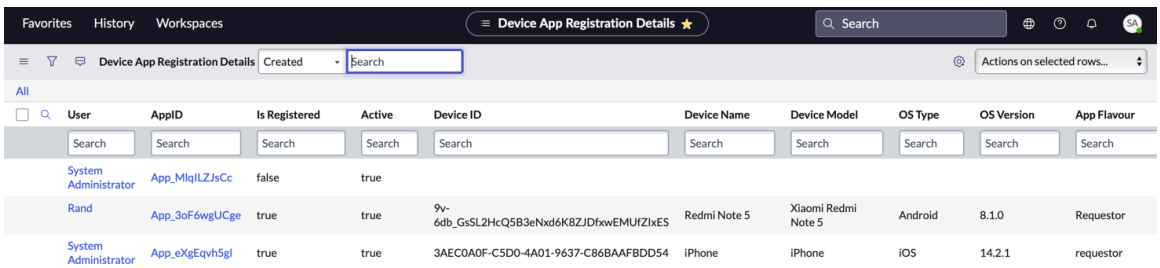
   注: 削除されたモバイルデバイスは、再登録しないとインスタンスにアクセスできません。

登録済みデバイスの登録の詳細

ServiceNow インスタンスに登録されているデバイスの詳細を確認します。

インスタンスに登録されているデバイスのすべての詳細を表示するには、次に移動します: すべて > 適応認証 > デバイス信頼 > デバイス登録.

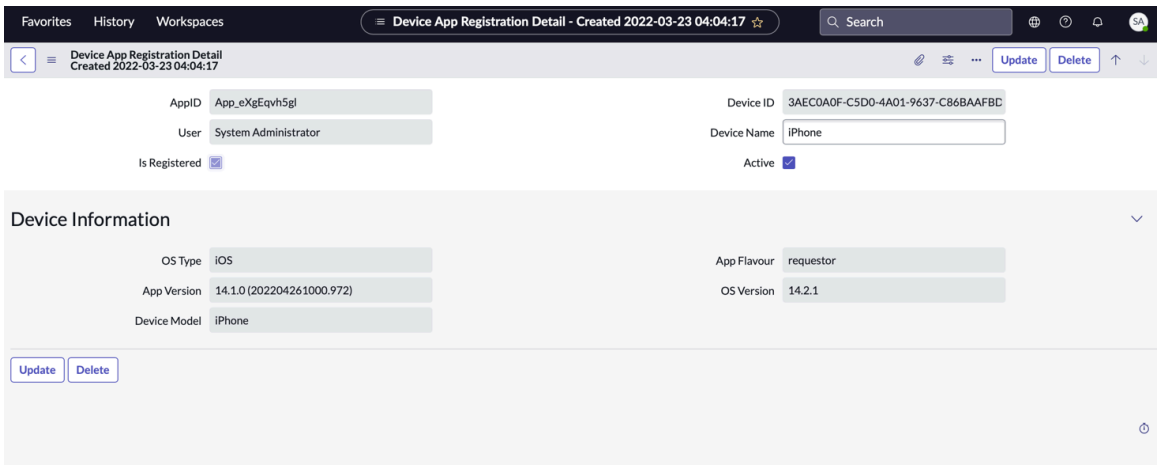
フィルターを使用して当該のデバイスを識別します。**[AppID]** フィールドをクリックすると、登録されたデバイスの詳細が表示されます。



表示されるデバイスの詳細情報は次のとおりです。

- AppID
- デバイス ID
- ユーザー
- デバイス名

- デバイス情報：
  - OS タイプ
  - アプリフレーバー
  - アプリバージョン
  - OS バージョン
  - デバイスモデル



信頼できるモバイルアプリのトラブルシューティング

信頼できるモバイルアプリの問題を解決するには、次のトラブルシューティングシナリオを確認してください。

以下のセクションで、いくつかのトラブルシューティングのシナリオとその方法について説明します。

インスタンスとモバイルに時差 (クロックスキュー) がある

クロックスキューがある場合、信頼できるデバイスを使用するにはモバイルデバイスのクロックを調整またはリセットするとよいでしょう。

QR コードが 5 分以内に期限切れになる

信頼できるデバイス機能の QR コードは 5 分以内で期限切れになります。このような場合は、クリックすると新しい QR コードを取得できます。

注: QR コードの有効期限は延長できます。コードの有効期限を延長するには、アドミンにお問い合わせください。

システムプロパティが変わっている

システムプロパティが変わっている場合は、ログインに使用できる有効な認証情報のシステムプロパティに基づいて登録を確認します。

登録デバイス数が最大に達した

デバイス登録の最大数に達した場合は、既存のデバイスを削除するか、アドミンに連絡して最大デバイス数を変更してください。

API 認証

API の認証構成。

ServiceNow® の API ベースの認証は、インスタンスにアクセスするユーザーの ID を検証し、ServiceNow® インスタンスに API コールを行う間にユーザーのロールまたは職務に一致する機能をユーザーに許可します。

ServiceNow® の API 認証のタイプは次のとおりです。

- 証明書ベースの認証
- OAuth
- トークンベースの認証

証明書ベースの認証

証明書ベースの認証では、信頼できる認証局 (CA) からの証明書を使用して、受信 API 要求を相互に認証できます。

受信 Web サービスの証明書ベースの認証

ServiceNow SOAP および REST API への受信要求を認証します。受信 Web サービスの相互認証を設定するには、「証明書ベースの認証の設定」を参照してください。

OAuth

OAuth ベースの認証は、認証プロトコルを使用してシステムで信頼を確立しようとするクライアントの ID を検証します。

OAuth 2.0 - オープン認証は認証の業界標準プロトコルであり、Web アプリケーション、デスクトップアプリケーション、およびモバイルデバイス向けの特定の認証フローを提供すると同時に、クライアント開発者の簡素化を実現します。

インバウンド

インスタンスにアクセスする外部クライアント用のエンドポイントを作成し ます。これにより、OAuth クライアントアプリケーションレコードが作成され、クライアントがインスタンスの制限付きリソースにアクセスするために必要なクライアント ID とクライアントシークレットが生成されます。詳細については、「OAuth 受信」を参照してください。

トークンベースの認証

API キーまたは HMAC を使用する受信 REST API 構成のトークンベース認証。

ServiceNow® でユーザー認証できるように、REST API エンドポイントの API トークンをサポートします。

受信 REST API 構成のトークンベース認証は、API キーまたは HMAC トークンを使用して ServiceNow® インスタンス上で実行できます。

受信 REST API の API キーと HMAC 認証

ServiceNow® ユーザー名とパスワードが Webhook URL に表示されないように、REST API エンドポイントの API トークンをサポートします。

API キーベースの認証を有効にして、受信 Webhook URL を安全に認証します。

API キーと HMAC 認証を使用するには、ServiceNow® インスタンスにプラグイン (com.glide.tokenbased_auth) をインストールする必要があります。

⚠ 警告: サーバーに機密情報を送信する場合は、**POST** 要求を使用します。

API キーと HMAC 認証のインストールには、次のプラグインの依存関係があります。

- REST API Auth Scope プラグイン (com.glide.rest.auth.scope)
- REST API アクセスポリシープラグイン (com.glide.rest.policy)
- 認証スコープ (com.glide.auth.scope)

メリット

受信 REST API の API キーと HMAC 認証により以下が有効になります。

- REST API 認証用の API キーまたは HMAC トークンを指定する機能。
- ユーザーアカウントを API キーまたは HMAC トークンに関連付ける機能。
- REST API コール内でトークンをクエリパラメーターまたはヘッダーとして指定する機能。
- 認証スコープを API キーまたは HMAC トークン構成に関連付けて、API キーを使用して特定のスコープに関連付けられた API を呼び出す機能。
- API キーまたは HMAC トークン構成を、API アクセスポリシーで使用できる認証プロファイルに関連付ける機能。

API キーと HMAC® 認証のアクティブ化

ServiceNow インスタンスでプラグイン API キーと HMAC 認証 (com.glide.tokenbased_auth) をアクティブ化することができます。

始める前に
必要なロール：admin。

手順

1. 移動先 すべて > システムアプリケーション > 利用可能なすべてのアプリケーション > すべて.
2. フィルター基準と検索バーを使用して、API キーと HMAC 認証プラグイン (com.glide.tokenbased_auth) を検索します。

   名前または ID でプラグインを検索できます。プラグインが見つからない場合は、ServiceNow 担当者から要求する必要があります。
3. [インストール] を選択して、インストールプロセスを開始します。

   ℹ 注: ドメインセパレーションと代理アドミンがインスタンスで有効になっている場合、管理ユーザーはグローバルドメインに含まれている必要があります。それ以外の場合、次のエラーが表示されます： 「別の操作が実行されているため、アプリケーションのインストールは利用できません： <プラグイン名> のプラグインの有効化 (Application installation is unavailable because another operation is running: Plugin Activation for <plugin name>)」

   インストールが完了するとメッセージが表示されます。プラグインとともにインストールされるコンポーネントの詳細については、「 アプリケーションとともにインストールされているコンポーネントの検索 」を参照してください。

API キーの構成 - トークンベースの認証

REST API エンドポイントの認証をサポートするように API キーを設定します。

始める前に
必要なロール：admin

必要なプラグイン：API キーと HMAC 認証 (com.glide.tokenbased_auth)

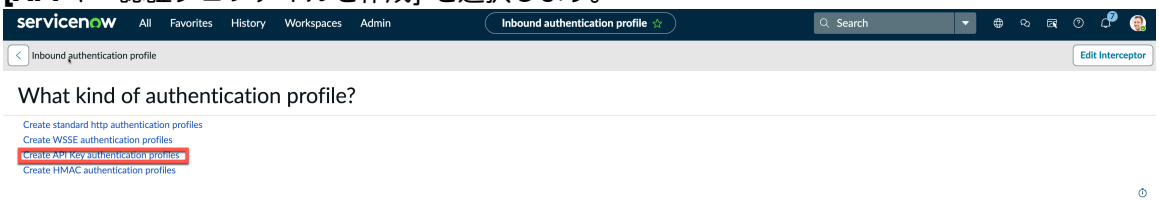
手順

1. 受信認証プロファイルを作成します。

   a. 移動先 すべて > システム **Web** サービス > **API** アクセスポリシー > 受信認証プロファイル.

   b. **[New** (新規)] を選択します。
   「認証プロファイルの種類は？」というメッセージが表示されます。

   c. **[API** キー認証プロファイルを作成] を選択します。

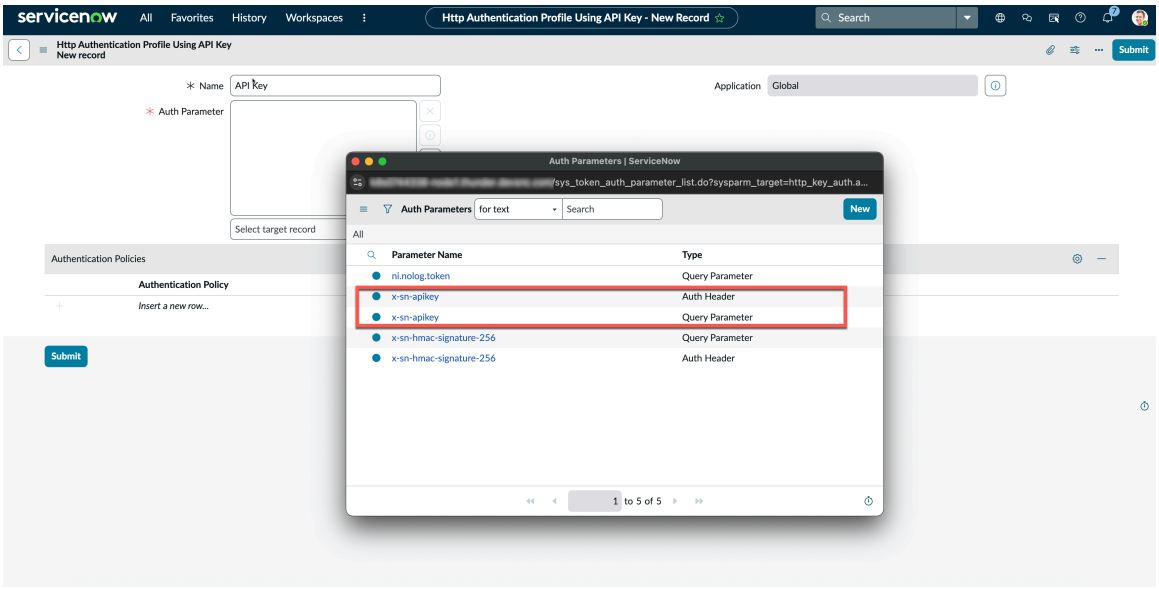


   d. フォームのフィールドに入力します。

API キー認証プロファイル

フィールド	説明
名前	認証ポリシーを識別する名前
アプリケーション	認証ポリシーの範囲
認証パラメーター	認証要求の認証パラメーターを選択します。デフォルトのオプションを選択するか、新しい認証パラメーターを作成できます。 ▪ x-sn-apikey：認証ヘッダー ▪ x-sn-apikey：クエリパラメーターのヘッダー

注: 選択したオプションは、認証ヘッダーまたはクエリパラメーターの REST 呼び出しで定義する必要があります。



注: API キーのプリフィックスを追加する場合は、選択した認証パラメーターファイルを開き、[プリフィックス] フィールドを指定します。

e. フォームを送信します。

2. REST API キーを作成します。

a. 移動先 すべて > システム **Web** サービス > **API** アクセスポリシー > **REST API** キー.

b. **[New** (新規)] を選択します。

c. フォームの各フィールドに入力します。

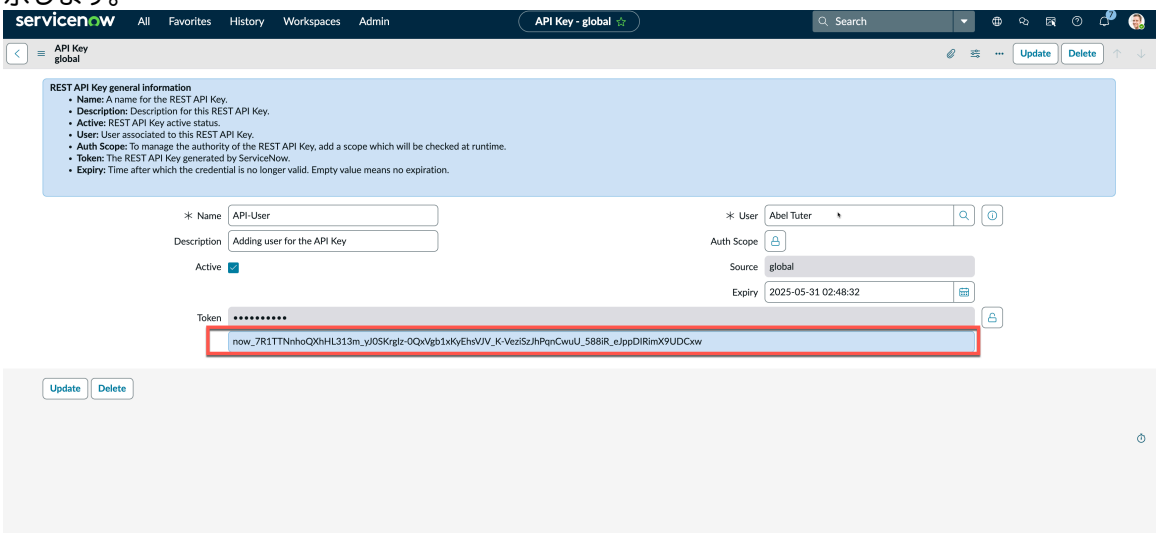
API キー

フィールド	説明
名前	REST API キーを識別するための名前
説明	REST API キーの説明。
アクティブ	REST API キーのステータス。
ユーザー	REST API キーに関連付けられたユーザー。ルックアップアイコンを使用してユーザーを選択します。
認証スコープ	REST API キーの権限を管理する認証スコープを追加するオプション。
トークン	ServiceNow AI Platform が生成した REST API キー。ヘッダーまたはクエリパラメーター内で REST API コールの一部として使用するキーをコピーします。
有効期限	認証情報が無効になるまでの時間。空の値は有効期限がないことを意味します。

フィールド	説明
	注: トークンの有効期限の詳細については、「トークンの有効期限のクリーンアップ」を参照してください。

d. フォームを送信します。

e. 作成されたレコードを開いて、ServiceNow AI Platform がユーザーのに生成したトークンを表示します。



3. REST API アクセスポリシーを作成します。

a. 移動先 すべて > システム **Web** サービス > **REST API** アクセスポリシー.

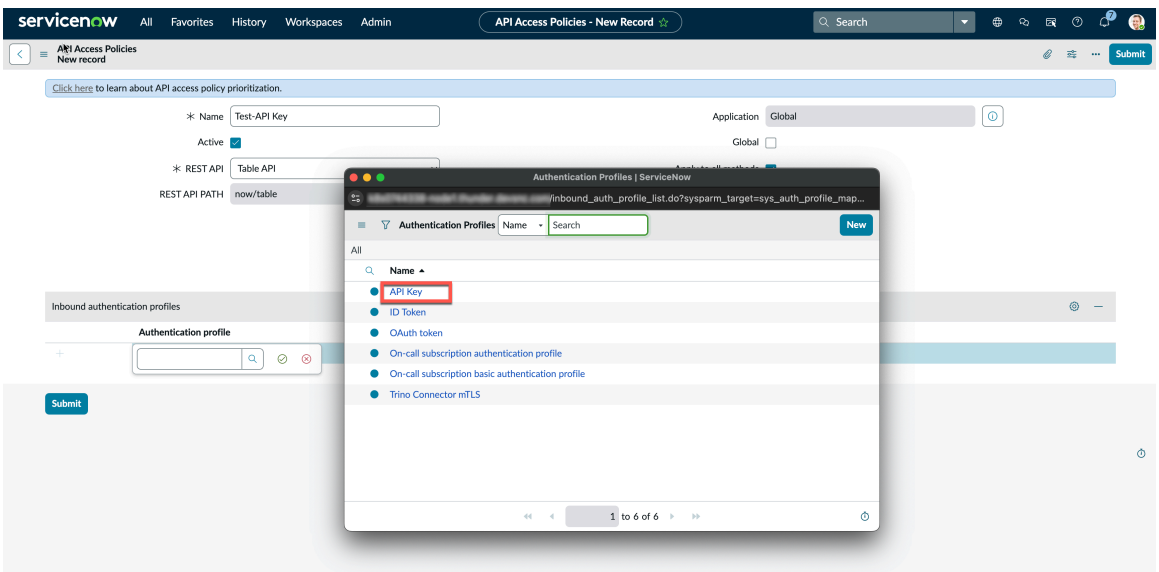
b. **[New** (新規)] を選択します。

c. フォームのフィールドに入力します。

API アクセスポリシー

フィールド	説明
名前	API アクセスポリシーの一意の名前。
有効	API アクセスポリシーをアクティブにするオプション
REST API	アクセスポリシーが適用される REST API。例：添付ファイル **API**
REST API パス	REST API の API パス。このフィールドは、選択した REST API に基づいて自動入力されます。たとえば、**now/attachment** です。
HTTP メソッド	API とのやり取りに使用されるメソッド。このフィールドは、選択した REST API に基づいて自動入力されます。

フィールド	説明
バージョン	API のバージョン。- 例：**v1**。このフィールドは、選択した REST API に基づいて自動入力されます。 **注:** REST API のすべてのバージョンに対して認証ポリシーを作成する場合は、バージョンごとに個別のポリシーを作成する必要があります。
リソース	REST API の子リソース。このフィールドは、選択した REST API に基づいて自動入力されます。例：**/now/attachment**
アプリケーション	アプリケーションのスコープ。
グローバル	このフィールドを有効にして、API のすべてのメソッド、バージョン、およびリソースに認証ポリシーを適用可能にします。 **注:** グローバル REST API ポリシーではトークンベースの認証は許可されていません。
すべてのメソッドに適用	このフィールドを有効にして、API のすべてのメソッド、バージョン、およびリソースに API の認証ポリシーを適用可能にします。
すべてのリソースに適用	このフィールドを有効にして、すべてのバージョンに API の認証ポリシーを適用可能にします。
すべてのバージョンに適用	このフィールドを有効にして、すべてのリソースに API の認証ポリシーを適用可能にします。



自動翻訳

d. 作成された API 認証プロファイルを追加します。

e. フォームを送信します。

認証の構成に基づいて、ヘッダーパラメーターまたはクエリパラメーター内で API キーの作成時に ServiceNow AI Platform が生成した x-sn-apikey (トークン) を使用して REST API 呼び出しを送信できます。

⚠ 警告: サーバーに機密情報を送信する場合は、**POST** 要求を使用します。

HMAC の構成 - トークンベースの認証

REST API エンドポイントの認証をサポートするように HMAC を設定します。

始める前に
必要なロール：admin

必要なプラグイン：API キーと HMAC 認証 (com.glide.tokenbased_auth)

手順

1. HMAC 構成を作成します。

a. 移動先 すべて > システム **Web** サービス > **API** アクセスポリシー > **HMAC** 構成.

b. [New (新規)] を選択します。

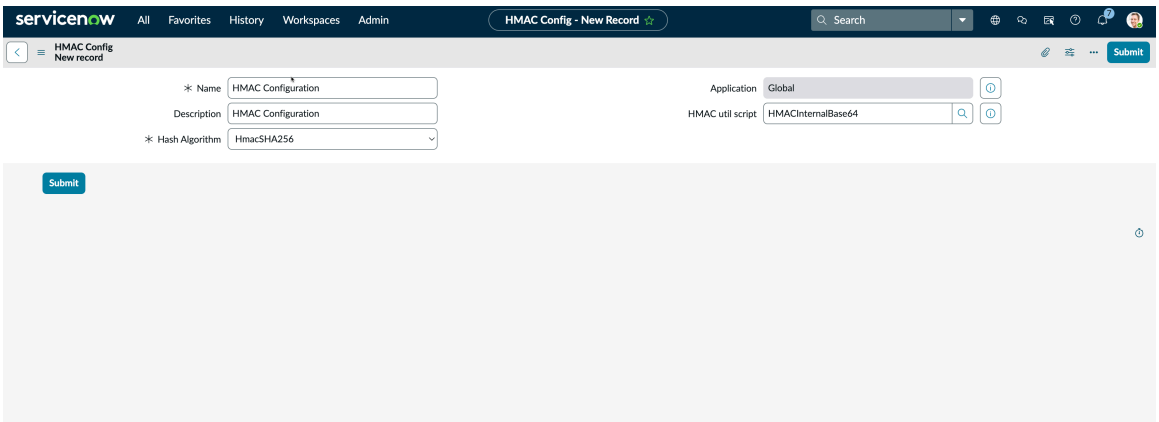
ℹ 注: プラグインのインストール時に作成されるデフォルトの **HMAC SHA256 Base64** エンコーディングを使用することもできます。

c. フォームの各フィールドに入力します。

HMAC 構成

フィールド	説明
名前	HMAC 構成の名前。
アプリケーション	構成のスコープ。
説明	構成に関する詳細な説明。
ハッシュアルゴリズム	ハッシュアルゴリズムを選択します。利用可能なオプション： ▪ HmacSHA256 ▪ HmacSHA384 ▪ HmacSHA512
HMAC ユーティリティスクリプト	HMAC のユーティリティスクリプト。

フィールド	説明
	注: キー ID なしで要求本文、タイムスタンプ、およびシークレットを使用して HMAC 認証を検証する場合は、次の手順を実行します。 ▪ カスタマイズされたスクリプトインクルードを HMAC ユーティリティスクリプトとして作成します。 ▪ そのスクリプトインクルードユーティリティを使用して、渡されたタイムスタンプと要求本文を解釈します。 ServiceNow® インスタンスで作成された共有シークレットごとに、キー ID があります。HMAC 認証プロファイルのデフォルトキー ID に、そのキー ID を構成する必要があります。



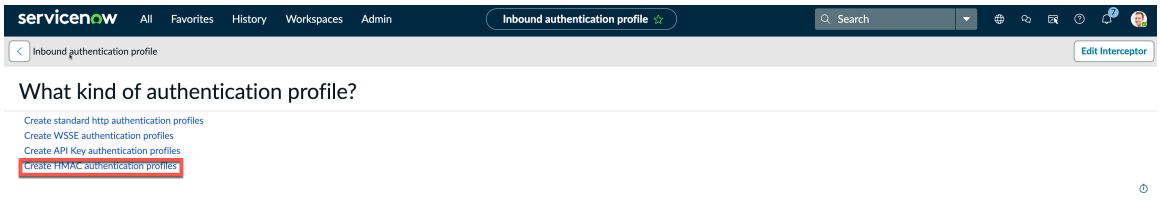
d. レコードを送信します。

2. 受信認証プロファイルを作成します。

a. 移動先 すべて > システム **Web** サービス > **API** アクセスポリシー > 受信認証プロファイル.

b. **[New** (新規)] を選択します。
「認証プロファイルの種類は？」というメッセージが表示されます。

c. **[HMAC** 認証プロファイルを作成] を選択します。

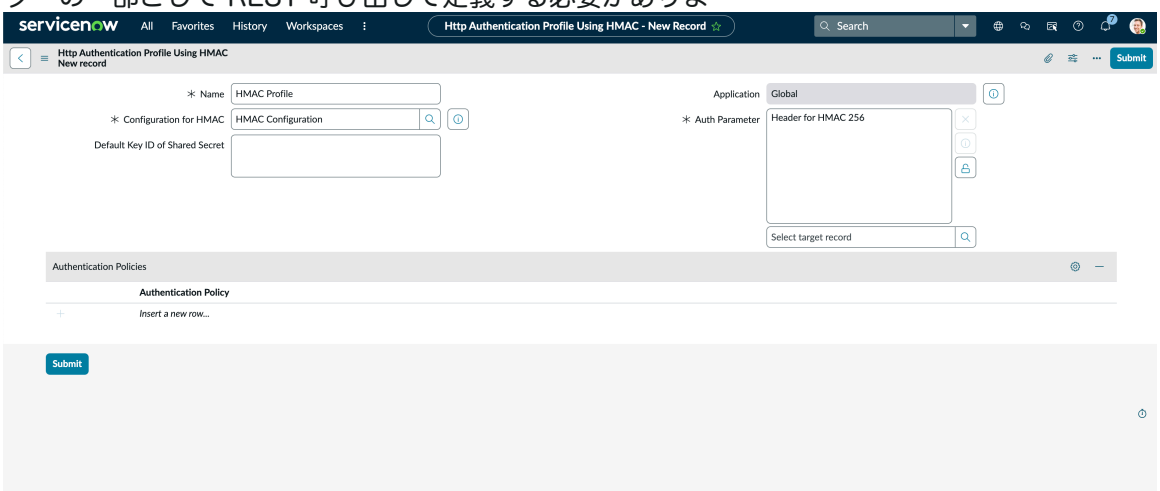


d. フォームのフィールドに入力します。

HMAC 認証プロファイル

フィールド	説明
名前	認証ポリシーを識別する名前
アプリケーション	認証ポリシーの範囲
HMAC の構成	作成された HMAC 構成を選択します。
認証パラメーター	認証要求の認証パラメーターを選択します。デフォルトのオプションを選択するか、新しい認証パラメーターを作成できます。 ▪ x-sn-hmac-signature-256：認証ヘッダー ▪ x-sn-hmac-signature-256：クエリパラメーター
共有シークレットのデフォルトキー ID	HMAC を使用するためにこのフィールドで更新できるトークン情報。

注: 選択したオプションは、認証ヘッダーまたはクエリパラメーターの一部として REST 呼び出しで定義する必要がありま



す。

e. フォームを送信します。

3. HMAC シークレットを作成します。

   a. 移動先 すべて > システム **Web** サービス > **API** アクセスポリシー > **REST API HMAC** シークレット.

   b. **[New** (新規)] を選択します。

   c. フォームの各フィールドに入力します。

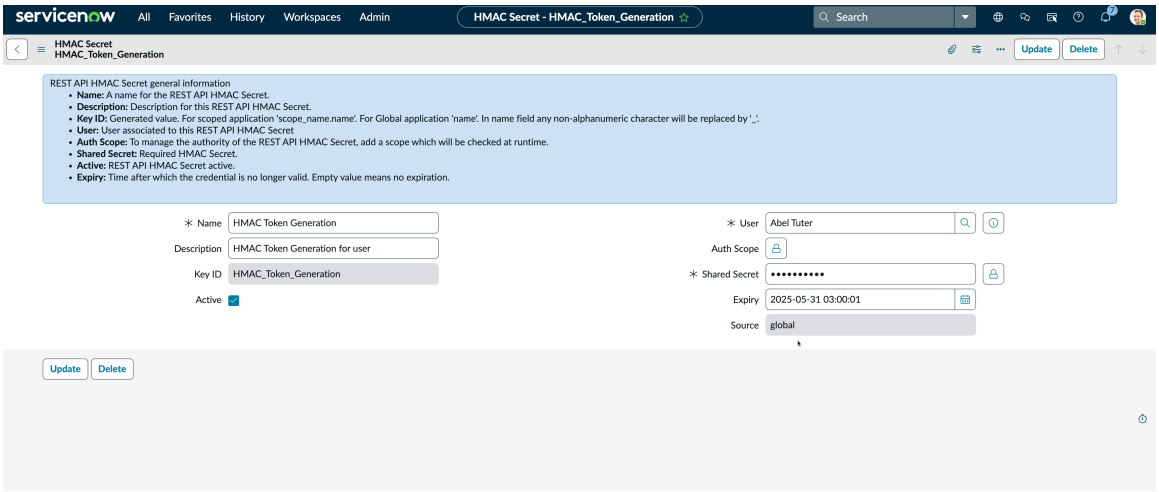
   REST API HMAC シークレット

フィールド	説明
名前	REST API HMAC シークレットを識別するための名前
説明	REST API HMAC シークレットの説明。
アクティブ	REST API HMAC シークレットのステータス。
ユーザー	REST API HMAC シークレットに関連付けられたユーザー。ルックアップアイコンを使用してユーザーを選択します。
キー ID	REST 呼び出しの一部として送信する必要があるキー ID。キー ID はフォームの送信後に生成されます。
共有シークレット	ユーザーの共有シークレット。パスワードなどです。
ソース	レコードのソース。
有効期限	認証情報が無効になるまでの時間。空の値は有効期限がないことを意味します。 注: トークンの有効期限の詳細については、「トークンの有効期限のクリーンアップ」を参照してください。

   d. フォームを送信します。

   e. 作成されたレコードを開きます。

   ServiceNow AI Platform がユーザーに生成したキー ID を検索します。



注: API 呼び出しの認証パラメーターまたはクエリパラメーターを指定しない場合は、HMAC 用に作成された認証プロファイルのキー ID 時に生成されたキー ID を追加できます。

4. REST API アクセスポリシーを作成します。

a. 移動先 すべて > システム **Web** サービス > **REST API** アクセスポリシー.

b. **[New** (新規)] を選択します。

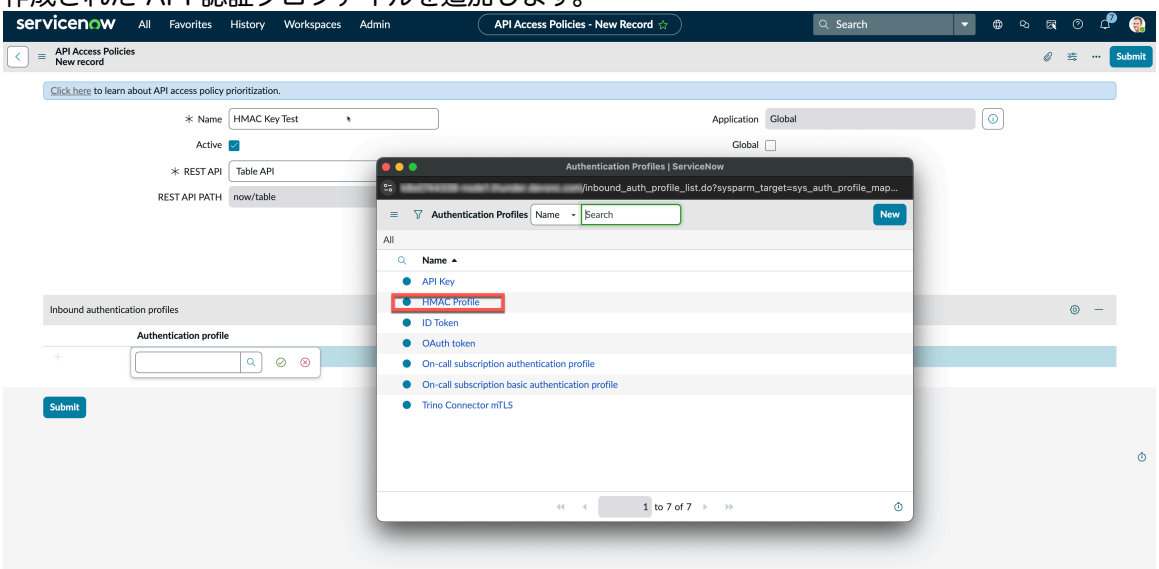
c. フォームのフィールドに入力します。

API アクセスポリシー

フィールド	説明
名前	API アクセスポリシーの一意の名前。
有効	API アクセスポリシーをアクティブにするオプション
REST API	アクセスポリシーが適用される REST API。例：添付ファイル **API**
REST API パス	REST API の API パス。このフィールドは、選択した REST API に基づいて自動入力されます。たとえば、**now/attachment** です。
HTTP メソッド	API とのやり取りに使用されるメソッド。このフィールドは、選択した REST API に基づいて自動入力されます。
バージョン	API のバージョン。- 例：**v1**。このフィールドは、選択した REST API に基づいて自動入力されます。 **注:** REST API のすべてのバージョンに対して認証ポリシーを作成する場合は、バージョンごとに個別のポリシーを作成する必要があります。

フィールド	説明
リソース	REST API の子リソース。このフィールドは、選択した REST API に基づいて自動入力されます。例：**/now/attachment**
アプリケーション	アプリケーションのスコープ。
グローバル	このフィールドを有効にして、API のすべてのメソッド、バージョン、およびリソースに認証ポリシーを適用可能にします。 **注:** グローバル REST API ポリシーではトークンベースの認証は許可されていません。
すべてのメソッドに適用	このフィールドを有効にして、API のすべてのメソッド、バージョン、およびリソースに API の認証ポリシーを適用可能にします。
すべてのリソースに適用	このフィールドを有効にして、すべてのバージョンに API の認証ポリシーを適用可能にします。
すべてのバージョンに適用	このフィールドを有効にして、すべてのリソースに API の認証ポリシーを適用可能にします。

d. 作成された API 認証プロファイルを追加します。



e. フォームを送信します。
次の方法で REST API 呼び出しを送信できます。

- 認証の構成に基づいて、ヘッダーパラメーターまたはクエリパラメーター内で API キーの作成時に ServiceNow が生成した x-sn-hmac-signature-256 を使用します。
- 共有シークレットが要求の一部として指定された事前要求スクリプトを使用します。

警告: サーバーに機密情報を送信する場合は、**POST** 要求を使用します。

トークンの有効期限のクリーンアップ

さまざまなシステムプロパティを使用してトークンの有効期限をクリーンアップする方法の詳細について説明します。

API キーまたは HMAC シークレットの有効期限が切れると、詳細は 7 日間保持されます。スケジュール済みジョブ Clean expired Token Auth Credentials は、その後、期限切れの API キーまたは HMAC シークレットを削除します。以下のプロパティを使用して、この動作を構成し、削除を制御できます。

- **com.snc.platform.security.token.auth.cleanup**：期限切れの API キーと HMAC トークンを削除する場合は、このプロパティを使用します。デフォルトは true です。
- **com.snc.platfrom.security.token.auth.days.expired.api_key.is.keeppt**：要件に基づいて値を設定し、期限切れの API トークンをシステムに保持する日数を決定します。デフォルトは 7 です。
- **com.snc.platfrom.security.token.auth.days.expired.hmac_key.is.keeppt**：要件に基づいて値を設定し、期限切れの HMAC トークンをシステムに保持する日数を決定します。デフォルトは 7 です。

トークンの有効期限プロパティに移動するには、ナビゲーションバーに「sys_properties.list」と入力し、次のプロパティを検索します。要件に基づいて有効期限を変更します。

API アクセスポリシー

API アクセスポリシーは、API へのアクセス権限と期間を定義します。

API アクセスポリシーを使用すると、認証タイプとアクセスポリシーの指定されたフィルター基準に基づいて、受信 ServiceNow API へのアクセスを制限できます。

非インタラクティブセッションの 3 つのレベルでの認証ポリシーサポートは以下のとおりです。

- **REST API**：アドミンは、グローバル API 認証ポリシーまたは API 固有の認証ポリシーを定義できます。
- **SOAP API**：アドミンは、グローバル API 認証ポリシーまたは API 固有の認証ポリシーを定義できます。
- システム/エクスポートプロセッサー：アドミンは、プロセッサー API 認証ポリシーまたはプロセッサー固有の認証ポリシーに認証プロファイルを適用するために使用できる 5 つのベースシステム認証プロファイルを定義できます。

注: グローバル認証ポリシーを追加すると、そのポリシーはすべての REST API、SOAP API、またはシステム/エクスポートプロセッサーに適用されます。

API ポリシーと認証スコープ用に次のプラグインが自動的にインストールされます。

- com.glide.rest.policy
- com.glide.soap.policy
- com.glide.processor.policy
- com.glide.rest.auth.scope

デフォルトのグローバルブロックポリシーを設定するか、セキュリティ要件に従ってカスタム API アクセスポリシーを作成できます。さらに、認証ポリシーのポリシー入力として使用されるフィルター条件またはクエリが含まれているフィルター基準も適用します。

ServiceNow® では次の API アクセスポリシーがサポートされています。

- REST API アクセスポリシー
- SOAP API アクセスポリシー

エクスポートプロセッサーに関連するポリシーの詳細については、「システムプロセッサーまたはエクスポートプロセッサーのアクセスポリシー」を参照してください。

REST API アクセスポリシー

REST API アクセスポリシーを使用すると、認証タイプとアクセスポリシーの指定されたフィルター基準に基づいて、受信 REST API へのアクセスを制限できます。

REST API は RESTful API ともいい、REST アーキテクチャスタイルのガイドラインに準拠したアプリケーションプログラミングインターフェイス (API) の一種です。REST API は柔軟性が高く、Web 全体で広く使用されています。

フィルター基準には、認証ポリシーのポリシー入力として使用されるフィルター条件またはクエリーが含まれています。

デフォルトのグローバルブロックポリシーを設定するか、セキュリティ要件に従ってカスタム API アクセスポリシーを作成できます。たとえば、指定された IP アドレス範囲から OAuth 2.0 認証タイプのみを許可するカスタム API アクセスポリシーを作成できます。他の認証タイプの認証要求、および指定された IP アドレス以外の IP アドレスからのアクセス要求は拒否されます。

REST API アクセスポリシーのアクティブ化

admin ロールがあれば、REST API Access Policy プラグイン (com.glide.rest.policy) をアクティブ化できます。このアプリケーションにはデモデータが含まれています。まだインストールされていない場合は、関連する ServiceNow Store アプリケーションとプラグインをインストールします。

始める前に
必要なロール：admin。

このタスクについて
REST API アクセスポリシーとともにインストールされるアイテム：

- プラグイン：
  com.gilde.auth.profile、com.snc.adaptive_authentication、com.snc.platform.security.oauth
- テーブル：
  sys_api_access_policy、sys_auth_profile_mapping、auth_policy_mapping、inbound_auth_profile、std_http

詳細については、「適応認証」を参照してください。

手順

1. 移動先 すべて > システムアプリケーション > 利用可能なすべてのアプリケーション > すべて.
2. フィルター基準と検索バーを使用して、REST API Access Policy プラグイン (com.glide.rest.policy) を検索します。

   名前または ID でプラグインを検索できます。プラグインが見つからない場合は、ServiceNow 担当者から要求する必要があります。
3. [インストール] を選択して、インストールプロセスを開始します。

> **注:** ドメインセパレーションと代理アドミンがインスタンスで有効になっている場合、管理ユーザーはグローバルドメインに含まれている必要があります。それ以外の場合、次のエラーが表示されます： 「別の操作が実行されているため、アプリケーションのインストールは利用できません： <プラグイン名> のプラグインの有効化 (Application installation is unavailable because another operation is running: Plugin Activation for <plugin name>)」

インストールが完了するとメッセージが表示されます。プラグインとともにインストールされるコンポーネントの詳細については、「 アプリケーションとともにインストールされているコンポーネントの検索 」を参照してください。

認証プロファイルの作成

認証プロファイルを作成し、1 つ以上の認証ポリシーをプロファイルに追加します。デフォルトで利用可能な **ID** トークンおよび **OAuth** トークンの認証プロファイルを設定することもできます。

始める前に

必要なロール：admin

> **注:** 相互認証とカスタマイズされた認証を使用して、認証ポリシー、IP 範囲、ロールベース、ユーザーベースなどを適用できます。

手順

1. 移動先 すべて > システム **Web** サービス > **API** アクセスポリシー > 受信認証プロファイル.
2. **[New** (新規)] を選択します。
   メッセージが表示されます。認証プロファイルの種類は？ (What kind of authentication profile?)
3. [標準の **http** 認証プロファイルを作成] を選択します。
4. フォームのフィールドに入力します。

[標準の認証プロファイル] フォーム

フィールド	説明
名前	認証ポリシーを識別する名前
説明	認証ポリシーの説明
有効	認証ポリシーをアクティブにするオプション
アプリケーション	認証ポリシーの範囲
タイプ	利用可能な認証のタイプ。[基本認証]、**[ID** トークン]、[証明書ベースの認証]、**[OAuth]** を選択できます。
OAuth エンティティ	OAuth エンティティプロファイル。このフィールドは、[タイプ] で **[ID** トークン] または **[OAuth]** を選択した場合のみ表示されます。

5. [新規行を挿入] をダブルクリックします。
6. リストから認証ポリシーを選択し、保存アイコン ( ) を選択します。

> **注:** [アクセス許可ポリシー (Allow Access Policy)] または [アクセス拒否ポリシー (Deny Access Policy)] は選択しないでください。これらのポリシーは、ユーザーログインのみを対象としています。

1 つ以上の認証ポリシーを認証プロファイルに追加できます。

注:

認証ヘッダー [WWW-Authenticate]。

REST API アクセスポリシーがアクティブな場合、認証ヘッダーで直近にマッピングされた認証プロファイルが返されます。サーバーにすべての認証スキームを返させる場合は、glide.security.response.authenticate.header.auth_profile.first_scheme_only プロパティを使用して **false** に設定します。応答は複数のヘッダーを使用して返されます。例：

```
< WWW-Authenticate: BEARER realm="Service-now"
< WWW-Authenticate: BASIC realm="Service-now"
```

REST API アクセスポリシーの作成

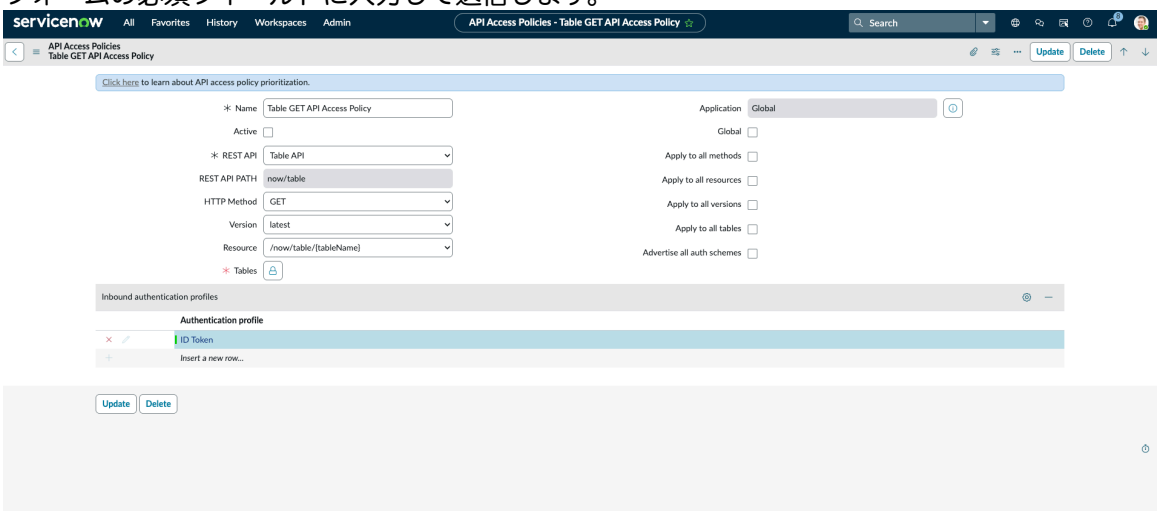
API アクセスポリシーを作成し、認証プロファイルをマップして REST API の認証タイプを制限します。たとえば、REST API の ID トークン認証のみを許可する API アクセスポリシーを作成できます。

始める前に
認証プロファイルが作成されていることを確認します。詳細については、「認証プロファイルの作成」を参照してください。

必要なロール：admin

手順

1. 移動先 すべて > システム **Web** サービス > **REST API** アクセスポリシー.
2. **[New** (新規)] を選択します。
3. フォームの必須フィールドに入力して送信します。



注: 追加のフィールドに入力するには、送信されたフォームを再度開く必要があります。

API アクセスポリシー

フィールド	説明
名前	API アクセスポリシーの一意の名前。
有効	API アクセスポリシーをアクティブにするオプション

フィールド	説明
REST API	アクセスポリシーが適用される REST API。例：添付ファイル **API**
REST API パス	REST API の API パス。このフィールドは、選択した REST API に基づいて自動入力されます。たとえば、**now/attachment** です。
HTTP メソッド	API とのやり取りに使用されるメソッド。このフィールドは、選択した REST API に基づいて自動入力されます。
バージョン	API のバージョン。- 例：**v1**。このフィールドは、選択した REST API に基づいて自動入力されます。 **注:** REST API のすべてのバージョンに対して認証ポリシーを作成する場合は、バージョンごとに個別のポリシーを作成する必要があります。
リソース	REST API の子リソース。このフィールドは、選択した REST API に基づいて自動入力されます。例：**/now/attachment**
テーブル	API アクセスポリシーが適用されるテーブル。このオプションは、 テーブル API のポリシーにのみ適用されます。
アプリケーション	アプリケーションのスコープ。
グローバル	API のすべてのメソッド、バージョン、およびリソースにポリシーを適用するオプション。
すべてのメソッドに適用	API のすべてのメソッド、バージョン、およびリソースにポリシーを適用するオプション。
すべてのリソースに適用	ポリシーをすべてのリソースまたは API リソースに適用するオプション。
すべてのバージョンに適用	すべてのバージョンまたは API バージョンにポリシーを適用するオプション。
全テーブルに適用	すべてのテーブルにポリシーを適用するオプション。このオプションは、 テーブル API のポリシーにのみ適用されます。
A:すべての認証スキームを公示	WWW-Authenticate ヘッダーに、構成されているすべての認証スキームを含めるかどうかを決定します。false (デフォルト) に設定すると、ヘッダーには、ポリシーに最後に設定された認証プロファイルのみが含まれます。true に設定すると、構成済みのすべての認証スキームがヘッダーにリストされます。

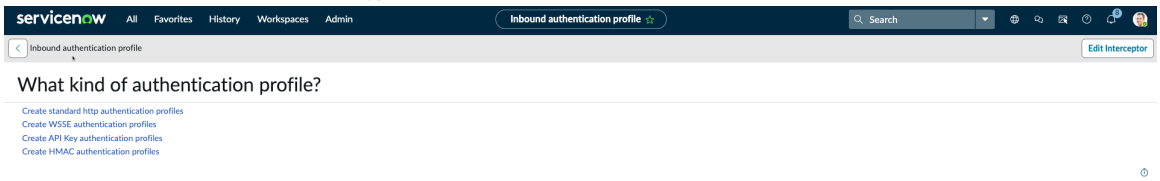
注: API アクセスポリシーの優先順位付けの詳細については、「API アクセスポリシーの優先順位付け」を参照してください。

4. [**Insert a new row** (新規行を挿入)] をダブルクリックします。

5. リストから受信認証プロファイルを選択し、保存アイコン [ ]を選択します。
   たとえば、基本認証、**ID** トークン、証明書ベースの認証、**OAuth**、または **WSSE** 認証 を追加できます。

   a. 1 つ以上の受信認証プロファイルを追加するには、[新規] を選択して新しいプロファイルを作成します。

   b. [認証プロファイルの種類は？ **(What kind of authentication profile?)**] を選択します。
      - 標準の **http** 認証プロファイルを作成
      - **WSSE** 認証プロファイルを作成
      - **API** キー認証プロファイルを作成
      - **HMAC** 認証プロファイルを作成



   c. 認証プロファイルを作成したら、レコードを保存します。

6. [送信] を選択して、REST API アクセスポリシーを送信します。

API アクセスポリシーの優先順位付け

ServiceNow® インスタンスに複数の API アクセスポリシーが設定されている場合のポリシーの優先順位付けのロジックについて説明します。

API アクセスポリシーは、ServiceNow® インスタンスに設定された REST API ポリシーのタイプに基づいて優先順位が付けられます。

このアプローチでは、メソッド、リソース、バージョンなどの API の各部分に異なる重み付けを定義します。

API ポリシーでは、最初に非グローバル、次にグローバルが優先されます。つまり、非グローバルアクセスポリシーは常にグローバル API アクセスポリシーを上書きします。

優先順位付けのロジックは次のとおりです。

優先順位付け

フィールド	優先度	優先順位付けのロジック
メソッド、リソース、およびバージョン	1	3 つのフィールドがポリシーと一致する場合、そのポリシーが 1 番目の優先順位になります。
メソッド + リソース	2	2 つのフィールドがポリシーと一致する場合、そのポリシーが 1 番目の優先順位になります。

優先順位付け (続く)

フィールド	優先度	優先順位付けのロジック
リソース + バージョン	3	2 つのフィールドおよび [すべてのメソッドに適用] フィールドがポリシーと一致する場合、そのポリシーが 3 番目の優先順位になります。
リソース	4	フィールドおよび [すべてのメソッドに適用] フィールドがポリシーと一致する場合、そのポリシーが 4 番目の優先順位になります。
メソッド + バージョン	5	2 つのフィールドおよび [すべてのリソースに適用] フィールドがポリシーと一致する場合、そのポリシーが 5 番目の優先順位になります。
メソッド	6	フィールドおよび [すべてのリソースに適用] フィールドがポリシーと一致する場合、そのポリシーが 5 番目の優先順位になります。
バージョン	7	フィールドおよび [すべてのメソッドに適用] フィールドおよび [すべてのバージョンに適用] フィールドがポリシーと一致する場合、そのポリシーが 7 番目の優先順位になります。
グローバルおよびすべてのメソッドに適用	8	[グローバル] フィールドが [true] で、[すべてのメソッドに適用] が [false] の場合、そのポリシーは 8 番目の優先順位になります。
グローバルおよびすべてのメソッドに適用	9	[グローバル] フィールドが [true] で、[すべてのメソッドに適用] が [true] の場合、そのポリシーは 9 番目の優先順位になります。

REST API 認証スコープ

REST API 認証スコープを使用して、特定の REST API へのアクセスを提供します

以前は、すべてのアクセストークンまたは OIDC トークンが、ユーザーの REST API へのフルアクセス権を持つ `ユーザーアカウント` スコープにリンクされていました。Zurich リリースから、特定の REST API にのみアクセスできるようにするために、**REST API** 認証スコープが導入されました。

REST API 認証スコープレコードを作成した後、この REST API にアクセスするには、この REST API にアクセスできる必要がある OAuth エンティティに同じ認証スコープを関連付ける必要があります。新しい OAuth エンティティの場合、デフォルトの認証スコープは空です。

注: REST API 認証スコープレコードがない限り、有効な OAuth エンティティであれば REST API にアクセスできます。

OAuth エンティティ内の認証スコープを手動でリンクする必要があります。`ユーザーアカウント`は特別なスコープです。OAuth エンティティに関連付けられている場合、別の認証スコープで REST API 認証スコープレコードを作成していても、任意の API にアクセスできます。

新しいインバウンド統合エクスペリエンスで API 認証スコープを使用する方法の詳細については、「インバウンド統合」を参照してください。

注:

- REST API 認証スコープが有効になって REST API の認証スコープに追加された後は、アドミニストレーターがこの認証スコープを対応する OAuth エンティティに追加しない限り、既存のすべての OAuth トークンがこの API にアクセスできなくなります
- アドミニストレーターは、認証スコープを REST API にリンクした後に、oauth_entityに適切な認証スコープがあることを確認する責任があります。
- ServiceNow によって発行された OAuth アクセストークンは認証スコープをサポートします。
- ServiceNow によって発行されていない OIDC トークンは ServiceNow によって検証されます。
- ID トークンが必要な場合、OIDC トークンには IDP からのスコープがあります。ここでは、認証スコープはサードパーティ (IdP) 用ではなく ServiceNow 用です。

REST API スコープの設定

REST API スコープを設定するには、次のタスクを実行します。

- 認証スコープを作成する
- 認証スコープを REST API にリンクする
- 認証スコープを OAuth エンティティにリンクする
- OAuth フローを実行して OAuth アクセストークンを取得する
- OAuth アクセストークンを使用して API 呼び出しを行う

REST API 認証スコープのアクティブ化

REST API Auth Scope プラグイン (com.glide.rest.auth.scope) をアクティブ化して、OAuth エンティティを認証スコープにリンクすることができます。

始める前に
次のプラグインをインストールします。

- OAuth 2.0
- REST API Provider
- Authentication scope
- REST API Scope

注: *REST API Scope* プラグインは Tokyo リリースの一部として追加されます。

必要なロール：admin

手順

1. 移動先 すべて > システムアプリケーション > 利用可能なすべてのアプリケーション > すべて.
2. フィルター基準と検索バーを使用して、REST API auth Scope (*com.glide.rest.auth.scope*) プラグインを検索します。

   名前または ID でプラグインを検索できます。プラグインが見つからない場合は、ServiceNow 担当者から要求する必要があります。
3. [インストール] を選択して、インストールプロセスを開始します。

   注: ドメインセパレーションと代理アドミンがインスタンスで有効になっている場合、管理ユーザーはグローバルドメインに含まれている必要があります。それ以外の場合、次のエラーが表示されます： 「別の操作が実行されているため、アプリケーションのインストールは利用できません： <プラグイン名> のプラグインの有効化 (Application installation is unavailable because another operation is running: Plugin Activation for <plugin name>)」

インストールが完了するとメッセージが表示されます。プラグインとともにインストールされるコンポーネントの詳細については、「 アプリケーションとともにインストールされているコンポーネントの検索 」を参照してください。

REST API Auth Scope のプロパティとテーブル

REST API Auth Scope プラグイン (com.glide.rest.auth.scope) には、次のシステムプロパティ、テーブル、およびスクリプトが含まれています。

REST API Auth Scope のプロパティ

REST API Auth Scope では、次のシステムプロパティが追加されています。

プロパティ

名前	説明
com.glide.rest.api.auth.scope.check.enable	このプロパティは、プラットフォームレベルの認証スコープのチェックをオフにするために使用します。 false に設定すると、実行時に REST API にリンクされているかどうかにかかわらず、認証スコープのチェックがスキップされます。 デフォルトでは、このプロパティは [true] に設定されます。このプロパティは、以前のリリースの動作に戻す場合に使用します。
com.glide.oauth.token.scope.useraccount	このプロパティは、エンドユーザーが *useraccount* 認証スコープを削除して手動で追加するときにのみ使用します。 その場合は、*useraccount* のシステム ID が変更されます。このプロパティを新しい sys_id に更新する必要があります。 実行時は、認証スコープ名の代わりに認証スコープのシステム ID が使用されます。

自動翻訳

REST API 認証スコープのテーブル

REST API 認証スコープには次のテーブルがあります。

テーブル

名前	説明
認証スコープ (sys_auth_scope)	このテーブルでは、REST API および OAuth エンティティとリンクできる認証スコープを定義します。 認証スコープ名は一意である必要があり、グローバルで使用します。
REST API 認証スコープ (sys_api_access_scope)	このテーブルでは、REST API を認証スコープにリンクします。

REST API 認証スコープの設定

OAuth エンティティを認証スコープにリンクすることで、認証スコープにリンクされている REST API にアクセスするためのトークンを管理します。

始める前に
次のプラグインをインストールします。

- OAuth 2.0
- REST API Provider
- Authentication scope
- REST API Auth Scope

注: *REST API Auth Scope* プラグインは Tokyo リリースの一部として追加されます。

必要なロール：admin

手順

1. 移動先 すべて > **API** 認証スコープ > **REST API** 認証スコープ.
   [REST API 認証スコープ] ページが表示されます。
2. 新しい REST API 認証スコープを設定するには、[新規] をクリックします。
3. フォームのフィールドに入力します。

REST API 認証スコープ

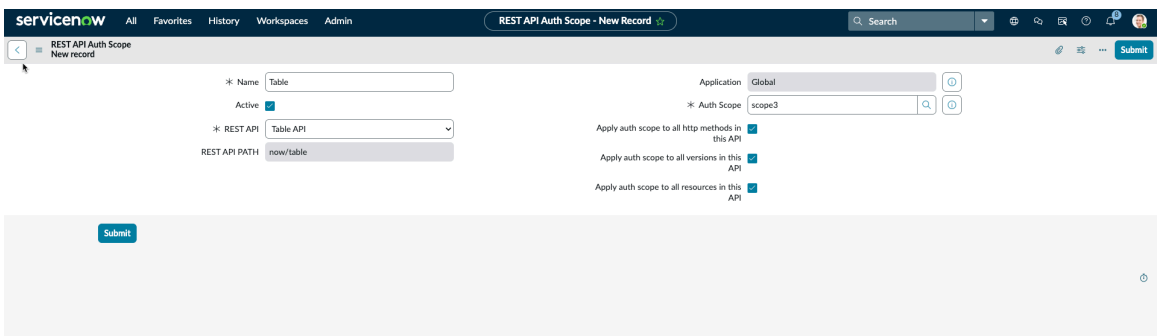
名前	REST API 認証スコープを識別する一意の名前。
有効	チェックボックスをオンにすると、設定がアクティブになります。
アプリケーション	読み込み専用のアプリケーションスコープ
REST API	認証スコープが適用される REST API。たとえば、テーブル API などです。
認証スコープ	検索アイコンから認証スコープを選択します。

REST API パス	REST API の API パス。このフィールドは、選択した REST API に基づいて自動入力されます。例：now/table
HTTP メソッド	API とのやり取りに使用されるメソッド。ドロップダウンリストからメソッドを選択します。 フォームの [この **API** のすべての **HTTP** メソッドに認証スコープを適用] フィールドを手動で無効にすると、メソッドを選択できます。
REST API バージョン	API のバージョン。- 例：v1。このフィールドは、選択した REST API に基づいて自動入力されます。 フォームの [この **API** のすべてのバージョンに認証スコープを適用] フィールドを手動で無効にすると、バージョンを選択できます。
リソース	REST API の子リソース。このフィールドは、選択した REST API に基づいて自動入力されます。例：/now/table フォームの [この **API** のすべてのリソースに認証スコープを適用] フィールドを手動で無効にすると、リソースを選択できます。
この API のすべての HTTP メソッドに認証スコープを適用	これを有効にすると、API のすべての HTTP メソッドに認証スコープが適用されます。
この API のすべてのバージョンに認証スコープを適用	これを有効にすると、API のすべてのバージョンに認証スコープが適用されます。
この API のすべてのリソースに認証スコープを適用	これを有効にすると、API のすべてのリソースに認証スコープが適用されます。

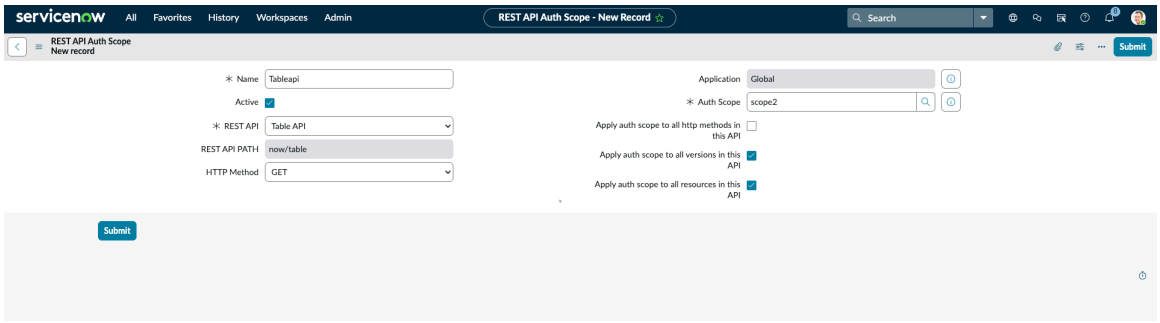
4. [送信] をクリックします。
   選択された REST API と認証スコープに基づいて、API はスコープに固有の情報を取得します。

Example: テーブル **API** に対して **3** つの **REST API** 認証スコープを作成することを検討してください。

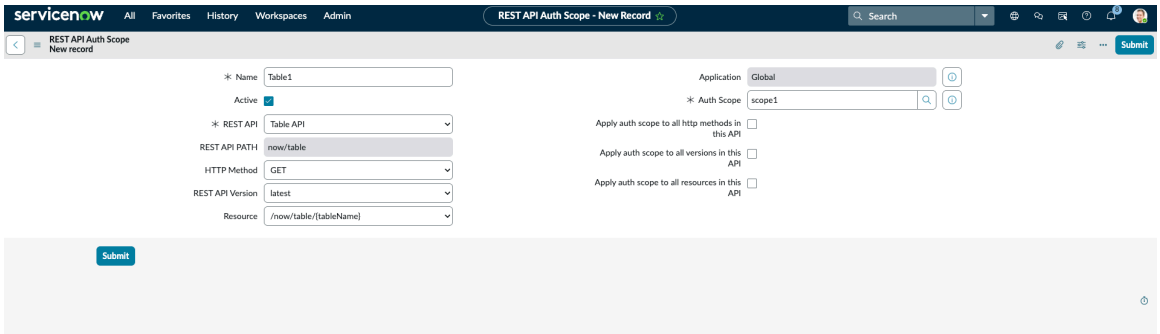
1 つ目の認証スコープは、すべての HTTP メソッド、バージョン、およびリソースが有効な状態で [テーブル **API**] にマッピングされます。



2 つ目の認証スコープは、すべてのバージョンとリソースが有効な状態で [テーブル **API]** にマッピングされます。ただし、HTTP メソッド (この例では **GET** メソッド) は選択してください。



3 つ目の認証スコープは、HTTP メソッド、バージョン、およびリソースが有効でない状態で [テーブル **API]** にマッピングされます。ただし、HTTP メソッド、バージョン、およびリソースは手動で選択してください。この例では、HTTP メソッドは **GET**、REST API のバージョンは最新、リソースは /now/table/{tableName} です。



これらの認証スコープがすべて作成されると、3 つのスコープすべてで **GET** メソッドを使用できますが、**POST**、**PUT**、**DELETE**、または **PATCH** メソッドでは **scope3** しか使用できません。

REST API スコープのトラブルシューティング

トラブルシューティングのアクションは、REST API スコープの設定や実行時の一般的な問題解決に役立ちます。

トラブルシューティング

問題	アクション
REST API が認証スコープにリンクされているのに、ベアラートークン認証を使用しても実行時に認証スコープのチェックが行われない。	• sys_api_access_policy レコードがアクティブであることを確認します。ランタイムでは非アクティブなレコードが無視されます。 • プロパティ *com.glide.rest.api.auth.scope.check.enable* が false に設定されているかどうかを確認します。 • OAuth トークンに useraccount 認証スコープがあるかどうかを確認します。

トラブルシューティング (続く)

問題	アクション
REST API が *auth_scope1*にリンクされているのに、*auth_scope2* があるアクセストークンでも REST API にアクセスできてしまう。	• このレコードがアクティブかどうかを確認します。 • この REST を確認して、API が同じでありながら異なるメソッド、バージョン、またはリソースを適用する他のレコードがあるかどうかを確認します。
REST API が認証スコープにリンクされているのに、*basicAuth* と *mutualAuth* に対して実行時に認証スコープのチェックが行われない。	REST API 認証スコープは OAuth アクセストークンまたは OIDC トークンにのみ適用されるため、これは予想どおりの動作です。BasicAuth、セッション Cookie、および証明書ベースの認証は適用されません。
OAuth アクセストークンを使用すると、REST API の呼び出しで 403 が返される。	「必要な API アクセススコープがありません (Missing required api access scope)」というエラーメッセージがあるかどうか確認してください。もしあれば、この REST API の認証スコープのチェックは失敗します。
事前定義した useraccount が削除されて復元できない。	useraccount を他のインスタンスから xml としてエクスポートしてインポートするか、useraccount を作成して、システムプロパティ *glide.oauth.token.scope.useraccount* を新しく作成された sys_id レコードに変更します。

よく寄せられる質問

REST API 認証スコープを使用する際によく寄せられる質問を以下に示します。

1 つの OAuth トークンを複数の認証スコープにリンクすることはできますか。

はい。1 つの oauth_entity を複数の認証スコープとリンクすることができます。この oauth_entity によって発行されるすべての OAuth トークンに同じ認証スコープが設定されます。

認証スコープが異なる別々の OAuth トークンで同じ REST API にアクセスできますか。

はい。異なる認証スコープで同じ REST API に対してアクセスできます。1 つの認証スコープが一致する限り、認証スコープから結果が返されます。

useraccount 認証スコープのある OAuth アクセストークンで任意の REST API にアクセスできますか。

はい。useraccount には認証スコープへのフルアクセス権があります。

OAuth アクセストークンの OAuth スコープは動的に変更できますか。

はい。oauth_credential テーブルでは認証スコープがアクセストークンでハードコードされていません。代わりに、実行時にリンクされた oauth_entity から認証スコープが取得されます。

OAuth トークンは更新後も同じ認証スコープを保持できますか。

はい。oauth_admin が oauth_entityにリンクされた認証スコープを変更しない限り、トークンの更新後も認証スコープは変更されません。

事前定義された useraccount 認証スコープのレコードが削除されました。useraccount という名前の新しい認証スコープを作成できますか。

同じ useraccount で新しい認証スコープを作成することはできません。ランタイムでは、名前の代わりに sys_id を使用して認証スコープのチェックを行い、システムプロパティ *glide.oauth.token.scope.useraccount* を新しく作成された sys_id レコードに変更します。

oauth_entity にリンクされた認証スコープをアドミニストレーターが変更すると、この OAuth エンティティによって発行された既存のすべての OAuth アクセストークンも変更されますか。

はい。認証スコープは OAuth アクセストークンに直接リンクされておらず、実行時に oauth_entity から取得されます。

同じ oauth_entity によって発行された別々の OAuth アクセストークンに異なる認証スコープを設定することはできますか。

いいえ。トークンへのすべてのアクセスが同じ oauth_entity によって発行されているため、認証スコープは常に同じになります。

ユーザーが特定のエンドポイントに対して複数の異なる認証スコープを定義することはできますか。

いいえ。特定の REST API エンドポイントに対しては一意の制約チェックがあります。ただし、同じ REST API エンドポイントに対しては、複数の一致する認証スコープが存在する場合があります。

認証スコープのチェックは BasicAuth にも使用されますか。

いいえ、認証スコープのチェックは OAuth アクセストークンと OIDC トークンのみであり、*basicAuth* と *mutualAuth*には適用されません。

SOAP API アクセスポリシー

SOAP API アクセスポリシーを使用すると、認証タイプとアクセスポリシーの指定されたフィルター基準に基づいて、受信 SOAP API へのアクセスを制限できます。

SOAP API アクセスポリシーを使用すると、受信認証プロファイルと API アクセスポリシーを受信 SOAP およびスクリプト化された SOAP API に適用できます。

IP 範囲やロールベースの制限などの ServiceNow API アクセスポリシーを活用し、認証に基づいて受信 SOAP API 呼び出しを許可または禁止できます。

アドミニストレーターは、次のアクションを実行してポリシーを適用できます。

- SOAP API Access Policy プラグインとAuthentication Profile プラグインを有効にします。詳細については、「SOAP API アクセスポリシーのアクティブ化」を参照してください。
- SOAP API アクセスポリシーを作成し、それらのポリシーを認証プロファイルに関連付けます。詳細については、「SOAP API アクセスポリシーの作成」と「認証プロファイルの作成」を参照してください。

  注: ポリシーは、SOAP テーブル API またはスクリプト化された SOAP API に適用されます。標準の http および WSSE 認証プロファイルに加えて。

- IP 範囲、ロールベースの制限などの認証ポリシーを作成し、このポリシーを認証プロファイルに関連付けます。詳細については、「API 認証ポリシーの作成」を参照してください。

SOAP API アクセスポリシーのアクティブ化

SOAP API アクセスポリシーの場合は、SOAP API Access Policy (com.glide.soap.policy) プラグインをインストールします。

始める前に
必要なロール：admin

このタスクについて
次のアイテムは、SOAP API Access Policy プラグインとともにインストールされます： プロセッサーアクセスポリシー (com.glide.processor.policy)

依存プラグイン：Authentication Profile (com.glide.auth.profile)

手順

1. 移動先 すべて > システムアプリケーション > 利用可能なすべてのアプリケーション > すべて.
2. フィルター基準と検索バーを使用して、SOAP API Access Policy プラグイン (com.glide.soap.policy) を検索します。

   名前または ID でプラグインを検索できます。プラグインが見つからない場合は、ServiceNow 担当者から要求する必要があります。
3. [インストール] を選択して、インストールプロセスを開始します。

   注: ドメインセパレーションと代理アドミンがインスタンスで有効になっている場合、管理ユーザーはグローバルドメインに含まれている必要があります。それ以外の場合、次のエラーが表示されます： 「別の操作が実行されているため、アプリケーションのインストールは利用できません： <プラグイン名> のプラグインの有効化 (Application installation is unavailable because another operation is running: Plugin Activation for <plugin name>)」

   インストールが完了するとメッセージが表示されます。プラグインとともにインストールされるコンポーネントの詳細については、「 アプリケーションとともにインストールされているコンポーネントの検索 」を参照してください。

認証プロファイルの作成

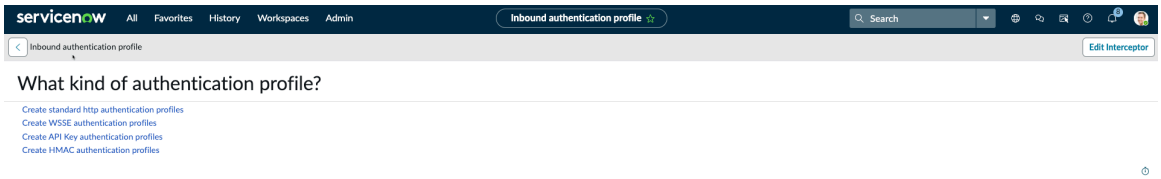
認証プロファイルを作成し、1 つ以上の認証ポリシーをプロファイルに追加します。デフォルトで利用可能な **ID** トークンおよび **OAuth** トークンの認証プロファイルを設定することもできます。

始める前に
必要なロール：admin

注: 相互認証とカスタマイズされた認証を使用して、認証ポリシー、IP 範囲、ロールベース、ユーザーベースなどを適用できます。

手順

1. 移動先 すべて > システム **Web** サービス > **API** アクセスポリシー > 受信認証プロファイル.
2. **[New** (新規)] を選択します。
   メッセージが表示されます。認証プロファイルの種類は？ (What kind of authentication profile?)
3. [認証プロファイルの種類は？ **(What kind of authentication profile?)]** を選択します。
   - 標準の **http** 認証プロファイルを作成
   - **WSSE** 認証プロファイルを作成



4. フォームのフィールドに入力します。

[標準の認証プロファイル] フォーム

フィールド	説明
名前	認証ポリシーを識別する名前
説明	認証ポリシーの説明
有効	認証ポリシーをアクティブにするオプション
アプリケーション	認証ポリシーの範囲
タイプ	利用可能な認証のタイプ。[基本認証]、**[ID トークン]**、[証明書ベースの認証]、**[OAuth]** または **[WSSE]** (WSSE 認証プロファイルの場合) を選択できます。
OAuth エンティティ	OAuth エンティティプロファイル。このフィールドは、[タイプ] で **[ID トークン]** または **[OAuth]** を選択した場合のみ表示されます。

5. [新規行を挿入] をダブルクリックします。

6. リストから認証ポリシーを選択し、保存アイコン ( ) を選択します。

> **注:** [アクセス許可ポリシー **(Allow Access Policy)]** または [アクセス拒否ポリシー **(Deny Access Policy)]** は選択しないでください。これらのポリシーは、ユーザーログインのみを対象としています。

1 つ以上のポリシーを認証プロファイルに追加できます。

認証プロファイルに変更がある場合は、認証ヘッダーが、その時点で行われた変更に固有の値を返します。「WWW-Authenticate」ヘッダーで返されるすべての認証スキーマを取得できるようにするには、glide.security.response.authenticate.header.auth_profile.first_scheme_only をアクティブ化して **false** に設定する必要があります。応答は複数のヘッダーを使用して返されます。例：

```
< WWW-Authenticate: BEARER realm="Service-now"
< WWW-Authenticate: BASIC realm="Service-now"
```

SOAP API アクセスポリシーの作成

API アクセスポリシーを作成し、認証プロファイルをマップして SOAP API の認証タイプを制限します。たとえば、SOAP API の ID トークン認証のみを許可する API アクセスポリシーを作成できます。

始める前に

- 認証プロファイルが作成されていることを確認します。詳細については、「認証プロファイルの作成」を参照してください。
- 必要なロール：admin

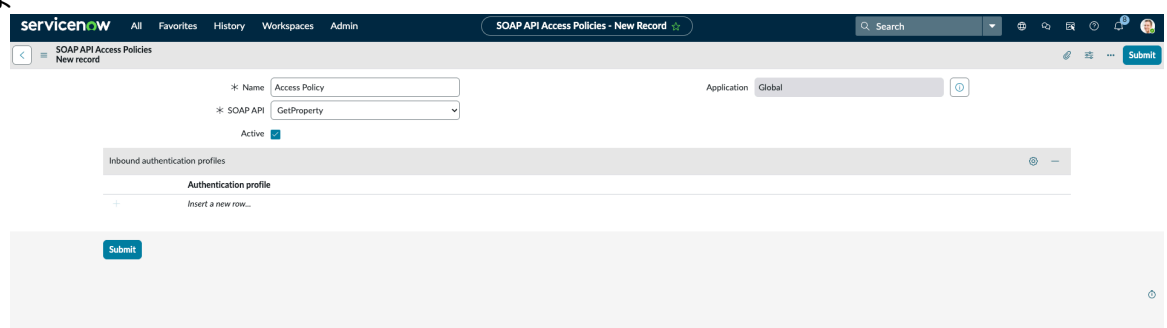
手順

1. 移動先 すべて > システム **Web** サービス > **API** アクセスポリシー > **SOAP API** アクセスポリシー.
2. [**New**] をクリックします。
3. フォームのフィールドに入力します。

[**API** アクセスポリシー] フォーム

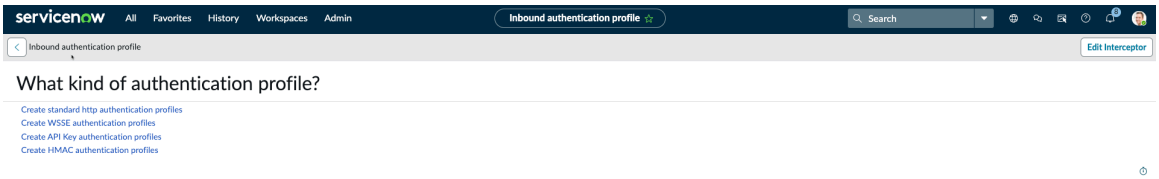
フィールド	説明
名前	API アクセスポリシーの一意の名前。
SOAP API	アクセスポリシーが適用される SOAP API。 例：**GetProperty API**。
アプリケーション	アプリケーションのスコープ。
有効	API アクセスポリシーをアクティブにするオプション。

4. [受信認証] セクションで、[新規行を挿入] をダブルクリックします。
5. リストから受信認証プロファイルを選択し、保存アイコン ( ) をクリックします。
   たとえば、基本認証、**ID** トークン、証明書ベースの認証、**OAuth**、または **WSSE** 認証 を追加できます。



   a. 1 つ以上の受信認証プロファイルを追加するには、[新規] をクリックして新しいプロファイルを作成します。

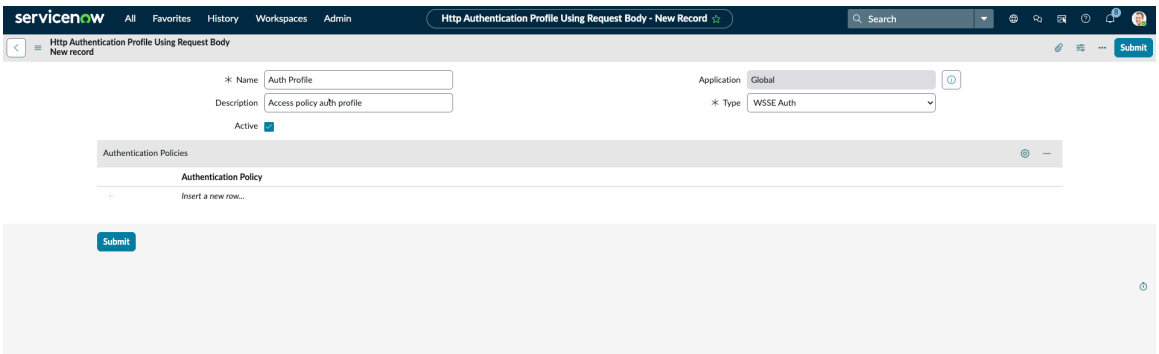
   b. [認証プロファイルの種類は？] を選択します。
      - 標準の **http** 認証プロファイルを作成
      - **WSSE** 認証プロファイルを作成



c. WSSE 認証プロファイルを作成するには、フォームのフィールドに入力します。

WSSE 認証プロファイル

フィールド	説明
名前	API アクセスポリシーの一意の名前。
説明	認証プロファイルの説明。
アプリケーション	アプリケーションのスコープ。
有効	API アクセスポリシーをアクティブにするオプション。
タイプ	認証プロファイル WSSE (Web セキュリティ) としての WSSE 認証。



d. 認証プロファイルを作成したら、レコードを保存します。

6. [送信] をクリックして SOAP API アクセスポリシーを送信します。

SOAP API を保護するためのグローバル **API** アクセスポリシーの作成

すべての SOAP API を保護する単一のグローバル API アクセスポリシーを作成します。

始める前に

- 必要なロール：admin
- **Processor Access policy** (com.glide.processor.policy) プラグインをインストールします。
- 認証プロファイルが作成されていることを確認します。詳細については、「認証プロファイルの作成」を参照してください。

次の手順では、すべての SOAP API を保護する単一のグローバル API アクセスポリシーを作成する方法について説明します。

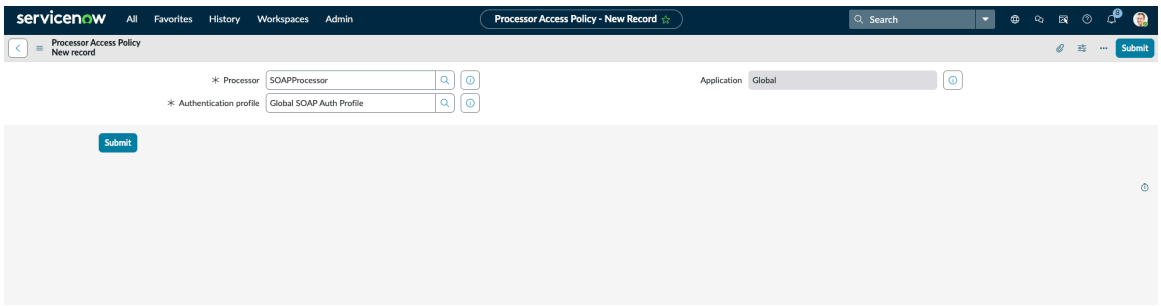
注: 個々の SOAP API レベルで定義されたポリシーは、**SOAPProcessor** レベルのグローバルアクセスポリシーを上書きします。

手順

1. 移動先 すべて > システムセキュリティ > プロセスアクセスポリシー.
2. フォームで、フィールドに入力します。

プロセスアクセスポリシーフォーム

フィールド	説明
プロセッサ	認証ポリシーを識別する名前。たとえば、**[SOAPProcessor]** (認証プロファイル) を選択します。
アプリケーション	認証ポリシーの範囲デフォルト：グローバル
認証プロファイル	認証プロファイルのタイプ。[グローバル **SOAP** 認証プロファイル **(Global SOAP Auth Profile)]** を選択します。



3. [送信] を選択します。

API のフィルター基準

フィルター基準には、認証ポリシーのポリシー入力として使用されるフィルター条件またはクエリーが含まれています。ポリシー入力は 1 つ以上のフィルター基準をグループ化し、認証ポリシーのポリシー条件を定義するために使用されます。たとえば、IP フィルター基準は、クラスレスドメイン間ルーティング (CIDR) 形式の IP アドレスまたは IP アドレス範囲を定義します。

ユーザーの IP アドレス、ロール、およびユーザーが属するユーザーグループに基づいて、フィルター基準を作成できます。

注: 適応認証モジュールから作成されたフィルター基準も使用します。詳細については、「適応認証」を参照してください。

適応認証のフィルター基準と同じプロセスを使用して、API のフィルター基準を作成できます。詳細については、「フィルター基準」を参照してください。

関連トピック

- IP フィルター基準の作成
- ロールフィルター基準の作成
- グループフィルター基準の作成

API 認証ポリシー

認証ポリシーは、指定されたポリシー条件に基づいて認証要求を評価し、一致する基準に応じてアクセスを許可または拒否します。

組み込みのグローバルブロックポリシーを使用するか、セキュリティ要件に従って認証ポリシーを作成できます。グローバルブロックポリシーは、指定されたフィルター基準に基づいてユーザーと API の認証要求を拒否します。

> **注:** [アクセスポリシーを許可] および [アクセスポリシーの拒否] を使用または変更しないでください。これらのポリシーは、ユーザーログインのみを対象としています。

API 認証ポリシーの作成

認証ポリシーを使用すると、指定されたフィルター基準に基づいて API のアクセス制限を適用できます。

始める前に

必要なロール：admin

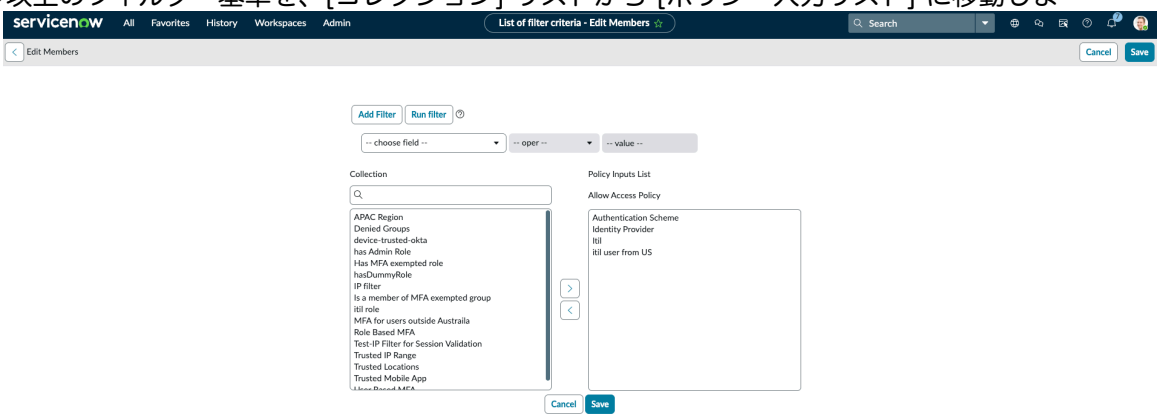
手順

1. 移動先 すべて > システム **Web** サービス > **API** 認証ポリシー.
2. [**New**] をクリックします。
3. フォーム上で、以下のフィールドに記入します。

ポリシーフォーム

フィールド	説明
名前	ポリシーを識別する名前。
説明	ポリシーの簡単な説明。
アプリケーション	アプリケーションのスコープ

4. フォームヘッダーを右クリックし、[保存] を選択します。
5. [ポリシーの入力] タブで、[編集] を選択して既存のフィルター基準を追加します。
   新しいポリシー入力を作成することもできます。詳細については、「ポリシー入力の作成」を参照してください。
6. 1 つ以上のフィルター基準を、[コレクション] リストから [ポリシー入力リスト] に移動します。



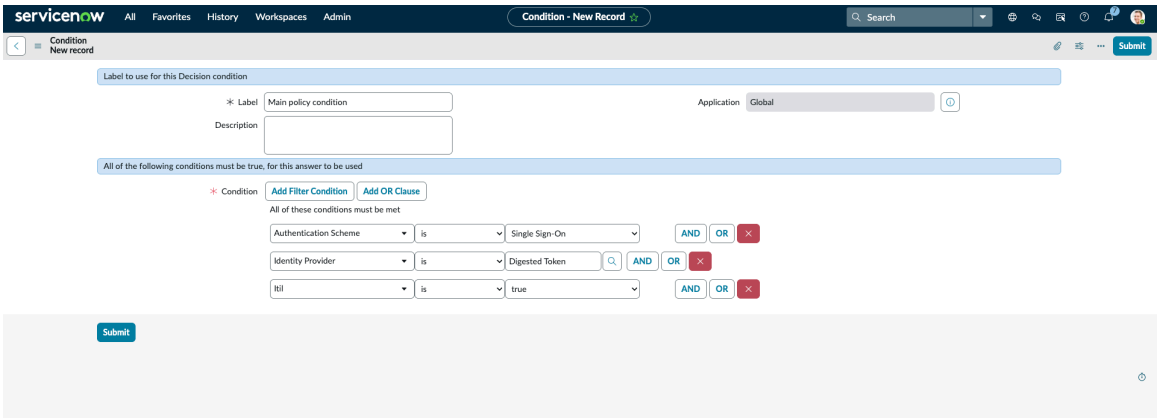
す。

7. [保存] を選択します。

8. [ポリシー条件] タブで、[新規] をクリックします。
9. フォーム上で、以下のフィールドに記入します。

[条件] フォーム

フィールド	説明
ラベル	ポリシー条件の名前
説明	ポリシー条件の簡単な説明
アプリケーション	アプリケーションのスコープ
条件	**OR** フィルターと組み合わされた 1 つ以上の条件。



10. [送信] を選択します。

API のグローバルブロックポリシーの設定

グローバルブロックポリシーは、指定されたフィルター条件に基づいてユーザーと API の認証要求を拒否します。このポリシーは、IP アドレスアクセス制御の代わりに使用できます。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > システム **Web** サービス > **API** アクセスポリシー > グローバルブロックポリシー.
2. [ポリシーの入力] タブで、[編集] をクリックします。
3. [コレクション] リストから 1 つ以上のフィルター基準を選択し、[グローバルブロックポリシー] リストに移動します。
   他のフィルターを追加することもできます。
4. [ポリシー条件] タブで、[新規] をクリックします。
5. フォームの各フィールドに入力します。

[条件] フォーム

フィールド	説明
ラベル	条件を識別する名前
説明	条件の説明

フィールド	説明
条件	認証要求を評価するために使用される複数のポリシー入力 (フィルター基準) の論理的な組み合わせ。たとえば、信頼できる IP アドレスのリストから請負会社のみを許可するような条件を作成できます。

システムプロセッサーまたはエクスポートプロセッサーのアクセスポリシー

システムプロセッサーまたはエクスポートプロセッサーがプロセッサーアクセスポリシーを活用してすべてのエクスポートエンドポイントを保護する機能。

すべてのエクスポートエンドポイントを保護し、グローバルまたはインスタンスレベルで受信認証プロファイルとプロセッサーアクセスポリシーを適用する機能。

注:

- アクセスポリシーでは、CSV、PDF などのエクスポートプロセッサーを含む非公開プロセッサーがサポートされています。
- アクセスポリシーではスクリプトプロセッサーもサポートされています。

サンプルユースケース

- admin は、API アクセスポリシーを利用することで、RSS プロセッサーを使用しない場合にそれをブロックできます。
- admin は、ベーシック認証を使用して認証プロファイルを作成し、常に false と評価される認証ポリシーを関連付けることができます。

  例：0.0.0.0 ～ 255.255.255.255 の範囲で IP 基準を作成し (Ipv6 アドレススペースも追加します)、false 演算子を使用してポリシー条件を追加します。このようにすると、ポリシー条件は常に false と評価され、API アクセスポリシーは要求の送信元に関係なくアクセスをブロックします。
- 信頼できるネットワークからのみプロセッサーへのアクセスを許可します。

プロセッサアクセスポリシーのアクティブ化

プロセッサーの場合は、Processor Access policy (com.glide.processor.policy) プラグインをインストールします。

始める前に
必要なロール：admin

このタスクについて
次のアイテムはプロセッサーアクセスポリシーとともにインストールされます。

- システムプロパティ：com.glide.auth.profile.supported.processor.list
- ナビゲーションのモジュール：プロセッサーアクセスポリシー

手順

1. 移動先 すべて > システムアプリケーション > 利用可能なすべてのアプリケーション > すべて.
2. フィルター基準と検索バーを使用して、Processor Access policy (com.glide.processor.policy) プラグインを検索します。

   名前または ID でプラグインを検索できます。プラグインが見つからない場合は、ServiceNow 担当者から要求する必要があります。

3. [インストール] を選択して、インストールプロセスを開始します。

   注: ドメインセパレーションと代理アドミンがインスタンスで有効になっている場合、管理ユーザーはグローバルドメインに含まれている必要があります。それ以外の場合、次のエラーが表示されます： 「別の操作が実行されているため、アプリケーションのインストールは利用できません： <プラグイン名> のプラグインの有効化 (Application installation is unavailable because another operation is running: Plugin Activation for <plugin name>)」

インストールが完了するとメッセージが表示されます。プラグインとともにインストールされるコンポーネントの詳細については、「 アプリケーションとともにインストールされているコンポーネントの検索 」を参照してください。

次のタスク

。
プロセッサーの認証プロファイルを設定します。詳細については、「プロセッサーの認証プロファイルの設定」を参照してください。

プロセッサーの認証プロファイルの設定

エクスポートプロセッサーの認証プロファイルを適用します。

始める前に
必要なロール：admin

必要なプラグイン：Processor Access Policy (com.glide.processor.policy)

手順

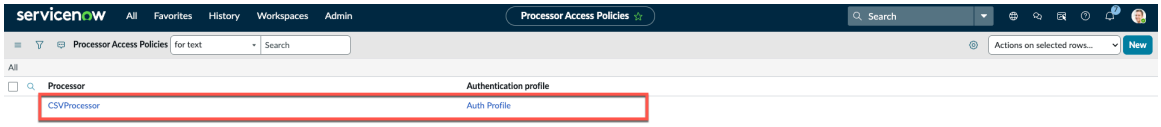
1. 移動先 すべて > システムセキュリティ > プロセッサアクセスポリシー.
2. プロセッサーに認証プロファイルを追加するには、[新規] をクリックします。
3. フォームのフィールドに入力します。

プロセスアクセスポリシーフォーム

フィールド	説明
プロセッサ	認証ポリシーを識別する名前。 注: 公開プロセッサーはサポートされていません。サポートされていないプロセッサーを選択すると、プロセッサーの送信中にエラーが表示されます。
アプリケーション	認証ポリシーの範囲デフォルト：グローバル
認証プロファイル	認証プロファイルのタイプ。**[ID** トークン] または **[OAuth** トークン] を選択できます。

4. [送信] をクリックします。
   認証プロファイルがプロセッサーに適用されます。

たとえば、CSV プロセッサーに対して OAuth 認証プロファイルが構成されているとします。この場合、エクスポートオプションとして CSV を使用し、OAuth アクセストークンを使用する必要があります。



証明書ベースの認証

証明書ベースの認証では、信頼できる認証局 (CA) からの証明書を使用して、ユーザーのログインまたは受信 API 要求を相互に認証できます。

注: 証明書ベースの認証は、オンプレミスおよびエッジ暗号化が有効なインスタンスではサポートされていません。

ユーザーインターフェイスログインのための証明書ベースの認証

エンドユーザーが、ユーザー名とパスワードを使用する代わりに PIV (個人 ID 検証) または CAC (Common Access Card) カードを使用して ServiceNow AI Platform または サービスポータル にログインできるようにします。ユーザーインターフェイスログインの相互認証を設定するには、「証明書ベースの認証の設定」を参照してください。

証明書ベースの認証が設定されると、エンドユーザーは設定を完了してログインできます。「証明書ベースの認証を使用したログイン」を参照してください。

受信 Web サービスの証明書ベースの認証

ServiceNow SOAP および REST API への受信要求を認証します。受信 Web サービスの相互認証を設定するには、「証明書ベースの認証の設定」を参照してください。

証明書ベースの認証の設定

ユーザーインターフェイスベースのログインまたは受信 Web サービスの相互認証を設定します。

始める前に
必要なロール：admin

インスタンスで ADCv2 ロードバランサーを使用していることを確認します。詳細は、ADCv2 移行に関するナレッジ記事 を参照してください。インスタンスで ADCv2 ロードバランサーを使用していない場合は、Now Support にお問い合わせください。

手順

証明書ベースの認証を設定する目的：

自動翻訳

- エンドユーザーが PIV または CAC カードを使用して ServiceNow AI Platform または サービスポータル に安全にログインできるようにするため。証明書ベースの認証を有効にした後、PEM 証明書を自己登録するか、アドミニストレーターが証明書をマッピングすることができます。「証明書ベースの認証を使用したログイン」を参照してください。
- 受信 Web サービスの相互認証を有効にするため。証明書ベースの認証が設定されると、システムでは提供された証明書を使用して、ServiceNow REST API および SOAP API にアクセスするために要求を相互認証します。

証明書ベースの認証のアクティブ化

admin ロールを持っている場合は、 ServiceNow AI Platform の 証明書ベースの認証プラグイン (com.glide.auth.mutual) をアクティブ化できます。

始める前に
必要なロール：admin

このタスクについて
次の テーブルが証明書ベースの認証とともにインストールされます。

- sys_user_certificate
- sys_ca_certificate
- sys_ca_certificate_api_track

手順

1. 移動先 すべて > システムアプリケーション > 利用可能なすべてのアプリケーション > すべて.
2. フィルター基準と検索バーを使用して Certificate-based authentication プラグイン (com.glide.auth.mutual) を検索します。

   名前または ID でプラグインを検索できます。プラグインが見つからない場合は、ServiceNow 担当者から要求する必要があります。
3. [インストール] を選択して、インストールプロセスを開始します。

   注: ドメインセパレーションと代理アドミンがインスタンスで有効になっている場合、管理ユーザーはグローバルドメインに含まれている必要があります。それ以外の場合、次のエラーが表示されます： 「別の操作が実行されているため、アプリケーションのインストールは利用できません： <プラグイン名> のプラグインの有効化 (Application installation is unavailable because another operation is running: Plugin Activation for <plugin name>)」

   インストールが完了するとメッセージが表示されます。プラグインとともにインストールされるコンポーネントの詳細については、「 アプリケーションとともにインストールされているコンポーネントの検索 」を参照してください。

CA 証明書の登録

ルート証明書または中間証明書を登録して、認証に使用できるようにします。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > 証明書ベースの認証 > **CA** 証明書チェーン.
2. [**New**] をクリックします。

3. フォームのフィールドに入力します。

[相互認証 **CA** 外部審査済み書] フォーム

フィールド	説明
名前	証明書を識別する名前
有効期限通知	証明書の有効期限が近づいたときにユーザーに警告するオプション。
期限切れ時に通知	証明書の有効期限が切れたときに通知を受信するユーザーのリスト。
有効期限切れ前に警告	証明書の有効期限が切れる前に通知を送信する日数。
有効	クライアント証明書をアクティブにするオプション。
フォーマット	PEM
タイプ	証明書のタイプ。次のオプションが含まれます。 ○ **CA** 証明書：ルート CA 証明書。チェーンに中間証明書を含めることもできます。CA 証明書は自動的にロードバランサーと同期されます。できるだけこのオプションを使用して、チェーン内の必要な証明書が欠落しないようにしてください。 ○ 中間証明書：証明書チェーン内の中間証明書。この証明書はインスタンスにのみ残り、ロードバランサーとは同期されません。このオプションは、既存のチェーンに中間証明書を追加する必要がある場合にのみ使用します。
簡単な説明	ユーザーのクライアント証明書の簡単な説明。

注: 証明書のアップロード中に、読み取り専用フィールドの [有効開始日]、[有効期限]、[数日中に期限切れ **(Expires in days)**]、[発行者]、[件名]、[証明書チェーン]、および **[PEM** 証明書] が抽出されて自動入力されます。

4. [送信] をクリックします。
5. オプション: 証明書を検証するには、[ストア/証明書を検証] をクリックします。

PEM 証明書をユーザーにマップ

PEM 証明書をユーザーにマップし、ユーザーが PIV または CAC カードを使用してログインしたり、受信要求を認証したりできるようにします。複数の PEM 証明書をユーザーにマップできます。

始める前に

- 必要なロール：admin
- ユーザーのプライバシー拡張メール (PEM) 証明書があることを確認します。

注: PEM 証明書をユーザー構成にマッピングすると、「証明書の検証」が失敗します。これは、PEM 証明書が保存されていないためです。

手順

1. 移動先 すべて > 証明書ベースの認証 > ユーザーから証明書へのマッピング をクリックし、[新規] をクリックします。
2. フォームのフィールドに入力します。

[ユーザークライアント外部審査済み書] フォーム

フィールド	説明
名前	ユーザーのクライアント証明書の名前
有効期限通知	証明書の有効期限が近づいたときにユーザーに警告するオプション。
有効期限切れ前に警告	証明書の有効期限が切れる前に通知を送信する日数。
期限切れ時に通知	証明書の有効期限が切れたときに通知を受信するユーザーのリスト。
有効	クライアント証明書をアクティブにするオプション。
ユーザー	クライアント証明書にマップされているユーザー。 システムは、受信要求または証明書登録からクライアント証明書を受信した後、このフィールドで指定されたユーザーを使用して要求を実行するセッションを開始します。
簡単な説明	ユーザーのクライアント証明書の簡単な説明。
フォーマット	Privacy Enhanced Mail (PEM) 形式は、base-64 でエンコードされた Distinguished Encoding Rules (DER) 証明書です。
タイプ	クライアント証明書。このフィールドは読み取り専用です。

注: 証明書のアップロード中に、読み取り専用フィールドの [有効開始日]、[有効期限]、[数日中に期限切れ **(Expires in days)**]、[発行者]、および [件名] が抽出されて自動入力されます。

3. 添付ファイルアイコンをクリックして証明書をアップロードします。
4. [送信] をクリックします。
   証明書が信頼できる認証局 (CA) からのものである場合、証明書が検証され、指定されたユーザーにマップされます。

証明書ベースの認証プロパティの設定

システムプロパティを使用して、証明書ベースの認証機能を有効または無効にします。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > 証明書ベースの認証 > プロパティ.
2. フォームのフィールドに入力します。

[証明書ベースの認証プロパティ **(Certificate Based Authentication Properties)**] フォーム

プロパティ	説明
外部審査済み書ベースの認証を有効にする	ユーザーインターフェイスのログインと受信 Web サービスの両方に対して証明書ベースの認証を有効にするオプション

プロパティ	説明
	デフォルト：true
ログイン画面に [PIV / CAC でログイン (Log in with PIV/CAC)] オプションを表示します	ログイン画面に **[PIV/CAC** カードでログイン] オプションを表示します。ユーザーがユーザーインターフェイスを使用して証明書ベースの認証を使用できるようにします。 デフォルト値：false
証明書ベースのログインの自動リダイレクトを有効にします	登録された証明書を選択して PIN を入力した後に、ユーザーが **[PIV/CAC** カードでログイン] をクリックする必要があるかどうかを決定します。アクティブ化すると、登録されたクライアント証明書を選択して PIN を入力した後に、ユーザーが自動的にログインされます。非アクティブ化すると、登録されたクライアント証明書を選択して PIN を入力した後に、ユーザーは **[PIV/CAC** カードでログイン] をクリックする必要があります。 デフォルト値：false

証明書ベースの認証を使用したログイン

アドミニストレーターが証明書ベースの認証を設定すると、PIV (Personal Identity Verification) または CAC (Common Access Card) カードを使用してクライアント証明書を登録し、ログインできるようになります。

PIV または **CAC** カードのクライアント証明書の登録

PIV または CAC カードを使用して ServiceNow AI Platform にログインする前に、PIV または CAC カードのクライアント証明書を登録する必要があります。クライアント証明書を登録できない場合は、アドミニストレーターにお問い合わせください。アドミニストレーターは、PIV または CAC カードのクライアント証明書を登録することもできます。

始める前に

- 証明書ベースの認証が有効になっていることを確認します。
- カードリーダーがコンピューターに接続され、PIV または CAC カードが挿入されていることを確認します。
- 必要なロール：なし

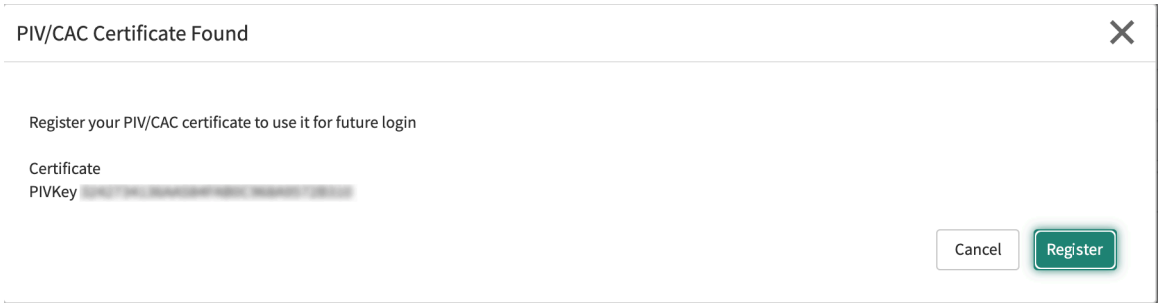
このタスクについて

クライアント証明書を登録するために admin が必要な場合は、「PEM 証明書をユーザーにマップ」を参照してください。

手順

1. ユーザー名とパスワードを使用して ServiceNow AI Platform にログインします。
2. ユーザーメニューから自分の名前をクリックし、[プロファイル] を選択します。
3. [関連リンク] から、[クライアント証明書を登録] をクリックします。

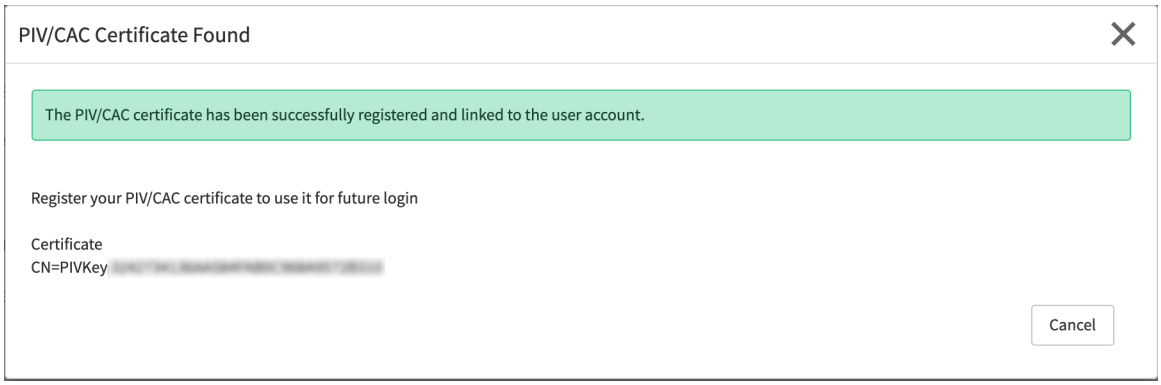
   有効な証明書が利用可能な場合は、次のメッセージが表示されます。



4. [登録] をクリックします。

   登録に成功すると、次のメッセージが表示されます。

   PIV/CAC 証明書が正常に登録され、ユーザーアカウントにリンクされました。



次回 ServiceNow AI Platform にログインするときは、PIV または CAC カードを使用してログインできます。詳細については、「PIV または CAC カードを使用して ServiceNow AI Platform にログイン」を参照してください。

PIV または CAC カードを使用して ServiceNow AI Platform にログイン

ServiceNow AI Platform で証明書ベースの認証が有効になっている場合は、ユーザー名とパスワードの代わりに PIV または CAC カードを使用してログインできます。

始める前に

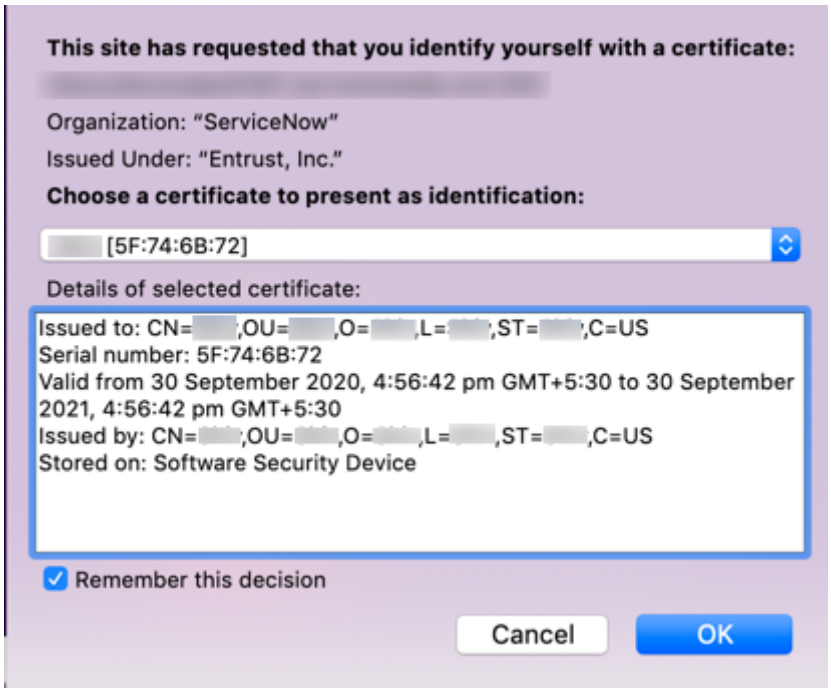
- 必要なロール：なし
- 証明書ベースの認証が有効になっていることを確認します。
- PIV または CAC のカードリーダーがコンピューターに接続されていることを確認します。
- PIV または CAC カードのクライアント証明書がマップされていることを確認します。詳細については、「CA 証明書の登録」を参照してください。

手順

1. PIV または CAC カードをカードリーダーに挿入します。
2. ブラウザーでインスタンスに移動します。
   PIV または CAC カードの PIN を入力するよう求めるプロンプトがブラウザーに表示されます。
3. ブラウザーのプロンプトに PIV または CAC カードの PIN を入力します。

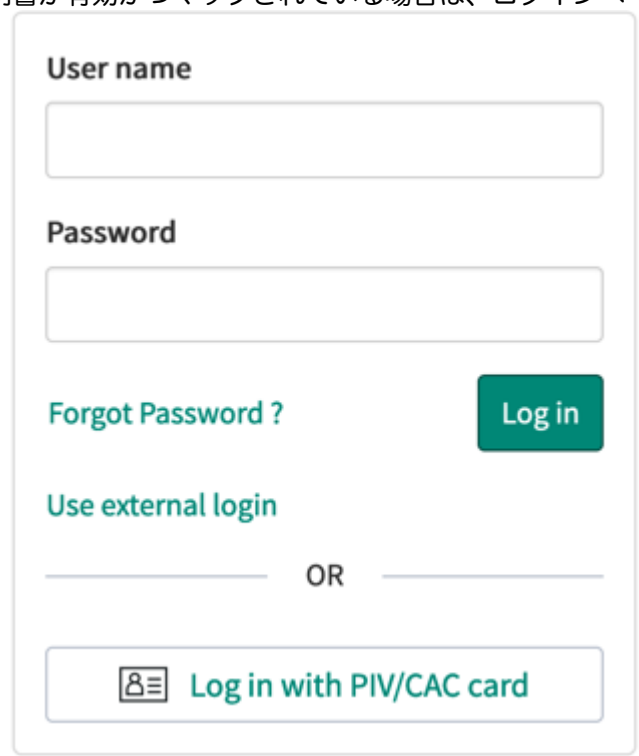
   注: PIN を忘れた場合は、アドミニストレーターにお問い合わせください。

4. 正しい PIN を入力すると、証明書を選択するためのプロンプトがブラウザーに表示されます。



5. ブラウザープロンプトから証明書を選択します。

   証明書が有効かつマップされている場合は、ログインページにリダイレクトされま



   す。

6. **[PIV/CAC** カードでログイン] ボタンをクリックします。

   ServiceNow AI Platformからログアウトするには、カードリーダーから PIV または CAC を取り出して、ブラウザーを閉じる必要があります。

クライアント証明書の管理

アカウントに関連付けられているクライアント証明書を表示および削除します。

始める前に

- 証明書ベースの認証が有効になっていることを確認します。
- 必要なロール：なし

手順

1. ユーザー名とパスワードを使用して ServiceNow AI Platform にログインします。
2. ユーザーメニューから自分の名前をクリックし、[プロファイル] を選択します。

3. [関連リンク] から、[クライアント証明書を管理] をクリックします。
   ユーザークライアント証明書 [sys_user_certificate] テーブルが開き、アカウントに関連付けられている証明書が表示されます。
4. アカウントに関連付けられている証明書を表示または削除します。

カスタムインスタンス URL

会社ブランドまたはカスタム URL から ServiceNow インスタンスにアクセスできるようにすることができます。

カスタム URL の概要

カスタム URL を使用すると、1 つまたは複数の会社ブランドやカスタム URL から ServiceNow インスタンスにアクセス可能にすることができます。

カスタム URL は特定のポータルに関連付けることができます。たとえば、アドミンであれば CSM ポータルでは http://support.acme.com を、人事ポータルでは http://hr.acme.com を定義できます。このような状況では、ユーザーがアクセスしているカスタム URL に基づいて、ユーザーを認証のため別の IdP にリダイレクトする必要があります。

重要:

- インスタンスあたりのドメイン数が 100 を超えるカスタム URL を作成しないでください。
- 最初に ServiceNow インスタンスからカスタム URL レコードを削除してから、任意のドメインネームサーバー (DNS) エントリを DNS サーバーから削除する必要があります。
- カスタム URL レコードを ServiceNow インスタンスから削除する前に DNS エントリを DNS サーバーから削除すると、ServiceNow から対応する他のカスタム URL レコードの削除がブロックされる可能性があります。

Tokyo 以降は、カスタム URL レコードで定義された指定の IdP にユーザーが自動でリダイレクトできるようにすることができます。

注: カスタム URL は、オンプレミスの顧客または開発者インスタンスには使用できません。また、URL は公開されている必要があります。

トップレベルドメイン (TLD) または任意のサブドメインの所有者のみが、DNS サブドメインにカスタム URL を設定できます。たとえば、インスタンスに次のような指定 URL と追加のカスタム URL があるとします。

カスタム **URL** の例

URL 例	使用法
https://acme.service-now.com	ServiceNow インスタンスに付属する Acme の初期ドメイン名。
https://support.acme.com	ServiceNow インスタンスに関連付けられているカスタム URL。この URL は、初期ドメイン名のエイリアス (CNAME) と呼ばれます。
https://US-support.acme.com	インスタンスのサービスポータルに関連付けられている予備のカスタム URL。インスタンスは、同じサービスポータルに対するカスタム URL を複数サポートできます。

インスタンス外のカスタム **URL** に関する考慮事項

カスタム URL を関連付けるには、その前にドメインプロバイダーを介して URL を所有 (または購入) する必要があります。また、カスタム URL を作成してインスタンスに関連付ける前に、特定の構成も必要になります。

カスタム **URL** 構成

構成アイテム	説明
プロバイダーでの CNAME 設定 (Set the CNAME with the provider)	CNAME レコードは ServiceNow インスタンス URL として設定する必要があります。
専用 VIP ステータスの確認 (Determine your dedicated VIP status)	VIP のステータス

注: インスタンスを指す CNAME レコードを削除または更新する場合は、インスタンス内でレコードが不安定な状態にならないよう、このシーケンスに従う必要があります。まずインスタンスから CNAME レコードを削除し、次に DNS プロバイダーで CNAME 設定を削除または更新します。

カスタム **URL** のアクティブ化

ServiceNow インスタンスでカスタム URL を設定できるようにします。アドミンロールを持っている場合は、カスタム URL プラグイン (com.snc.customurl) をアクティブ化することができます。

始める前に

必要なロール：admin

手順

1. 移動先 すべて > システム定義 > プラグイン.
2. [カスタム **URL]** プラグインを検索してクリックします。

注: [カスタム **URL - Internal]** プラグインは選択しないでください。これはスクリプト化されたカスタム URL API の内部コンポーネントです。

3. カスタム URL レコードで、[アクティブ化/修復] 関連リンクをクリックします。
4. [プラグインアクティベーション] ウィンドウで、[アクティブ化 **(Activate)]** をクリックします。[プラグインのアクティベーション] ウィンドウが再度開き、プラグインがアクティブ化されたというメッセージが表示されたら、[フォームを閉じて再ロード] をクリックしてこのフォーム に留まります。
5. [プラグインファイル] 関連リストで、次のプロパティを検索して設定値を変更します。

オプション	説明
glide.customurl.enabled	カスタム URL を有効にするには、値を **True** に設定します。デフォルトでは、このプロパティは **False** に設定されています。この場合、カスタム URL を関連付けることはできません。 注: この機能を無効にするには、 プロパティを **False** に設定します。

6. [更新] をクリックします。

カスタム **URL** をインスタンス **URL** として設定

ServiceNow URL の代わりにカスタム URL を使用するインスタンス構成に追加します。

始める前に
必要なロール：admin
カスタム URL をインスタンスに追加する前に、カスタム URL プラグインをアクティブ化して URL を購入または登録する必要があります。

手順

1. 移動先 すべて > カスタム **URL** > カスタム **URL**.
2. 次のいずれかを実行する必要があります。
   - [新規] をクリックして、インスタンスに新しいドメイン名を関連付けます。
   - インスタンス URL として設定するために、すでに設定されているカスタム URL を選択します。
3. 適宜フィールドに入力します。

カスタム **URL** フィールド

フィールド	説明
ドメイン名	カスタム URL の完全修飾ドメイン名 (FQDN) FQDN は、カスタムドメインのネームサーバーレコードに作成された CNAME リダイレクトです。

フィールド	説明
	注: たとえば、acme.com のネームサーバーで、次のようにエントリを作成します。 `support.acme.com 300 IN CNAME acme.servicenow.com`
インスタンス URL である	すべての送信 URL に対してこのカスタム URL を有効にするチェックボックス。アクティブなカスタム URL を 1 つだけインスタンス URL にすることができます。 カスタム URL に対してこの設定を有効にするには、URL レコードの [インスタンス **URL** として設定] をクリックします。以前のカスタム URL はすべて削除されます。
ステータス	カスタム URL レコードのステータス。ステータスが [アクティブ] の場合、カスタム URL はプロビジョニングされ、使用する準備ができています。
サービスポータル	カスタム URL を使用してユーザーをインスタンスにリダイレクトするときに使用するサービスポータル
ID プロバイダー	カスタム URL の ID プロバイダーでは、カスタム URL レコードで定義された指定の IdP にユーザーが自動でリダイレクトできるようにすることが可能です。

カスタム URL は、インスタンスで 6 時間以内にアクティブ化する必要があります。

注:

- 最初に ServiceNow インスタンスからカスタム URL レコードを削除してから、任意のドメインネームサーバー (DNS) エントリを DNS サーバーから削除する必要があります。
- カスタム URL レコードを ServiceNow インスタンスから削除する前に DNS エントリを DNS サーバーから削除すると、ServiceNow から対応する他のカスタム URL レコードの削除がブロックされる可能性があります。

ID プロバイダーを含むカスタム URL

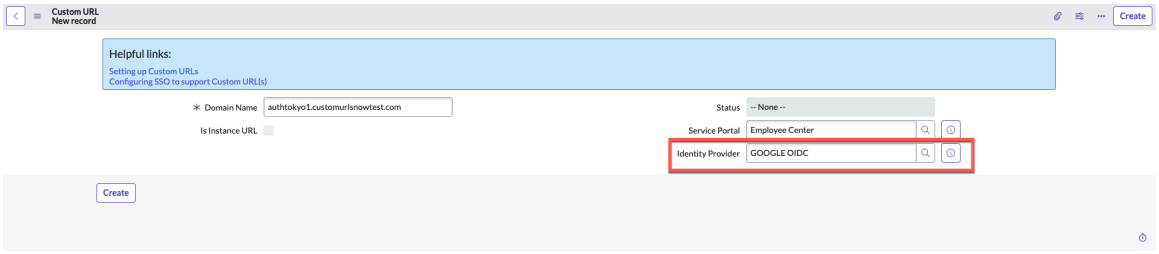
ID プロバイダーを含むカスタム URL を設定し、ユーザーが自身の IdP でログインできるようにします。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > カスタム **URL** > カスタム **URL**.
2. [**New**] をクリックします。
3. **[ID** プロバイダー] フィールドに IdP の詳細を入力します。

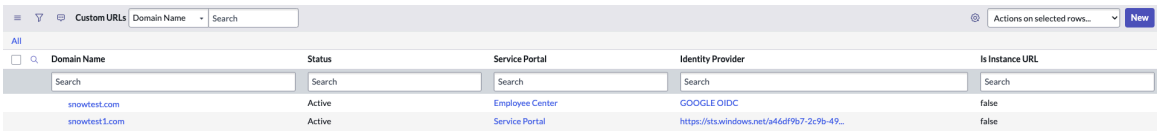
その他のフィールドの詳細については、「カスタム URL をインスタンス URL として設定」を参照してくださ



い。

4. [作成] をクリックします。

レコードが作成され、[カスタム URL] ページに表示されます。



ユーザーがカスタム URL にアクセスすると、設定されている ID プロバイダーにリダイレクトされます。このケースでは、snowtest.com にアクセスすると、ユーザーは従業員センターに移動し、Google ID プロバイダーにリダイレクトされます。

注:

- [サービスポータル] フィールドが空で [ID プロバイダー] フィールドが定義されている場合、ユーザーがカスタム URL にアクセスすると、ユーザーは設定された ID プロバイダーへ直接移動されします。
- [サービスポータル] フィールドと [ID プロバイダー] フィールドがどちらも定義されている場合、サービスポータル が定義されているカスタム URL にユーザーがアクセスすると、ユーザーは設定された ID プロバイダーへ移動します。
- [サービスポータル] フィールドと [ID プロバイダー] フィールドがどちらも定義されている場合、別のポータルが定義されているカスタム URL にユーザーがアクセスすると、自動リダイレクト ID プロバイダーがインスタンス上で設定されていれば、ユーザーはそこへ移動します。

5. 認証情報を使用してアプリケーションにログインします。

カスタム URL データセンタージョブ情報

インスタンスに関連付けられているすべてのカスタム URL には、対応する ServiceNow データセンタージョブがあり、このジョブが実行されて、インスタンスに関連する URL 情報が表示されます (表を参照)。

ジョブフィールド	説明
ジョブ ID	カスタム URL のドメインを確認するジョブの一意の ID
前回実行日時	ジョブが実行された最新の日時
ペイロード	証明書プロビジョニングのためにデータセンターに送信されたドメインまたはカスタム URL のリスト

ジョブフィールド	説明
ポーリング数	このジョブに対して結果がポーリングされた回数
結果	ペイロードで送信された各ドメインまたはカスタム URL を検証します。
ステータス	データセンタージョブのステータス

SAML/SSO カスタム URL インストール用の SP メタデータの生成

SAML または SSO のインストールでは、カスタム URL インスタンスが生成される前に、IdP に対して生成された SP メタデータが必要です。

始める前に
必要なロール：admin
IdP には、要求を認証して転送するために、インスタンスの SP メタデータが必要です。

注: Assertion Consumer Service URL (SP ログイン URL) の追加は、IdP (Azure、ADFS、または Okta) ごとに異なる場合があります。

手順

1. インストールされている SSO プラグインを選択します。

オプション	説明
マルチプロバイダー SSO	移動先 マルチプロバイダー **SSO** > **ID** プロバイダー. IdP を選択し、[メタデータを生成] ボタンをクリックします。統合により、システムプロパティ設定からインスタンスの SP メタデータが自動的に生成されます。
SAML 2 SSO	移動先 **SAML 2** シングルサインオン > メタデータ. 統合により、システムプロパティ設定からインスタンスの SP メタデータが自動的に生成されます。

2. テキストボックスに SP メタデータをコピーします。

例：

```
<EntityDescriptor xmlns="urn:oasis:names:tc:SAML:2.0:metadata"
 entityID="https://yourinstance.service-now.com">
  <SPSSODescriptor AuthnRequestsSigned="false" WantAssertionsSigned="true"
 protocolSupportEnumeration="urn:oasis:names:tc:SAML:2.0:protocol">
  <SingleLogoutService Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect"
 Location="https://yourinstance.service-now.com/navpage.do" />

 <NameIDFormat>urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress</NameIDFor
mat>
  <AssertionConsumerService isDefault="true" index="0"
 Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
 Location="https://yourinstance.service-now.com/navpage.do" />
  <AssertionConsumerService index="1"
 Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
 Location="https://yourinstance.service-now.com/consumer.do"/>
```

```
</SPSSODescriptor>
</EntityDescriptor>
```

3. インスタンス SP メタデータを IdP に提供します。
   たとえば、SSOCircle を使用すると、ユーザーは SP メタデータをオンラインで提供できます。
4. オプション: Azure でカスタム URL を設定するには：
   a. [アプリの登録] に移動します。
   b. メニューから [すべてのアプリ] を選択します。
   c. **[ServiceNow** アプリ] を選択します。
   d. [設定] をクリックして URL を構成します。
5. オプション: Okta でカスタム URL を設定するには：
   a. 2 つの ServiceNow UD Okta アプリケーションを作成します。
   b. 「service-now.com」インスタンス URL 用の 1 つの Okta アプリケーション。
   c. カスタム URL 用の 1 つの Okta アプリケーション。

   注:
   - テスト接続を正常に実行するには、Okta 構成内で [強制認証の無効化 **(Disable Force Authentication)]** を無効にします。
   - ベース URL に関連付けられた ID プロバイダーレコードをテストしている場合は、ベース URL を使用してインスタンスにログインする必要があります。
   - カスタム URL に関連付けられた ID プロバイダーをテストしている場合は、カスタム URL を使用してインスタンスにログインする必要があります。
6. オプション: OAuth 認証を使用するには、外部クライアントアプリケーションの OAuth アプリケーションエンドポイント構成で、登録されたすべてのカスタム URL としてリダイレクト URL を設定します。
   リダイレクト URL は、認証サーバーがリダイレクトするコールバック URL と同義です。
7. オプション: Google reCAPTCHA サービスを使用するには、API キーペアを設定します 。

カスタム **URL** のエラーと修正

カスタム URL のセットアップと構成に関する一般的なエラーと関連する修正のリストを以下に示します。

セットアップ中のエラー

エラーメッセージ	修正
カスタム URL を作成できません。(Unable to create a Custom URL.) 後でもう一度お試しください。(Try again later.)	制御範囲外の問題があり、カスタム URL を作成できない可能性があります。サポートに連絡する前に、時間をおいてから実行してみてください。

自動翻訳

エラーメッセージ	修正
	注: 通常、カスタム URL の作成には 30 分かかります。それ以上かかる場合は、https://support.servicenow.com/now?draw=case にお問い合わせください。
インスタンスに対する別のカスタム URL 要求がまだ処理中であるため、新しいカスタム URL 要求を送信できません。	新しい要求を送信する前に、自分のカスタム URL ジョブのステータスを確認してください。
インスタンス URL を設定する前に、プロパティ Glide Proxy、Glide Servlet をクリアする必要があります。(You must clear the following properties before the instance URL can be set: Glide Proxy, Glide Servlet.)	
<custom_domain> のプロビジョニングはまだ処理中です。(The provisioning of <custom_domain> is still in progress.) このプロセスには最大 X 時間かかる場合があります。(This process can take up to X hours.) このプロセスが完了すると、インスタンスアドミニストレーターに通知が送信されます。(Your instance administrator will receive a notification when this process completes.)	カスタム URL のインスタンスでプロビジョニングが開始された後、ステータスが変更される前に、プロセスの完了を待つ必要があります。
<custom_domain> が新しいインスタンス URL として設定されました。<base.service-now.com> はサービス中ですが、通知などに含まれる新しい URL には、すべて <custom_domain> が使用されます。(<custom_domain> is now set as the new instance URL. <base.service-now.com> is still in service, but all new URLs, such as in notifications, will use <custom_domain>.)	インスタンス URL として指定できる URL は 1 つだけです。インスタンスに関連付けられている他の URL をアクティブにすることはできますが、通知を使用できるのはインスタンス URL のみです。
カスタム URL <custom_domain> はインスタンス構成からすぐに削除され、<base.service-now.com> に戻るように設定されています。<custom_domain> は、DNS の CNAME レコードが設定されているかぎり、このインスタンスに関連したままになります。(The Custom URL <custom_domain> is set to be immediately removed from the instance configuration and revert back to <base.service-now.com>. <custom_domain> continues an association to this instance as long as the CNAME record in your DNS is set.)	このメッセージは、カスタム URL を変更する意図があることを確認します。この確認を受け入れることで、インスタンスへの URL の変更を開始します。[ドメイン名] リストにある URL はいずれもカスタム URL としてアクティブにすることができますが、プロバイダーの CNAME レコードから削除した場合は例外です。
glide.servlet.uri を変更することはできません。(You cannot modify glide.servlet.uri.) このプロパティはカスタム URL によって設定されます。	

エラーメッセージ	修正
glide.proxy.host を変更することはできません。(You cannot modify glide.proxy.host.) このプロパティはカスタム URL によって設定されます。	
<custom_domain> の CNAME レコードは <base.service-now.com> をポイントしません。(The CNAME record for <custom_domain> does not point to <base.service-now.com>.)	URL プロバイダー側の構成が正しくありません。CNAME レコードの設定を確認してください。
<custom_domain> の CNAME レコードがありません。(Missing CNAME record for <custom_domain>.)	インスタンスにカスタム URL を設定する前に、URL プロバイダーから CNAME レコードを設定する必要があります。

インストレーションイグジット

インストレーションイグジットは、Java からイグジットしてスクリプトを呼び出してから Java に戻るカスタマイズ機能です。

注: ここで説明する機能には、**admin** ロールが必要です。

利用可能なインストレーションイグジット

移動先 システム定義 > インストレーションイグジット. 一部のインストレーションイグジット名 (Login、Logout、ValidatePassword、ExternalAuthentication) は予約されており、変更できません。他のインストレーションイグジットは、デフォルトのインストレーションイグジットのスクリプトを置き換えるカスタムスクリプトで上書きすることができます。

ベースシステムでは、次のインストレーションイグジットを使用できます。

インストレーションイグジット	説明
ログイン	ユーザー名とパスワードのペアを取得し、ユーザーオブジェクトで認証します。
ログアウト	サインアウト時にようこそページに移動します。LogoutRedirect で上書きすることができます。
LogoutRedirect	サインアウト時に指定された URL に移動します。
ExternalAuthentication	ヘッダー、パラメーター、または cookie を使用して認証します。DigestSingleSignOn と PGPSingleSignOn で上書きすることができます。
DigestSingleSignOn	ヘッダー、パラメーター、または cookie を使用して認証し、ダイジェスト暗号を復号化します。
PGPSingleSignOn	ヘッダー、パラメーター、または cookie を使用して認証し、PGP 暗号を復号化します。
ValidatePassword	Helsinki リリース以降、デフォルトでアクティブ化されます。それにより、顧客が独自のパスワード検証を定義できます。また、ValidatePasswordStronger で上書きできます。
ValidatePasswordStronger	パスワードは 8 文字以上で、数字、大文字、および小文字を含める必要があります。

インストレーションイグジット	説明
GetIntegrationSessionTimeout	デフォルトの統合セッションのタイムアウト動作を実装します。

ログインの変更

ログインインストレーションイグジットを次のように変更すると、ユーザーがログインする時に、各ユーザーのセッションタイムアウト値が設定されます。この特定の例では、ユーザー名が *admin* の場合、セッションは 30 秒でタイムアウトに設定されます。

```
gs.include("PrototypeServer");

var Login = Class.create();
Login.prototype = {
 initialize : function() {
 },

      process : function() {
        // the request is passed in as a global
        var userName = request.getParameter("user_name");
        var userPassword = request.getParameter("user_password");

        var authed = GlideUser.authenticate(userName, userPassword);
        if (authed) {
          // ************************************************************
          // customization - if the userName == admin, set the session
          // timeout to be 30 seconds. You can implement your own
          // session timeout algorithm here by checking to see if a user
          // belongs to a certain group or has a certain role.
          // Values of setMaxInactiveInterval exceeding 1440 minutes are
          // treated as one day (1440 minutes).

          if (userName == "admin") {
            request.getSession().setMaxInactiveInterval(30);
          }
          // ************************************************************
          return GlideUser.getUser(userName);
        }

        this.loginFailed();

        return "login.failed";
      },

      loginFailed : function() {
        var message = GlideSysMessage.format("login_invalid");
        var gSession = GlideSession.get();
        gSession.addErrorMessage(message);

        var userName = request.getParameter("user_name");
        EventManager.queue("login.failed", "", userName, "");
      }

}
```

セッションタイムアウトは、IP アドレスに基づいて設定することもできます。

```
gs.include("PrototypeServer");

var Login = Class.create();
Login.prototype = {
 initialize : function() {
 },

      process : function() {
        // the request is passed in as a global
        var userName = request.getParameter("user_name");
        var userPassword = request.getParameter("user_password");

        var authed = GlideUser.authenticate(userName, userPassword);
        if (authed) {

        // ************************************************************
        // customization - if the user is logging in from a particular IP
        // range starting with XXX.XXX you can implement your own
        // session timeout algorithm here by checking the login IP
        //
        // Values of setMaxInactiveInterval exceeding 1440 minutes are
        // treated as one day (1440 minutes).

        var clientIP = gs.getSession().getClientIP().toString();

        // if client IP starts with specified range
        if (clientIP.indexOf('XXX.XXX') == 0) {
          // set to 10 hours
          request.getSession().setMaxInactiveInterval(60 * 60 * 10);
        }
        // ************************************************************

          return GlideUser.getUser(userName);
        }

        this.loginFailed();

        return "login.failed";
      },

      loginFailed : function() {
        var message = GlideSysMessage.format("login_invalid");
        var gSession = GlideSession.get();
        gSession.addErrorMessage(message);

        var userName = request.getParameter("user_name");
        EventManager.queue("login.failed", "", userName, "");
      }

}
```

関連トピック

IP 範囲ベースの認証

Web ベースのアプリケーションを保護する 1 つの方法は、IP アドレスに基づいてアクセスを制限することです。

悪意のある個人に属すると思われる特定のアドレスまたはアドレス範囲へのアクセスをブロックできます。インスタンスにより、IP アドレスでアクセスを制御できます。

注: 適応認証 (AA) 事前認証コンテキストポリシーを使用して、IP ベースの認証と追加機能の制限を強制します。詳細については、「適応認証」を参照してください。

注意事項および制限事項：

- 自分自身をロックアウトすることはできないため、現在のアドレスがロックアウトされるようなルールを追加しようとすると、警告が表示されて挿入が拒否されます。
- 企業イントラネット内にいる場合は、IP ルールの設定に十分注意してください。通常、自分のコンピューターに表示される IP アドレス (10.10.10.25 など) は、実際にインターネット上に表示される IP アドレスとは無関係です。会社は、ユーザーのアドレスを予測可能な送信アドレスにプロキシしたり、NAT を行ったりします。これについては、ネットワークチームに問い合わせる必要があります。
- アクセスルールに基づいてアクセスが制限されているユーザーには、ブラウザーで 403 エラーが表示されます。
- 制限付きユーザーは、トランザクション、セマフォ、またはサーバーリソースカウントを使用しません。
- たとえば、データセンターへの VPN を実行している場合、この機能が既存のアクセス制御ルールに優先されることも上書きされることもありません。これは、PIX で設定されている可能性があるアクセス制御に加えて満たす必要がある追加のチェックです。
- 許可ルールは常に拒否ルールより優先されます。そのため、アドレスが (1 つのルールで) 許可され、(2 番目のルールで) 拒否される場合、実際には許可されます。
- 現在、アスタリスクと CIDR ブロックはサポートされていません。
- 転送プロキシアドレスについては、チェーン内の各アドレスに許可ルールが適用され、一致する許可ルールがない場合はチェーン内の各アドレスに拒否ルールが適用されます。
- IP 範囲ベースの認証は、更新セットの転送に影響を与える可能性があります。ソースインスタンスで IP アドレスアクセス制御が有効になっている場合は、インスタンスをサポートするすべてのアプリケーションノードの IP アドレスを例外として追加します。

  注: インスタンスの IP 情報を見つけるには、 ServiceNow - NOW サポート 、 自分の **IP** 情報 サービスカタログアイテムを検索します。

注: 特定の IP 範囲へのインスタンスアクセスを制限する *com.snc.ipauthenticator* プロパティと *glide.ip.authenticate.strict* プロパティの詳細については、「インスタンスセキュリティ強化設定」の以下のトピックを参照してください。

特定の IP 範囲プラグインへのアクセスを制限する (Security Center 1.3 で更新)

IP アドレスアクセス制御

送信トラフィック、受信トラフィック、または双方向トラフィックに IP アクセス制御を適用します。一致する拒否ルールが存在し、一致する許可ルールが存在しない場合にのみ、IP アドレスがブロックされます。デフォルトでは、インスタンスへのアクセスに制限はありません。

始める前に

注: 適応認証 (AA) 事前認証コンテキストポリシーを使用して、IP ベースの認証と追加機能の制限を強制します。詳細については、「適応認証」を参照してください。

必要なロール：admin

警告: IP アドレス制御は外部 IP に対してのみ構成され、ServiceNow からの内部 IP アドレスはブロックされません。

手順

1. 移動先 すべて > システムセキュリティ > **IP** アドレスアクセス制御 を使用して、IP アクセス制御のリストを表示します。
   IP 範囲ベースの認証 [com.snc.ipauthenticator] プラグインを有効にする必要がある場合があります。
2. フォームを完了します。

   注: インスタンスの IP 情報を見つけるには、 ServiceNow - NOW サポート 、 自分の **IP** 情報 サービスカタログアイテムを検索します。

フィールド	説明
タイプ	含めるアクセス制御ルールのタイプ。 ○ 許可：この範囲内のすべての IP アドレスがこのインスタンスとやり取りができます。 ○ 拒否：この範囲内の IP アドレスは、許可ルールにリストされていない限り、このインスタンスとやり取りができません。また、拒否ルールを追加するときに、自分のパブリック IP アドレスを拒否できないか、インスタンスが拒否ルールを更新しません。 注: メンテナンス、アップグレード、および カスタマーサービス & サポート をサポートするため、一部の ServiceNow の内部 IP は拒否ルールでブロックできません。
方向	IP アクセス制御ルールの方向。 ○ 受信：受信トランザクションを許可または拒否します。これらは、インスタンスの外部から開始されたトランザクションです。 ○ 送信：送信トランザクションを許可または拒否します。これらは、インスタンスの内部から開始されたトランザクションです。 ○ 双方向：受信トランザクションと送信トランザクションの両方を許可または拒否します。
有効	選択すると、フォームがアクティブ化されます。
説明	アクセス制御の説明。
範囲開始	許可または拒否する IP アドレスの開始範囲。 注: これらのルールは、更新セットの転送にも影響します。IP アドレスアクセス制御によって更新セットが失敗しないようにするには、ターゲットインスタンスを例外として追加します。
範囲終了	許可または拒否する IP アドレスの終了範囲。

自動翻訳

フィールド	説明
	注: 特定の VPN アドレスのみにアクセスを制限するには、[拒否] フィールドに **0.0.0.0** ～ **255.255.255.255** の範囲を入力し、特定の許可された VPN 範囲のみを入力します。

3. [送信] をクリックします。

拒否された **IP** アドレスを検索する

インスタンスのノードログファイルで、拒否された IP アドレスを検索します。

始める前に
必要なロール：admin。

このタスクについて
ブロックされた IP アドレスのログエントリは、2015-10-21 18:37:43 (175) http-30 警告 *** 警告 *** セキュリティ制限：アクセスが制限されています (xx.xx.xxx.xxx は許可されていません) のように表示されます。

注: 拒否された IP アドレスは、表示可能なインスタンスのノードログファイルであり、システムログからは表示できません。

手順

1. 移動先 すべて > システムログ > ユーティリティ > ノードログファイルブラウザ。.
2. 期間やメッセージなどの基準でログを参照します。
3. 探しているログがわかっている場合は、 システムログ > ユーティリティ > ノードログファイルのダウンロード.

ライトウェイトディレクトリアクセスプロトコル統合

LDAP 統合により、インスタンスで既存の LDAP サーバーをユーザーデータのプライマリソースとして使用できます。

アドミニストレーターは、ライトウェイトディレクトリアクセスプロトコル (LDAP) ディレクトリと統合して、ユーザーログインプロセスを簡素化し、ユーザーの作成やユーザーへのロールの割り当てなどの管理タスクを自動化します。LDAP 統合により、システムは既存の LDAP サーバーをユーザーデータのプライマリソースとして使用できます。通常、LDAP 統合もシングルサインオン実装の一部です。

統合では、LDAP サービスアカウントの認証情報を使用して、LDAP サーバーからユーザーの識別名 (DN) を取得します。ユーザーの DN 値を指定すると、統合はユーザーの DN とパスワードを使用して LDAP に再バインドします。ユーザーが入力するパスワードは、HTTPS セッションに完全に含まれています。統合では LDAP パスワードは保存されません。

統合は、LDAP ディレクトリに書き込むことのない読み取り専用接続を使用します。統合は情報をクエリするだけで、それに応じて内部データベースを更新します。

注: 統合の設定の詳細については、「LDAP 統合のセットアップ」を参照してください。

注: インスタンスが LDAP 統合を使用していて、Active Directory 設定でユーザーがログイン時にパスワードをリセットする必要がある場合、ユーザーはインスタンスにログインできません。インスタンスは、ユーザーの Active Directory のパスワードを変更できません。

LDAP 統合の機能

LDAP 統合機能は次のとおりです。

スケジュールされた LDAP リフレッシュ

LDAP サーバーのスケジュール設定済みスキャンは通常、夜間に 1 回実行されます。該当するすべてのユーザーレコードの属性を照会し、それらをサーバー上のアカウントと比較します。差異がある場合は、変更された属性を使用してユーザーレコードを変更します。リフレッシュ中に LDAP サーバーにかかる負荷は、照会されるレコードの数と比較される属性の数によって異なります。ピーク時間外にリフレッシュをスケジュールすることをお勧めします。大規模なリフレッシュ操作は、レポートの実行などの他のスケジュールされた操作に影響を与える可能性があるため、競合を最小限に抑えるように計画する必要があります。

LDAP リスナー

LDAP リスナーは永続クエリ (または永続検索) のバージョンです。LDAP サーバーに加えられた変更に対して継続的なクエリを発行し、常に応答をリッスンします。サーバーが永続検索をサポートしていると仮定すると、該当する LDAP アカウントに加えられた変更はすべて LDAP リスナーに返され、約 10 秒以内にインスタンスに送信されます。これは非常に便利なツールであり、次にスケジュールされたリフレッシュを待たずに、ユーザーのアカウントの詳細をほぼリアルタイムでコピーできます。

オンデマンド LDAP ログイン

LDAP 統合が確立されると、インスタンスのアカウントがまだない場合でも、新しいユーザーがシステムにログインできるようになります。新しいユーザーがインスタンスにログインしようとすると、統合はこのユーザーがインスタンスにアカウントを持っているかどうかを確認します。統合で既存のユーザーアカウントが見つからない場合は、入力されたユーザー名が LDAP サーバーに自動的に照会されます。一致する LDAP アカウントが見つかった場合、統合はユーザーが入力したパスワードを使用して認証を試行します。パスワードが有効な場合、インスタンスはユーザーのアカウントを作成し、該当するすべての LDAP 情報をアカウントに指定して、ユーザーをインスタンスにログインさせます。

オンデマンドログインでは、LDAP ユーザーインポート変換マップを使用します。変換マップの要件の詳細については、「LDAP 変換マップ」を参照してください。

LDAP データの入力

注: この統合で説明されている機能は、デフォルトでは利用できません。この統合では、経験豊富なアドミニストレーターまたは ServiceNow プロフェッショナルサービスコンサルタントによって実行される展開後のカスタマイズが含まれます。

LDAP サーバーとの統合により、既存の LDAP データベースのユーザーレコードをインスタンスのデータベースに迅速かつ簡単に入力できます。データの不整合を防ぐために、着信 LDAP レコードを作成、無視、またはスキップすることができます。

LDAP 属性を指定して統合がインポートするデータを制限し、インスタンスに公開するデータのみをインポートすることもできます。通常、指定する LDAP 属性は統合変換マップの一部になります。LDAP 属性を指定しない場合、統合は LDAP サーバーから利用可能なすべてのオブジェクト属性をインポートします。インスタンスは、インポートされた LDAP データを一時インポートセットテーブルに格納するため、インポートする属性が多いほど、インポート時間が長くなります。詳細については、「LDAP 属性の指定」を参照してください。

LDAP 認証

LDAP 認証を使用するには、LDAP 認証情報を使用してアクセスします。

ユーザーがログインページでネットワーク認証情報を入力すると、次のようになります。

1. インスタンスは、認証情報を LDAP サーバーに渡してインスタンスを検索します。
2. RDN を使用して、ユーザーの DN 文字列を検証します。table=sys_user の LDAP OU 構成の少なくとも 1 つに RDN が構成されている場合にのみ検証されます。
3. LDAP サーバーは、システムがアクセスを許可するかどうかを判断するために使用する、許可または許可されていないメッセージで応答します。

LDAP サーバーに対して認証することで、ユーザーはネットワークドメイン上の他の内部リソースに使用するものと同じ認証情報を使用してプラットフォームにアクセスします。また、既存のパスワードとセキュリティポリシーを再利用することもできます。たとえば、LDAP サーバーには既にアカウントロックアウトとパスワードの有効期限ポリシーが設定されている場合があります。

LDAP を有効にすると、次のフィールドでユーザーレコードが更新されます。

LDAP ユーザーレコードの更新

フィールド	説明
ソース	LDAP を使用してユーザーを検証するかどうかを示します。ソースが ldap で始まる場合、ユーザーは LDAP を介して検証されます。ソースが ldap で始まらない場合、ユーザーレコードのパスワードは、ログイン時にユーザーを検証するために使用されます。
LDAP サーバー	複数の LDAP サーバーがある場合に、ユーザーを認証する LDAP サーバーを識別します。

注: システムは、MID サーバーを介した LDAP パスワード認証をサポートしていません。インスタンスは、パスワード認証をサポートするために LDAP サーバーに直接接続できる必要があります。

LDAP 統合を理解する

LDAP 統合により、インスタンスで既存の LDAP サーバーをユーザーデータのプライマリソースとして使用できます。

LDAP 統合の前提条件

- ディレクトリサービスサーバーは LDAP v3 に準拠している必要がある
- ファイアウォール経由の受信ネットワークアクセスを (LDAP サーバーに対して) 許可する必要がある
- LDAP サーバーの外部 IP または名前
- 読み取り専用アクセス権を持つユーザー認証情報
- LDAPS の場合は PKI 証明書

LDAP 統合のタイミング

LDAP 統合は通常、インスタンスの本番稼働前に行われますが、いつでも統合できます。

LDAP サーバーのデータ整合性

一部のユーザーは、第三者 (この場合はインスタンス) が LDAP サーバーを変更 (書き込み) することを懸念しています。LDAP 統合では、インスタンスは内部 LDAP ディレクトリに書き込みません。インスタンスは情報を照会し、それに応じてデータベースを更新します。

インスタンスによって内部 LDAP サーバーが変更されることはありません。サービスアカウントは読み取り専用です。

LDAP サーバーへのほとんどの変更 (追加を含む) は、完全な LDAP 統合のコンポーネントの数に応じて、数秒でインスタンスで利用可能になります。

LDAP レコードの同期を維持するには、LDAP サーバーの定期的なスキャンをスケジュールして変更を取得します。

インスタンスは部門レコードを同期しません。ユーザーとグループのメンバーシップは、LDAP リスナーメカニズムと日次の完全な LDAP ブラウザーによって最新の状態に維持されますが、LDAP から消えたこれらのエントリは削除されません。

エントリを削除すると、履歴全体も削除され、そのエントリへの参照がクリアされるか削除されます。構成アイテム (CI)、 SLA 契約、ソフトウェアライセンス、発注書 (PO)、およびサービスカタログエントリにはすべて部門への参照があり、部門が削除されると、それらの参照はクリアされます。ユーザーへの参照が多数あるため、ユーザーを削除すると、そのユーザーが行ったすべての履歴が失われます。現在、削除するかどうかはお客様が決定します。

セキュリティ

接続は、ファイアウォール上の特定のポートを介して、固定 IP アドレスを使用して単一のマシンから行われます。認証は、選択した読み取り専用の LDAP アカウントで行われます。標準 LDAP を使用するか、ディレクトリにインストールされている SSL 証明書の公開側をロードできます。その場合は、LDAPS を使用できます。別のセキュリティレイヤーを追加するために、ポイントツーポイント IPSEC VPN トンネルのオプションも提供しています。詳細と価格については、アカウントマネージャーにお問い合わせください。

セキュア LDAP 接続

接続	説明
MID サーバー	LDAP サーバーを外部ネットワークトラフィックから保護するには、ローカルネットワークに MID サーバーをインストールし、安全なチャネルを介して MID サーバーと通信するようにシステムを設定します。
LDAPS	暗号化された LDAPS 接続を確立するには、LDAP サーバーの SSL 証明書の公開側をロードします。統合では、証明書を使用して LDAP サーバーとインスタンス間のすべての通信を暗号化します。
VPN	暗号化されたポイントツーポイント IPSEC VPN トンネルを使用して LDAP サーバーを保護するには、詳細と価格についてアカウントマネージャーにお問い合わせください。

考慮すべきもう 1 つのセキュリティ面は、LDAP 統合で共有されるデータです。インスタンスに公開されるデータを制限するには、変換マップで属性を指定します。詳細については、「LDAP 変換マップ」を参照してください。

LDAP データのインスタンスへのインポート

必要なデータのみをインポートするように属性を定義することをお勧めします。定義された属性がインスタンスユーザーデータベースにマッピングされます。

どの属性が必要かは、プロジェクトのスコープとビジネス要件によって決まるため、回答できません。

サポートされている LDAP サーバーのタイプ

インスタンスは、Microsoft Active Directory、Novell、Domino (Louis Notes)、および Open LDAP と正常に統合されています。LDAP サーバーとのインターフェイスには JNDI を使用します。LDAP サーバーが LDAP v3 に準拠している限り、統合は成功します。

LDAP Single-sign-on

LDAP インポートで提供されるデータ入力機能に加えて、アプリケーションでサポートされている外部認証機能を使用して、ユーザーが毎回サインオンする必要がないようにすることができます。

複数の LDAP ドメイン

複数のドメインを処理する場合は、ドメインごとに個別の LDAP サーバーレコードを作成することをお勧めします。各 LDAP サーバーレコードは、そのドメインのドメインコントローラーを指す必要があります。これは、ローカルネットワークが各ドメインコントローラーへの接続を許可する必要があることを意味します。

複数のネットワークドメインに展開した後、アプリケーションユーザー名の一意の LDAP 属性を特定し、結合値をインポートすることが重要です。Active Directory の一般的な一意の結合属性は objectSid です。一意のユーザー名は、LDAP データ設計によって異なる場合があります。一般的な属性は email または userPrincipalName です。

クエリ制限の処理

デフォルトでは、Active Directory 2000/2003 には、過度の負荷とサービス拒否攻撃を防ぐために、1000 オブジェクトの LDAP クエリ制限 (maxPageSize) があります。この制限に対処する方法は 2 つあります。

デフォルトの方法では、一度に 1000 未満のオブジェクトを返すようにクエリを分割します。たとえば、「a」で始まるオブジェクトのみをクエリしてから、「b」オブジェクトをクエリします。大規模な環境でより効率的に行う方法は、ページングを有効にすることです。ページングは、すべての Microsoft Active Directory サーバーでデフォルトでサポートされています。結果は自動的に複数の結果セットに分割されるため、クエリを複数の要求に分割する必要はありません。

LDAP クエリタイプ。

LDAP パスワードが指定されている場合は、「簡易バインド」が実行されます。LDAP パスワードが指定されていない場合は「none」が使用されます。この場合、LDAP サーバーは匿名ログインを許可する必要があります。

LDAP 認証

提供された LDAP のサービスアカウント認証情報を使用して、LDAP サーバーからユーザー DN を取得します。ユーザーの DN 値を指定して、ユーザーの DN とパスワードを指定して LDAP に再バインドします。

パスワードストレージ

ユーザーが入力するパスワードはすべて HTTPS セッションに含まれます。そのパスワードはどこにも保存されません。

LDAP 認証の設定

ユーザーレコードの次のフィールドは LDAP に関連しています。

- ソース：[ソース] フィールドは、ユーザーが LDAP を使用して検証されているかどうかを示します。[ソース] フィールドが「ldap」で始まる場合、ユーザーは LDAP を介して検証されます。[ソース] フィールドが「ldap」で始まっていない場合、ユーザーレコードのパスワードを使用してログイン時にユーザーが検証されます。
- **LDAP** サーバー：インスタンスは複数の LDAP サーバーをサポートしているため、[LDAP サーバー] フィールドによってユーザーの認証に使用するサーバーが決まります。

LDAP 統合要件

PKI 証明書、LDAP 準拠のディレクトリサービスサーバーを含む、LDAP 統合の要件を確認します。

LDAP 統合には以下が必要です。

- LDAP v3 準拠のディレクトリサービスサーバー
  - ファイアウォール経由の (LDAP サーバーへの) 受信ネットワークアクセスを許可
  - (オプション) 匿名ログインを承認
  - (オプション) 大規模な LDAP クエリのページングをサポート
- LDAP サーバーの外部 IP アドレスまたは完全修飾ドメイン名。MID サーバーを使用することもできます。
- 選択した読み取り専用の LDAP アカウント
- 複数のドメインの場合、各ドメインコントローラのネットワークアクセス
- LDAPS の場合は PKI 証明書
- LDAP リスナーの場合は、永続クエリ (ADNotify) をサポートする Microsoft Active Directory サーバー

サポート対象 LDAP サーバー

JNDI を使用して LDAP サーバーと連携することで、インスタンスは次のサーバーと正常に統合されています。

- Microsoft Active Directory
- Novell
- Domino (Locus Notes)
- LDAP を開く

LDAP クエリの制限

デフォルトでは、Active Directory 2000/2003 には、過度の負荷とサービス拒否攻撃を防ぐために、1000 オブジェクトの LDAP クエリ制限 (maxPageSize) があります。システムには、この制限に対処する 2 つの方法があります。

- デフォルトの方法では、一度に 1000 未満のオブジェクトを返すようにクエリを分割します。たとえば、「a」で始まるオブジェクトのみをクエリしてから、「b」オブジェクトをクエリします。
- 大規模な環境でより効率的な方法は、すべての Microsoft Active Directory サーバーでデフォルトでサポートされているページングを有効にすることです。ページングでは、結果を自動的に複数の結果セットに分割するため、統合によってクエリを複数の要求に分割する必要はありません。

LDAP 統合のセットアップ

アドミニストレーターは、LDAP 統合を有効にして、会社の LDAP ディレクトリからのユーザーのサインオンを許可できます。

LDAP は通常、次のいずれかの通信チャネルを使用します。

LDAP 通信チャネル

接続	説明	LDAP インポートのサポートは？	LDAP 認証のサポートは？
MID サーバー接続	デフォルトでは、ポート 80 で HTTP を介して通信します。この通信チャネルには証明書は必要ありません。MID サーバーとインスタンス間の接続は HTTPS (ポート 443) です。MID サーバーを使用して LDAP 経由でデータをインポートできますが、LDAP 認証に MID サーバーを使用することはできません。「LDAP サーバーの定義」に進みます。	あり	なし
標準 LDAP 統合	デフォルトでは、ポート 389 で TCP を介して通信します。この通信チャネルには証明書は必要ありません。「LDAP サーバーの定義」に進みます。	あり	あり
SSL 暗号化 LDAP 統合 (LDAPS)	デフォルトではポート 636 で通信します。この通信チャネルには証明書が必要です。LDAP X.509 SSL 証明書をインストールする に進み、証明書を取得してアップロードしてください。	あり	あり
VPN 接続	IPSEC トンネルを介して通信します。ローカルネットワークで IPSEC トンネルを購入または作成します。「LDAP サーバーの定義」に進みます。	あり	あり

MID サーバーを使用する場合、MID サーバーはインスタンスに接続し、続いて MID サーバーは LDAP サーバーにも接続します。どちらの場合も、MID サーバーが接続を開始します。

1. まず、MID サーバーはポート 389 で LDAP を介して LDAP サーバーに接続します。
2. 次に、MID サーバーはポート 443 でインスタンスへの HTTPS 暗号化接続を開始し、データをインスタンスにプッシュします。

VPN、MID サーバー、および LDAP の詳細については、コミュニティの「VPN Part II は必要ありません (You Don't Need A VPN Part II)」を参照してください。

LDAP X.509 SSL 証明書をインストールする

LDAP 統合用の X.509 証明書をインストールできます。

始める前に
必要なロール：admin

手順

1. LDAP サーバーで SSL 証明書を購入または生成します。
2. 移動先 **LDAP** > 証明書 をクリックし、[新規] をクリックします。
3. フォームフィールドに記入します。

フィールド	説明
名前	証明書名。
有効期限通知	[期限切れ時に通知] フィールドで選択したユーザーに通知を送信するには、このオプションを選択します。デフォルトでは、これは有効になっています。
期限切れ時に通知	証明書の有効期限に関する通知を有効にするユーザーを選択します。ユーザーが選択されていない場合、ログインしているユーザーは、最後にログインした 2 人のアドミニストレーターロールとともに追加されます。
有効期限切れ前に警告	インスタンスが通知を送信する期限切れまでの日数。20 以上の値を入力してください。Istanbul 以降のリリースにアップグレードしたインスタンスでは、より大きい値を指定しない限り、この値は 20 に設定されます。
アクティブ	この証明書がアクティブであることを示すチェックボックス。
フォーマット	証明書の形式。
タイプ	証明書コンテナー。インスタンスは、トラストストア、Java キーストア、および PKCS#12 キーストアからの証明書を認識します。
有効開始日	インスタンスは、証明書の有効開始日を自動的にこのフィールドに追加します。証明書を X.509 証明書レコードに添付して、このフィールドに入力します。
有効期限	インスタンスは、証明書の有効期限をこのフィールドに自動的に追加します。証明書を X.509 証明書レコードに添付して、このフィールドに入力します。
数日中に期限切れ	有効期限までの計算された日数。
簡単な説明	証明書の説明。
発行者	インスタンスは、証明書発行者をこのフィールドに自動的に追加します。証明書を X.509 証明書レコードに添付して、このフィールドに入力します。
件名	インスタンスは、このフィールドの対象となる証明書を自動的に追加します。証明書を X.509 証明書レコードに添付して、このフィールドに入力します。
PEM 証明書	X509 証明書の値を入力します。

注: 統合は現在、インスタンスと IdP 間の通信で証明書に署名しません。

4. [保存] をクリックします。

次のタスク
[ストア/証明書を検証] をクリックして、トラストストアと証明書をテストします。

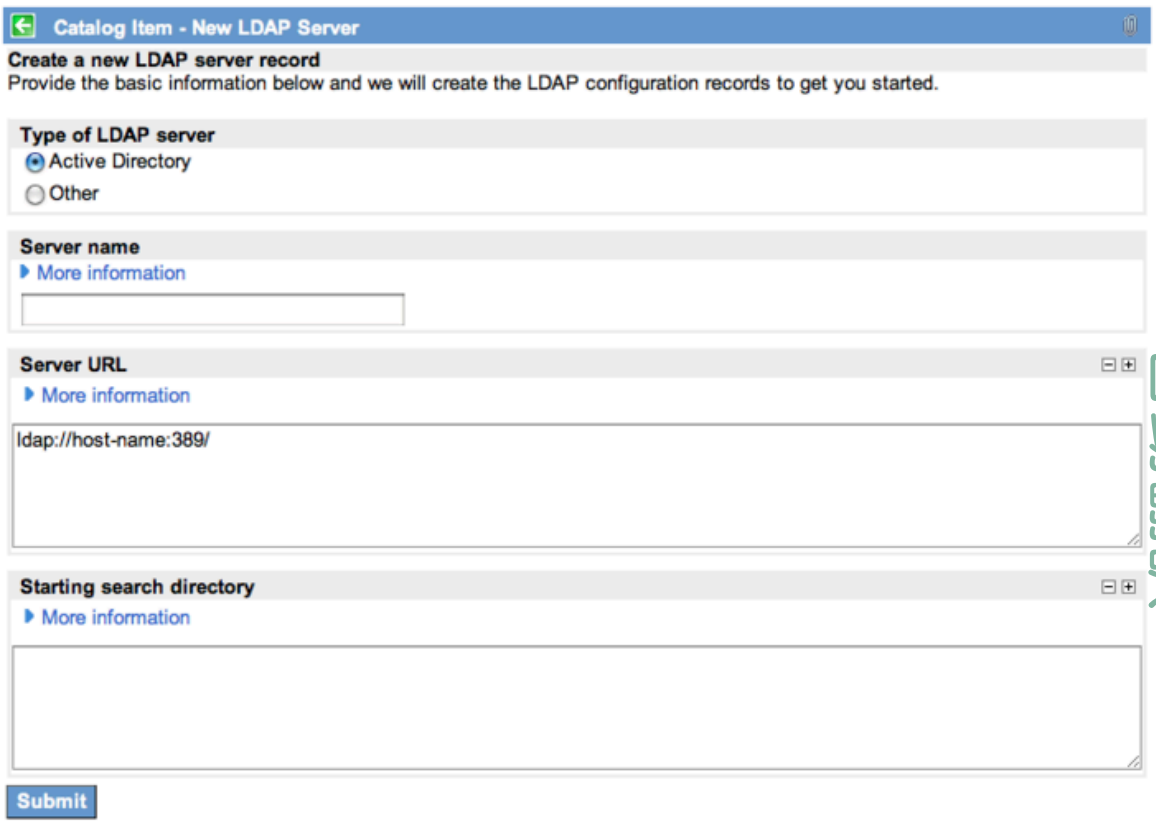
LDAP サーバーを定義する

インスタンスに新しい LDAP サーバーレコードを作成します。

始める前に
必要なロール：admin。

手順

1. 移動先 すべて > システム **LDAP** > 新規サーバーを作成.
2. フォームフィールドに記入します。



[サーバー **URL**] フィールドには、すべてのサーバーの有効な URL がスペースで区切られて表示されます。サーバーは最初に運用ステータス順に並べ替えられます。[稼働中] のサーバーが最初に表示され、次に指定した [順序] の値順に並べ替えられます。リストの最初のサーバーはプライマリ LDAP サーバーです。その他は冗長サーバーです。

> **注:** 実際の運用ステータスの変更と表示の間にはわずかな遅延があります。

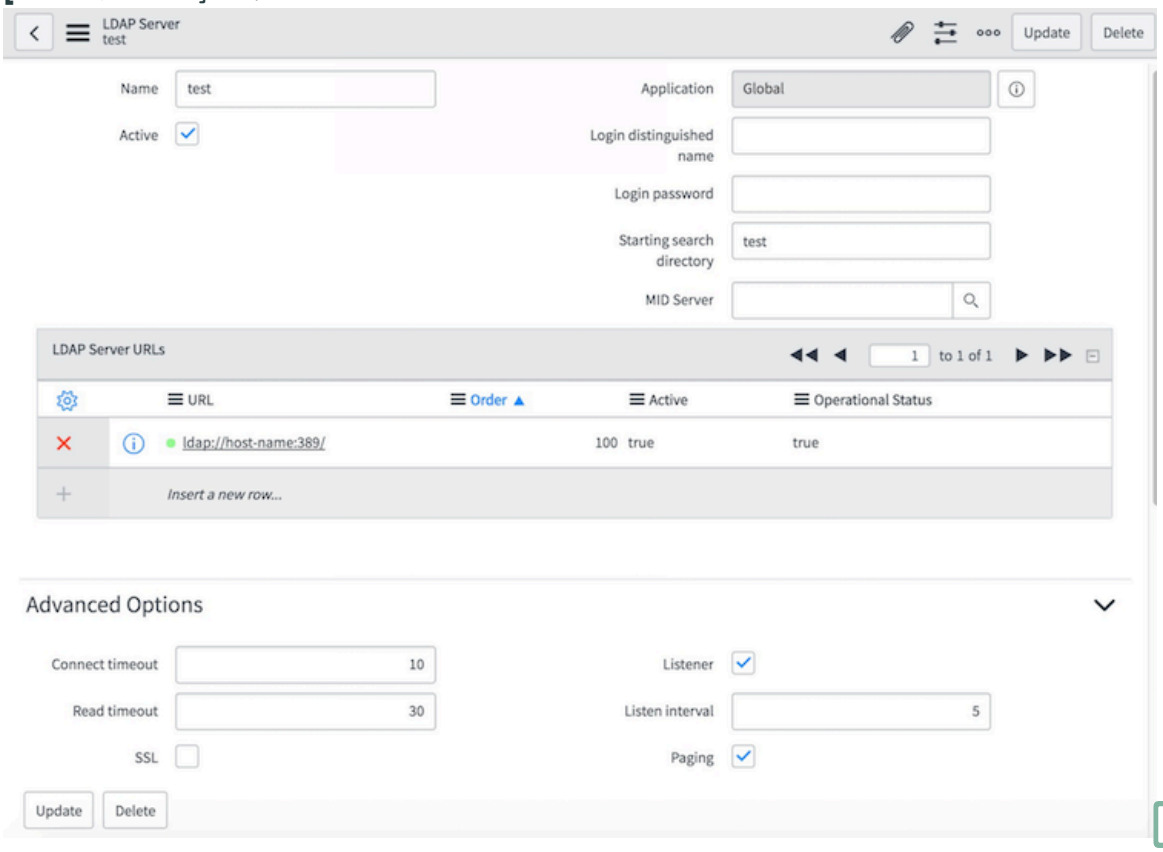
または、既存の LDAP サーバーレコードに移動し、[LDAP サーバー URL] 埋め込みリストに行を挿入することで、冗長な LDAP サーバーを追加することもできます。

3. [**Submit** (送信)] を選択します。

> **注:** に移動して、既存の LDAP サーバーレコードを変更することもできます。 システム **LDAP** > **LDAP** サーバー 必要な変更を行います。

4. 必要に応じてフィールドを変更します。

[LDAP サーバー] フォーム



フィールド	説明
名前	サーバーの名前を入力します。
アクティブ	サーバーがアクティブな場合は、このチェックボックスをオンにします。
LDAP サーバー URL	プライマリ LDAP サーバーとバックアップ LDAP サーバーの URL を入力します。サーバーは最初に運用ステータス順に並べ替えられます。[稼働中] のサーバーが最初に表示され、次に指定した [順序] の値順に並べ替えられます。リストの最初のサーバーはプライマリ LDAP サーバーです。その他は冗長サーバーです。
サーバー URL	サーバーの URL を入力します。必要に応じて、このフィールドを追加するフォームを設定します。これは計算された読み取り専用フィールドで、**[LDAP サーバー URL]** フィールドにも表示される LDAP サーバーのリストをスペースで区切って、運用ステータスと URL の順序値で表示します。
ログイン識別名	LDAP 接続を認証するユーザーの識別名 (DN) を入力します。 LDAP ディレクトリサーバーにアクセスするには、ユーザー名を完全な識別名形式 (servicenow@service-now.com) にする必要があります。
ログインパスワード	サーバーのパスワードを入力します。

フィールド	説明
検索開始ディレクトリ	デフォルトの検索ディレクトリの相対識別名 (RDN) を入力します。この LDAP サーバーに対するすべてのクエリは、この RDN から開始されます。
MID サーバー	LDAP サーバーへの接続に使用する MID サーバーを選択します。MID サーバーを使用して LDAP 接続を確立すると、LDAP サーバーを外部ネットワークトラフィックに公開する必要がなくなります。また、LDAP サーバーと ServiceNow データセンター間に VPN トンネルを確立する必要がなくなります。 **注:** ○ LDAP サーバー設定レコードを読み取れるようにするには、MID サーバーユーザーに user_admin ロールが必要です。 ○ 以下は MID サーバーでは利用できません。 ▪ LDAP 認証 ▪ SSL 接続
接続タイムアウト	MID サーバーが設定されている場合、この設定に関係なく、接続は 10 秒後にタイムアウトします。この設定はハードコードされており、変更することはできません。
読み込みタイムアウト	統合が LDAP データを読み取るために必要な秒数を指定します。接続が読み込みタイムアウトを超えると、統合は LDAP データの読み込みを停止します。SSL 接続を有効にすると、*com.glide.ssl.read.timeout*システムプロパティを使用して読み取りタイムアウト値を設定することもできます。このフィールドとシステムプロパティの両方にタイムアウト値を入力すると、最も低いタイムアウト値が優先されます。
SSL	LDAP サーバーに SSL 暗号化接続を要求するには、このチェックボックスをオンにします。MID サーバーを選択した場合、このフィールドは使用できません。 LDAPS 統合を使用していて、デフォルトの SSL ポートが 636 である場合、それ以上の設定は必要ありません。SSL は自動的に有効になります。LDAPS 統合で別の SSL ポートを使用する場合は、代替 SSL 接続プロパティを定義します。 **注:** アプリケーションサーバーが LDAP サーバーにアクセスできるように、ネットワークアドミニストレーターがローカルファイアウォールを設定していることを確認してください。LDAP サーバーが内部ネットワーク内にある場合、ファイアウォールはアプリケーションサーバーの IP アドレスを正しいポートのファイアウォールで転送 (NAT) します。
リスナー	このチェックボックスをオンにすると、統合により、永続的な検索要求のコントロールをサポートする Microsoft Active Directory サーバーまたは LDAP サーバーを定期的にポーリングできます。さらに、MID サーバーを選択した場合は、その MID サーバーでリスナー機能を使用できます。詳細については、「LDAP リスナーを有効にしてシステムプロパティを設定する」を参照してください。

フィールド	説明
リッスン間隔 (タイムアウト値)	統合がすべての接続で LDAP データをリッスンするリスナーのタイムアウト値を分単位で指定します。接続がリッスン間隔を超えると、統合は LDAP データのリッスンを停止します。
ページング	このチェックボックスをオンにすると、LDAP サーバーは LDAP 属性データを複数のクエリを送信するのではなく、複数の結果セットに分割します。

注: LDAP パスワードを指定すると、統合によって簡易バインド操作が実行されます。LDAP パスワードを指定しない場合、LDAP サーバーは匿名ログインを許可する必要があります。そうしないと、統合は LDAP サーバーにバインドできません。

結果
LDAP サーバーレコードがアクティブに設定されている場合、システムはすべての接続を自動的にテストして検証します。
検証には以下が含まれます。

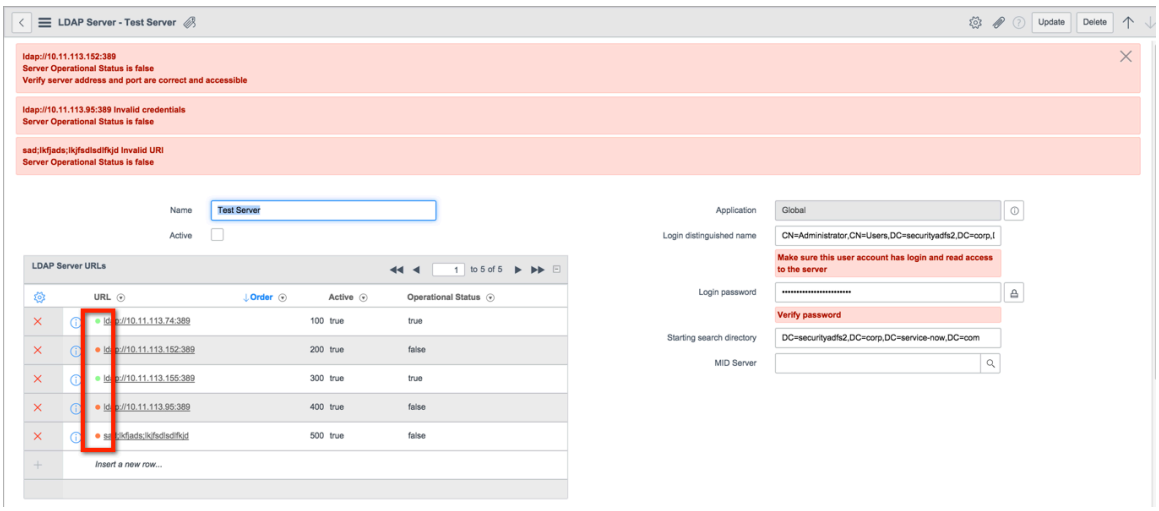
- 指定された URL とポートで LDAP サーバーにアクセスできる
- LDAP サーバーの URL が適切に形式化されている
- ログイン認証情報が有効である

Fuji リリース以降、各サーバーの URL の横に色付きのドットが表示されます。

LDAP サーバー接続アイコン

色	説明
緑	サーバーはアクティブで運用中です。
グレイ	サーバーはアクティブでも運用中でもありません。
赤	サーバーはアクティブですが、運用されていません。

LDAP サーバー接続ステータス



自動翻訳

LDAP リスナーを有効にしてシステムプロパティを設定する

リスナーの有効化はオプションです。有効にすると、LDAP サーバーで更新が行われた直後に、LDAP レコードを処理するようにリスナーがシステムに通知します。

始める前に
必要なロール：admin。

このタスクについて
リスナーは、LDAP サーバー上の変更を定期的に検索する専用のプロセスです。

リスナーは、永続クエリ (ADNotify) をサポートする Microsoft Active Directory サーバー、または永続的な検索要求コントロール (OID 2.16.840.1.113730.3.4.3) をサポートする LDAP サーバーに展開できます。

LDAP サーバーが永続的な検索をサポートしている場合、LDAP リスナーは、該当する LDAP アカウントに加えられたユーザーとグループの変更を認識し、約 10 秒以内にインスタンスに転送します。これにより、インスタンスは、次にスケジュールされたリフレッシュを待たずに、ユーザーのアカウントの詳細のコピーをほぼリアルタイムで保持できます。LDAP リスナーは、ユーザー [sys_users] テーブルとグループ [sys_user_group] テーブルにマップされるオブジェクトのみを同期できます。

注: ユーザーがリスナーを介して追加されても、そのユーザーが OU フィルターで定義されている要件を満たしていない場合、インスタンスは LDAP サーバーのレコードを無視します。条件を満たす場合、ユーザーはインスタンスに追加されます。

リスナーを有効にするには：

手順

1. 移動先 すべて > システム **LDAP** > **LDAP** サーバー.
2. 設定する LDAP サーバーを選択します。
3. [リスナー] チェックボックスをオンにします。
4. [更新] をクリックします。

   注:

   LDAP OU フィルターに一致するユーザーレコードのみがインポートされます。フィルター要件を満たさない受信ユーザーレコードは無効としてフラグが付けられ、インポートによって無視されます。アドミニストレーターは、詳細な LDAP ログ記録を有効にして、受信レコードが LDAP OU フィルターに一致していないかどうかを判断できます。

5. オプション: システムのプロパティ [sys_properties] テーブルに移動し、LDAP リスナーのシステムプロパティを設定します。

LDAP リスナーのプロパティ

プロパティ	説明
glide.ldap.listener.use_background_transaction	true の場合、LDAP リスナーはバックグラウンドトランして開始されます。LDAP リスナーをバックグラウンドンとして実行すると、最大期間 (デフォルトでは 5 分) にクォータルール LDAP リスナーの開始/停止トランザクザクションをキャンセルできます。この動作により、LD無制限に待機することを防止します。

自動翻訳

プロパティ	説明
	注: このプロパティは、MID サーバーを使用しない LDAP 接続にのみ適用されます。*glide.ldap.listener.mid.use_backgrou* を使用して、MID サーバーを経由する LDAP 接続のロールします。 ○ タイプ：true | false ○ デフォルト値：false ○ 場所：システムプロパティ [sys_properties] テーブル
glide.ldap.listener.mid.use_background_transaction	true の場合、LDAP リスナーはバックグラウンドトランして開始されます。LDAP リスナーをバックグラウンドトンとして実行すると、最大期間 (デフォルトでは 5 分) にクォータルール LDAP リスナーの開始/停止 MID トラントランザクションをキャンセルできます。この動作により ナーが無制限に待機することを防止します。 注: このプロパティは、MID サーバーを使用する LDAP 接続にのみ適用されます。*glide.ldap.listener.use_background_t* を使用して、MID サーバーを経由しない L DAP 接ントロールします。 ○ タイプ：true | false ○ デフォルト値：false ○ 場所：システムプロパティ [sys_properties] テーブル
glide.ldap.listener.mid.one_listener	true の場合、MID サーバーを介して LDAP リスナーを開するための ECC キューメッセージが 1 つだけ作成されま 場合、複数の ECC キューメッセージが作成され、LDAP 始または停止する複数のスレッドが作成されます。 ○ タイプ：true | false ○ デフォルト値：true ○ 場所：システムプロパティ [sys_properties] テーブル

LDAP 属性の指定

LDAP サーバーの [属性] フィールドを使用して、LDAP サーバーのクエリに含まれる属性を指定します。これにより、パフォーマンスとセキュリティを強化できます。

始める前に
必要なロール：admin

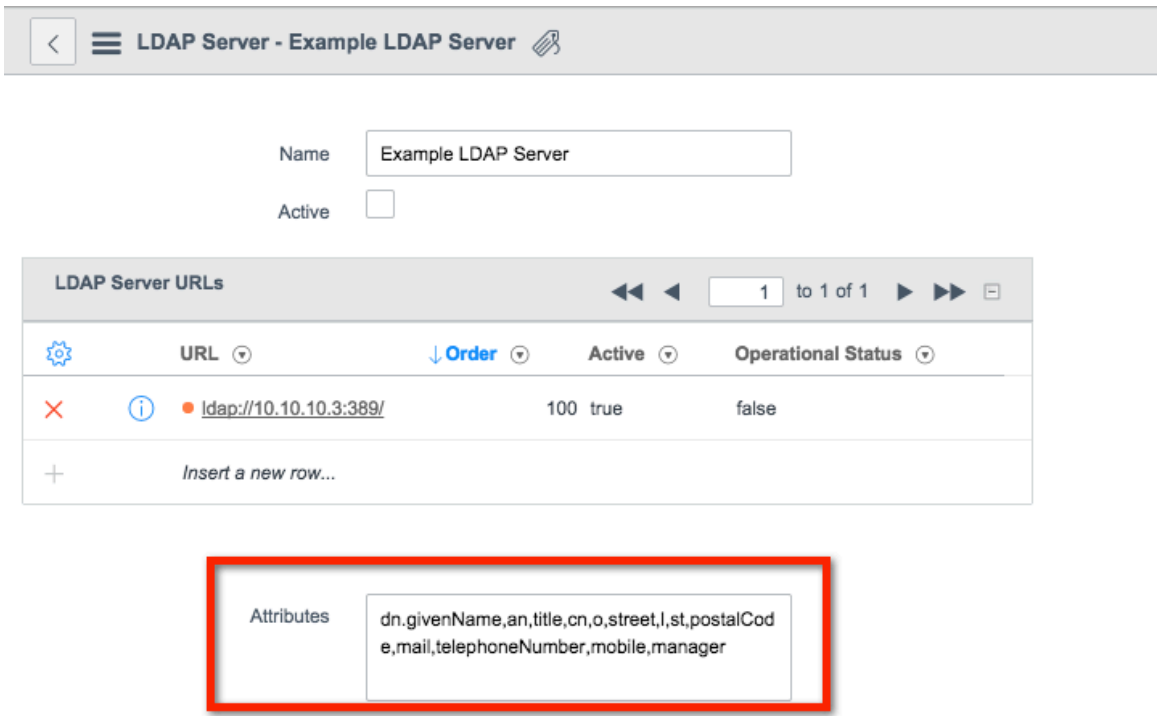
このタスクについて
デフォルトでは、LDAP サーバーからの読み取り権限を持つ各オブジェクトのすべての属性がロードされます。[属性] フィールドを使用して、LDAP クエリが返す属性を指定して制限することができます。大規模な LDAP インポートにこのアプローチを使用すると、インポートの速度を大幅に向上させることができます。

手順

可能な場合は属性を明示的に定義します。

インスタンスに公開しない情報がある場合は、 属性を除外します。LDAP サーバー属性を指定しないと、新しい属性が LDAP サーバーオブジェクトに追加されたときに、新しい属性からのデータのロードがビジー状態になるため、ユーザートランザクションが長時間フリーズすることがあります。

注: 「LDAP データの変換マップの選択または作成 (Select or Create a Transform Map for LDAP Data)」で説明されているマネージャールックアップスクリプトを使用するには、[属性] フィールドで **[manager]** と **[dn]** (識別名) を指定します。どちらの属性も変換マップの一部である必要はありません。



LDAP 接続をテストする

ユーザーが [LDAP サーバー] フォームを開くたびに、インスタンスによって接続が自動的にテストされます。または、[LDAP サーバー] フォームから LDAP サーバーへの接続を手動でテストすることもできます。

始める前に
必要なロール：admin

このタスクについて
デフォルトでは、LDAP サーバーへの接続に問題がある場合、[LDAP サーバー] フォームにエラーメッセージが表示されます。

注: 従業員は、インスタンスと LDAP サーバー間の接続を確認することもできます。LDAP 接続の確認については、テクニカルサポートにお問い合わせください。

接続を手動でテストするには：

手順

1. 移動先 すべて > システム **LDAP** > **LDAP** サーバー.
2. テストする LDAP サーバーを選択します。

3. [関連リンク] で、[テスト接続] をクリックします。
4. [関連リンク] で [参照] をクリックし、適切な LDAP ディレクトリ構造がシステムに表示されていることを確認します。
5. オプション: 接続に成功した場合は、[参照] をクリックして、インスタンスに表示されるソース LDAP ディレクトリ構造を表示します。

   注: [参照] ウィンドウの左側の [フィルター] および **[RDN]** フィールドは、右側の検索フィールドを使用すると無視されます。

結果
インスタンスは、接続テストの結果に応じて LDAP サーバーの運用ステータスを変更します。

- インスタンスが [運用ステータス] の値が [ダウン] のサーバーへの接続を確立すると、[運用ステータス] の値は自動的に [稼働中] に変更されます。この機能は、自動接続テストと手動接続テストの両方でサポートされています。
- [運用ステータス] の値が [稼働中] のサーバーへの接続を確立できない場合、[運用ステータス] の値は自動的に [ダウン] に変更されます。この機能は、手動テストではなく、自動接続テストでのみサポートされています。

LDAP 組織単位の定義

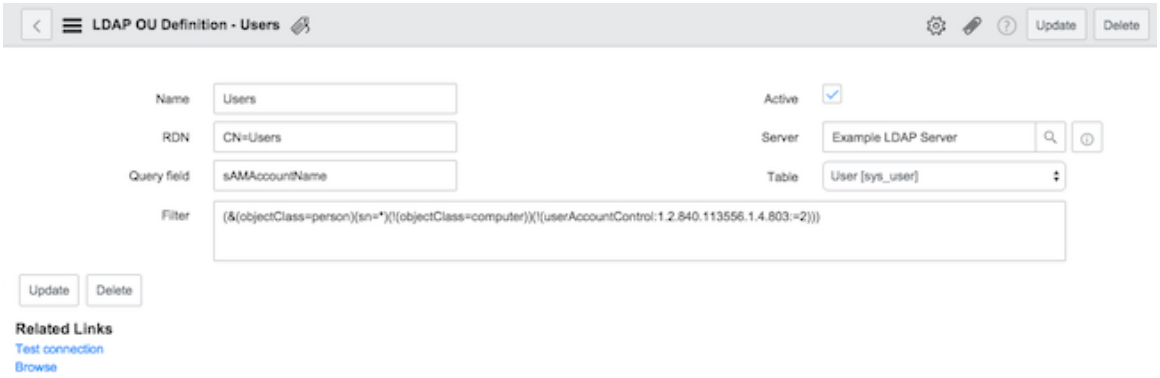
組織単位 (OU) の定義は、統合に利用可能な LDAP ソースディレクトリを指定します。

始める前に
必要なロール：admin。

このタスクについて
OU 定義には、場所、ユーザー、またはユーザーグループを含めることができます。すべての LDAP サーバー定義には、システムへのグループのインポート用とユーザー用の 2 つのサンプル OU 定義が含まれています。

手順

1. 移動先 すべて > システム **LDAP** > **LDAP** サーバー.
2. 設定する LDAP サーバーを選択します。
3. **[LDAP OU** 定義] 関連リストで、[グループ] または [ユーザー] のいずれかのサンプル OU 定義を選択します。
4. [LDAP OU 定義] フォームに入力します (表を参照)。
5. [更新] をクリックします。
   LDAP サーバーへの接続が自動的にテストされます。
6. [関連リンク] で、[参照] をクリックして、OU 定義が返す LDAP ディレクトリレコードを表示します。



OU 定義フォーム

フィールド	説明
名前	この OU を参照するときに統合で使用する名前を指定します。ここに入力した名前は、データソースレコードの LDAP ターゲットになります。
RDN	検索するサブディレクトリーの相対識別名を指定します。この RDN は、LDAP サーバー定義の検索開始ディレクトリーと組み合わされて、この組織単位の情報を含むサブディレクトリーを識別します。たとえば、サンプル OU 定義では、RDN 値 CN = Users を使用して、LDAP ディレクトリー CN=Users,DC=service-now,DC=com およびこのポイントより下の任意のディレクトリーを検索します。このフィールドは、LDAP システムのサブディレクトリーと一致する必要があります。
クエリフィールド	レコードを照会する LDAP サーバー内の属性の名前を指定します。クエリフィールドは、単一のドメインインスタンスと複数のドメインインスタンスの両方で一意である必要があります。最良の結果を得るには、複数のドメインインスタンスでユーザーを一意に識別するメールアドレスまたはその他の認証情報を使用します。Active Directory は **sAMAccountName** 属性を使用します。他の LDAP サーバーは、**cn** 属性を使用する傾向があります。 **注:** [クエリ] フィールドは、ユーザー [sys_user] テーブルの [ユーザー **ID]** フィールドにマップする必要があります。たとえば、Active Directory ユーザーが joe.example としてログインする場合、[ユーザー**ID]** の値 **joe.example** を持つユーザーレコードと、**sAMAccountName** の値 **joe.example** を持つ LDAP レコードが必要です。
有効	OU 定義を有効にし、アドミニストレーターがデータのインポートをテストできるようにするには、このチェックボックスをオンにします。ただし、統合では、有効な OU 定義からのみシステムにデータを取り込むことができます。
テーブル	LDAP サーバーからマップされたデータを受け取るテーブルを指定します。ユーザーの場合は [ユーザー **(sys_user)]** を選択し、グループの場合は [グループ **(sys_group)]** を選択します。
フィルター	LDAP フィルター文字列を入力して、OU からインポートする特定のレコードを選択します。LDAP フィルタークエリが具体的であればあるほど、クエリの効率は高くなります。 たとえば、ユーザー LDAP OU 定義では、次のフィルターを使用して、個人として分類され、**sn** 属性値を持ち、コンピューターではなく、無効としてフラグ付けされていないレコードを選択します。

自動翻訳

フィールド	説明
	(&(objectClass=person)(sn=*)(!(objectClass=computer)) (!(userAccountControl:1.2.840.113556.1.4.803:=2))) LDAP フィルター構文の説明については、インターネットで LDAP フィルター RFC を検索してください。

Example: 組織単位の定義の例

次のディレクトリ構造の LDAP サーバーがあるとします。

dc=my-domain,dc=com

- ou=Groups
  - cn=Development
  - cn=HR
  - cn=Sales
- ou=Users
  - ou=Development
  - ou=HR
  - ou=Sales

さらに、アプリケーションから人事グループと人事ユーザーを除外するとします。次の操作を実行します。

1. 検索開始ディレクトリが dc=my-domain,dc=com の LDAP サーバーレコードを作成します。
2. cn=HR を除外するフィルターで ou=Groups の OU 定義レコードを作成します。
3. ou=HR を除外するフィルターで ou=Users の OU 定義レコードを作成します。

OU 定義で追加の属性またはフィルターを指定しない場合、LDAP クエリは開始ディレクトリと RDN からサブツリー全体を返します。

これらの例では、RDN 値が ou=Groups で、フィルターがない OU 定義はすべてのグループを返します。同様に、RDN 値が ou=Users でフィルターがない OU 定義では、すべてのユーザーと子組織単位が返されます。

LDAP のデータソースを作成する

各 LDAP 組織単位 (OU) 定義には、独自のデータソースの関連リストがあります。

始める前に
必要なロール：admin

このタスクについて

注: テストロードアクションを正しく機能させるには、**[LDAP** サーバー] と **[LDAP OU** 定義] の両方を有効にする必要があります。テストロードが初めて有効になると、インポートセットフィールドの長さを決定するために最大 20 のレコードがサンプリングされます。サンプリングされたレコードに [ユーザー **ID]** フィールドの値が含まれていない場合、後続のすべてのインポートのフィールド長がデフォルトの長さの 40 に設定されます。インポートでは、インポートされたデータがインポートセットテーブルのフィールド長を超えている場合、超えた分は切り捨てられます。また、[ユーザー **ID]** フィールドは最大 40 文字に短縮されます。ロードされた 20 件のレコードは変換できず、テストのみを目的としていることに注意してください。テストレコードに [ユーザー **ID]** フィールドの値が含まれている場合、フィールド長はテストレコード内の最長のユーザー ID のフィールド長に基づいて設定されます。

新しいデータソースを作成するには：

手順

1. 移動先 すべて > システム **LDAP** > **LDAP** サーバー.
2. 設定する LDAP サーバーを選択します。
3. **[LDAP OU** 定義] 関連リストで、[グループ] や [ユーザー] などのアイテムを選択します。
4. [データソース] 関連リストで、[新規] をクリックします。
5. [データソース] フォームに入力します (表を参照)。
6. [送信] をクリックします。
7. [関連リンク] で、**[20** 件のレコードのテストロード] をクリックして、データソースが LDAP データをインポートテーブルに取り込むことができるかどうかをテストします。

[データソース] フォーム

フィールド	説明
名前	このデータソースを参照するときに統合で使用する名前を指定します。
インポートセットテーブル名	インポートされた LDAP レコードと属性が一時的に配置されるステージングテーブルの名前を入力します。インポートされた LDAP レコードを表示するには、このテーブルを確認してください。すべての LDAP データソースに同じインポートセットテーブル名を使用できます。
タイプ	インポートされたデータが LDAP データであることを示すには、**[LDAP]** を選択します。タイプ **[LDAP]** を選択すると、フォームに **[LDAP** ターゲット] フィールドが表示されます。
LDAP ターゲット	このデータソースに関連付けられた LDAP OU 定義を選択します。

LDAP ユーザーを自動でプロビジョニングする

LDAP サーバーにいるが、まだインスタンスにないユーザーは自動的にプロビジョニングされます。

始める前に
必要なロール：admin

手順

システムのプロパティ [sys_properties] テーブルに次のプロパティを作成します。

LDAP プロパティ

LDAP プロパティ	説明
glide.ldap.authentication	LDAP を使用して LDAP 認証を有効にし、ユーザーを認証します。このプロパティを **[true]** (デフォルト値) に設定します。
glide.ldap.user.autoprovision	ユーザーが LDAP に存在するが、まだインスタンスに存在しない場合、LDAP によりユーザー [sys_user] テーブルにユーザーが自動的に作成されます。このプロパティを **[true]** (デフォルト値) に設定します。

自動プロビジョニングを機能させるには、これらのプロパティを両方とも **[true]** に設定する必要があります。

MID サーバーを介した **LDAP** 統合

アドミニストレーターは、管理、計測、ディスカバリー (MID) サーバー上で LDAP データソースを使用して統合できます。

MID サーバーは、ServiceNow AI Platform と外部アプリケーション、データソース、およびサービス間のデータの通信と移動を可能にします。MID サーバーのインストールの詳細については、「MID サーバーのインストール」を参照してください。

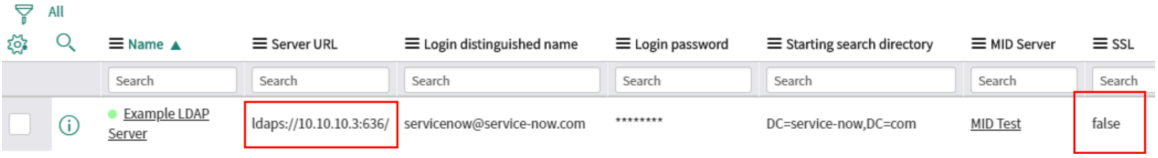
MID サーバーを使用して LDAP 接続を確立すると、LDAP サーバーを外部ネットワークトラフィックに公開する必要がなくなります。LDAP サーバーとデータセンター間に VPN トンネルを確立する必要がなくなります。LDAP サーバー設定レコードを読み取れるようにするには、MID サーバーユーザーに user_admin ロールが必要です。

注: MID サーバーでは UI アクション <インスタンス>/sys_ui_action.do?sys_id=1b4f7ef30a0001060058e223c9a5744c を使用して、LDAP からユーザーおよびグループのレコードをリフレッシュすることはできません。

MID サーバー接続は、デフォルトではポート 80 で HTTP を介して通信します。この通信チャネルには証明書は必要ありません。MID サーバーとインスタンス間の接続は HTTPS (ポート 443) です。インスタンスは、LDAP または LDAPS を使用して LDAP サーバーに直接接続します。この接続は、インターネット経由または VPN トンネル経由のいずれかを使用することができます。

注: LDAP は、パスワード認証を使用して MID サーバーを介して通信することはできません。

SSL を介した安全な通信のために、MID サーバーに SSL 証明書を追加する 必要があります。LDAP サーバーの URL を LDAP から LDAPS に変更し、ポート 636 に変更します。



注: 新しい LDAP サーバーを作成すると、MID サーバーの SSL フラグはデフォルトで [false] に設定されます。この動作は無視できます。

特定の LDAP サーバーの接続プロパティを設定するには、「LDAP サーバーを定義する」を参照してください。

自動翻訳

LDAP 接続の監視を設定する

LDAP 接続の監視と通知を変更または無効にします。

始める前に
必要なロール：admin

このタスクについて
LDAP サーバーの接続が失敗すると、LDAP Admins グループに設定されたユーザーにインスタンスが自動的にメールを送信します。これは、**LDAP** 接続テストのスケジュール済みジョブによって起動される のメール通知を使用します。このメール通知はデフォルトで有効になっています。

> **注:** LDAP アドミングループに少なくとも 1 人のメンバーが存在しない限り、インスタンスはメール通知を送信しません。メールを受信するユーザーをこのグループに入力してください。

デフォルトでは、スケジュール済みジョブは 15 分ごとに接続をテストします。この間隔を変更するか、監視を無効にするには：

手順

1. 移動先 すべて > システム定義 > スケジュール済みジョブ.
2. **[LDAP** 接続テスト **(LDAP Connection Test)]** を開きます。
3. 次のいずれかの操作を行います。
   - [繰り返し間隔] フィールドで間隔を変更します。
   - [アクティブ] チェックボックスをオフにして、監視を無効にします。

MID サーバーを介してバイナリデータをインポートする

アドミニストレーターは、MID サーバーを介して LDAP 統合によりバイナリラージオブジェクト (BLOB) データをインポートできます。

始める前に
必要なロール：admin

このタスクについて

手順

1. バイナリデータをインポートする LDAP 列の名前をシステムプロパティ glide.ldap.binary_attributes に追加します。
2. 名前が glide.ldap.binary_attributes で、システムプロパティに設定した値と同じ MID サーバープロパティを追加します。

MID サーバーを介して LDAP 統合をトラブルシューティングする

MID サーバーを介して LDAP を統合するときに、次の領域で問題が発生する可能性があります。

これらの問題のトラブルシューティングを行うには、外部通信チャネル (ECC) キュー (ディスカバリー > 出力とアーティファクト > **ECC** キュー) を開きます。

テスト接続の問題

サーバー内で OU を定義する場合は、[テスト接続] 関連リストを使用して LDAP 接続を確認します。このリンクをクリックすると、ECC キューには、**[LDAPConnectionTesterProbe]** のトピック名を持つ単一の出力メッセージが表示されます。MID サーバーでテストが完了すると、ECC キューに同じトピック名の入力メッセージが表示されます。入力メッセージの [名前] 列に **[true]** が表示さ

れている場合、テストは成功しています。レコードにドリルダウンしてペイロードを表示し、エラーメッセージが含まれていないことを確認します。

テスト接続

Created	Agent	Topic	Name	Source	Queue	State	Processed
2013-07-29 13:24:17	mid.server.local_mid	LDAPConnectionTesterProbe	true	04a952038f21010036bf21ca47e79a30	input	processed	2013-07-29 13:24:19
2013-07-29 13:24:04	mid.server.local_mid	LDAPConnectionTesterProbe		04a952038f21010036bf21ca47e79a30	output	processed	2013-07-29 13:24:17

問題を参照する

サーバー内で OU を定義する場合、OU 定義が返す LDAP ディレクトリレコードを表示するために使用される [参照] 関連リストがあります。このリンクをクリックすると、 ECC キューに **[LDAPBrowseProbe]** のトピック名を持つ単一の出力メッセージが表示されます。MID サーバーからデータが返された後、ECC キューに同じトピック名の入力メッセージが表示されます。入力メッセージの [名前] 列に **[true]** が表示されている場合、テストは成功しています。レコードにドリルダウンしてペイロードを表示し、エラーメッセージが含まれていないことを確認します。

インポート負荷の問題

データをアップロードするとき (たとえば、20 件のレコードのテストロード機能を使用するとき)、ECC キューには、**[LDAPProbe]** のトピック名を持つ単一の出力メッセージが表示されます。

MID サーバーからデータが返された後、ECC キューには **[LDAPProbeCompleted]** と呼ばれる別の入力メッセージが表示されます。この入力メッセージの [名前] 列には、返されたレコードの合計数が表示されます。

[LDAPProbe] という名前の追加の入力メッセージが表示されます。この入力メッセージの [名前] 列には、バッチ内の最大のレコード番号が表示されます。返されたレコードの合計数が 258 で、バッチサイズが 200 (デフォルト) の場合、2 つの LDAPProbe (200、258) 受信メッセージが受信され、1 つの LDAPProbeCompleted (258) 受信メッセージが受信されます。

レコードにドリルダウンしてペイロードを表示し、エラーメッセージが含まれていないことを確認します。

インポート負荷

Created	Agent	Topic	Name	Source	Queue	State	Processed
2013-07-29 13:09:48	mid.server.local_mid	LDAPProbeCompleted	11	ed0a0d7a8f32010036bf21ca47e79a56	input	processed	2013-07-29 13:09:51
2013-07-29 13:09:48	LDAPProbeResult	LDAPProbe	11	ed0a0d7a8f32010036bf21ca47e79a56	input	processed	2013-07-29 13:09:51
2013-07-29 13:09:36	mid.server.local_mid	LDAPProbe		ed0a0d7a8f32010036bf21ca47e79a56	output	processed	2013-07-29 13:09:46

[LDAPProbeError] という名前の出力メッセージにも注意してください。

エラーメッセージ

All > Created on Today > Topic = LDAPProbeError

Created	Agent	Topic	Source	Queue	State	Processed	Error string
2014-02-20 14:52:55	mid.server.localdublinmid	LDAPProbeError	MID Server reported error: java.lang.Exc...	output	error	2014-02-20 14:53:02	No message handler for this message.

[名前] 列のリンクをクリックして、エラーの詳細を表示します。

LDAP ページング

LDAP サーバーのページングサイズが 1000 未満の場合、LDAP ページングは機能しません。MID サーバープロパティ *glide.ldap.max_results* を LDAP サーバーのページングサイズ以下の値に設定します。

LDAP がバイナリデータのインポートに失敗する

ユーザーの写真などのバイナリデータを LDAP 経由でインポートするには、MID サーバープロパティ *glide.ldap.binary_attributes* にバイナリ属性を含める必要があります。ユーザーの写真の例では、属性は「jpegphoto」になります。

データのインポートとマッピング

LDAP インポートマップは、LDAP データベースのフィールドをインスタンスのフィールドと照合します。

注: LDAP マッピングはパフォーマンスに影響するため、ピーク時間外にスケジュールするか、一度に処理するレコードを少数にしてシステムの可用性を維持することをお勧めします。

必要な属性または要求されている属性のみをインポートする変換マップを定義します。使用しているインスタンスのバージョンに応じて、LDAP マッピング関係を指定する方法は異なります。

LDAP 統合にシステム LDAP アプリケーションを使用するバージョンを実行しているかどうかを確認する最も簡単な方法は、アプリケーションナビゲーターからアプリケーションを見つけることです。

[ビジネスルールを実行] オプションはターゲットテーブルにのみ適用されます。ターゲットテーブルに関連付けられた変換マップのみが、さまざまなテーブルに関連付けられたビジネスルールを実行します。ユーザーグループを更新していて、ユーザーグループテーブルで実行されているビジネスルールがある場合は、そのグループにロールが定義されている必要があります。

LDAP インポートマッピングオプション

システム LDAP アプリケーションは？	マップ
あり	変換マップを使用してマッピングを指定します。
なし	LDAP の従来のインポートマップを使用して、マッピング、またはベースラインインスタンスに含まれるデフォルトの LDAP 変換を指定します。正しいフィールドと一致するように [結合] フィールドを忘れずに調整してください。

スケジュール済みインポート

アドミニストレーターは、スケジュール済みインポートを使用して、定期的に LDAP データをインポートできます。デフォルトでは、LDAP 統合には次の 2 つのスケジュール済みインポートが含まれています。

- サンプル LDAP ユーザーインポート (Example LDAP User Import)
- サンプル LDAP グループインポート (Example LDAP Group Import)

どちらのサンプルもデフォルトではアクティブではありません。会社のビジネスニーズに合わせてこれらのスケジュール済みインポートを変更します。

LDAP 変換マップ

変換マップは、データをインポートセットテーブルからターゲットテーブル (ユーザーまたはグループ) に移動します。

LDAP 統合では、標準のインポートセットと変換マップが使用されます。カスタム LDAP 変換マップを作成することもできます。

重要: カスタム LDAP 変換マップを選択するか作成するかにかかわらず、一連のソースおよびターゲットテーブルに対して 1 つのアクティブな変換マップが必要です。一致するフィールドに対して結合しない限り、同じソーステーブルとターゲットテーブルに対して複数の変換マップを有効にすると、ターゲットテーブルに重複するエントリが生成される可能性があります。

デフォルトの LDAP 変換マップ

デフォルトでは、LDAP データの変換マップは 2 つあります。

デフォルトの **LDAP** 変換マップ

変換マップ	ソーステーブル	ターゲットテーブル	説明
LDAP ユーザーインポート	[ldap_import]	[sys_user]	LDAP オンデマンドログインの一部として LDAP 認証情報からユーザーレコードを作成するためのデフォルトの変換マップ。Active Directory LDAP サーバーのマッピングが含まれています。
LDAP グループインポート	[ldap_group_import]	[sys_user_group]	LDAP OU からグループレコードを作成するためのデフォルトの変換マップ。Active Directory LDAP サーバーのマッピングが含まれています。

注: デフォルトでは、LDAP 部門レコードの変換マップはありません。

カスタム LDAP 変換マップの要件

カスタム変換マップを作成する場合、変換マップは次のマッピング要件を満たす必要があります。

カスタム **LDAP** 変換マップの要件

ソーステーブル	ソースフィールド	ターゲットテーブル	ターゲットフィールド	結合	説明
ldap_import	u_source	sys_user	ソース	false	**[u_source]** フィールドは、インポートされたユーザーまたはグループの LDAP DN を識別します。システムはこのフィールドを使用して、ユーザーに LDAP 認証が必要かどうかを判断し、ユーザーのマネージャーを見つけて、ユーザーをグループに入れます。
ldap_import	次のいずれかのフィールドを選択します。	sys_user	user_name	true	LDAP が Active Directory に統合されている場合は、ソースフィールドとして

自動翻訳

カスタム **LDAP** 変換マップの要件 (続く)

ソーステーブル	ソースフィールド	ターゲットテーブル	ターゲットフィールド	結合	説明
	• u_samaccountname • u_dn • u_cn				**[u_samaccountname]** を選択します。他の LDAP ディレクトリを使用する場合は、ソースフィールドとして **[u_dn]** または **[u_cn]** を選択します。

LDAP 変換マップと従来のインポートマップの違い

変換マップを使用して LDAP マッピング関係を指定する場合、マネージャーと部門で参照フィールドの設定方法に大きな違いがあります。

変換マップを使用する場合は、変換スクリプトを使用して参照を作成する必要があります。これは、「manager」などの LDAP 属性に関連付けられた値がマネージャーの識別名 (DN) であるためです。

追加のロジックがない場合、LDAP でのユーザーの識別名であるマネージャー名でユーザーレコードが作成されます。統合には、これらの参照の作成を容易にする変換スクリプトが含まれています。デフォルトの変換マップ「LDAP ユーザーインポート」には、これらの参照の変換スクリプトが含まれています。

既存のマッピング関係

従来のインポートマップを変換マップに更新する場合、システム LDAP アプリケーションが追加される前に存在していた LDAP マッピング関係を保持できます。LDAP サーバーには、従来のインポートマップへの参照である [マップ] フィールドがあります。

注: デフォルトでは、このフィールドは非表示になっているため、表示するようにフォームを構成する必要があります。

変換マップの使用に移行する場合は、従来のインポートマップへの参照をクリアします。

LDAP インポートマップ設定

属性を確認して使用し、統合が LDAP ソースからインポートするフィールドを制限します。さらに、user_name フィールドをユーザーのログイン ID を含む LDAP 属性にマッピングすることが重要です。Active Directory の場合、これは通常 sAMAccountName 属性です。バイナリ属性 (objectSID や objectGUID など) をインポートして結合する場合は、カスタム変換スクリプトを作成する必要があります。

注: user_name フィールドにマッピングされた値は一意である必要があります。

変換マップ (LDAP ユーザーインポートなど) を指定しない場合、統合では次のデフォルトマッピングが使用されます。

LDAP インポートのデフォルトマッピング

ユーザーフィールドまたは変数	**LDAP** 属性
user_name	sAMAccountName

自動翻訳

LDAP インポートのデフォルトマッピング (続く)

ユーザーフィールドまたは変数	**LDAP** 属性
email	mail
phone	telephoneNumber
home_phone	homePhone
mobile_phone	mobile
first_name	givenName
last_name	sn
title	title
department	department
manager	manager
middle_name	initials
u_memberof	groups
u_member	メンバー
u_manager	manager

LDAP データ変換

LDAP 属性に簡易データが含まれている場合、変換マップはインポートされた LDAP 属性をターゲットテーブル (ユーザーまたはグループ) にリンクします。たとえば、sAMAccountName 属性のサンプルデータは、ユーザーテーブルの [ユーザー **ID**] フィールドにマッピングされます。

インポートされた LDAP データが参照フィールドにマッピングされると、インスタンスは既存の一致するレコードを検索します。一致するレコードが存在しない場合、フィールドマッピングで指定されていない限り、インスタンスは参照フィールドの新しいレコードを作成します。

たとえば、ユーザーテーブルの [場所] 参照フィールドに対する LDAP 属性 l マップを想定します。インポートによって既存の場所レコード値と一致しない属性値が取り込まれるたびに、変換マップによって新しい場所レコードが作成されます。新しい場所レコードはインポートされた属性と同じ値になり、インポートされたユーザーレコードには新しい場所レコードへのリンクが含まれます。

ただし、LDAP 属性が識別名 (DN) を返す場合があります。これは、基本的に LDAP ディレクトリ内の別のレコードへの参照です。たとえば、マネージャー属性には通常、現在の LDAP ディレクトリエントリのマネージャーの識別名が含まれています。インポートされた DN は通常、 cn=Beth Anglin,ou=Users,dc=my-domain,dc=com のような長いテキスト文字列を使用します。

⚠ **警告:** ターゲットフィールドが DN を含むのに十分な長さであることを確認してください。多くのテキストフィールドでは、デフォルトの長さである 40 が使用されています。これは、一部の DN 値では長さが足りない場合があります。ServiceNow システムでは、フィールドの長さを超える値は切り捨てられます。

新しいユーザーは既存のユーザーと関連付けられていないため、アドミニストレーターは通常、DN 値から新しいユーザーが作成されることを望んでいません。代わりに、アドミニストレーターは、インポートでマネージャーの既存のユーザーレコードを見つけて、新しくインポートされたユーザーに関連付けたいと考えています。LDAPUtils スクリプトインクルードには、DN を解析して既存のユーザーを検索できる setManager 関数と processManagers 関数が含まれています。最良の結果を得るには、これらの関数を使用してカスタム変換マップを作成します。

たとえば、LDAP User Import 変換マップスクリプトは setManager 関数を呼び出します。

```
//
// The manager coming in from LDAP is the DN value for the manager.
// The line of code below will locate the manager that matches the
// DN value and set it into the target record.  If you are not
// interested in getting the manager from LDAP then remove or
// comment out the line below
ldapUtils. setManager (source , target ) ;
```

場合によっては、統合により、関連付けられたマネージャーのユーザーレコードがインポートされる前に、ユーザーのレコードがインポートされます。このようなケースを処理するには、変換の完了後に processManagers 関数を呼び出します。たとえば、**[LDAP** ユーザーインポート] 変換マップは、onComplete 変換スクリプトを使用して processManagers 関数を呼び出します。

```
// It is possible that the manager for a user did not exist in the database when // the
 user was processed and therefore we could not locate and set the manager field. // The
 processManagers call below will find all those records for which a manager could  // not be
 found and attempt to locate the manager again. This happens at the end of the  // import and
 therefore all users should have been created and we should be able to  // locate the manager
 at this point
ldapUtils. processManagers ( ) ;
```

LDAP 統合でマネージャー属性を使用しない場合は、setManager および processManagers 関数呼び出しを削除するかコメントアウトします。

LDAP スクリプティング

データをインポートするときの要件を指定するカスタム変換マップ、スクリプト、およびビジネスルールを作成します。

カスタム変換マップには、onStart および onAfter 変換スクリプトを含める必要があります。

onStart スクリプトは LDAPUtils スクリプトインクルードを呼び出してログ記録を開始する必要があります。たとえば、**[LDAP** ユーザーインポート] 変換マップには、次のコードを使用する onStart スクリプトがあります。

```
gs.include ( "LDAPUtils" ) ; var ldapUtils  = new LDAPUtils ( ) ;
ldapUtils. setLog (log ) ;
```

onAfter スクリプトは addMembers 関数を呼び出します。例：

```
ldapUtils.addMembers (source , target ) ;
```

無効な **Active Directory** ユーザーを非アクティブに設定する

関連付けられた AD ユーザーが無効になっている場合は、次のスクリプトを使用してユーザーを自動的に非アクティブ化します。

始める前に

必要なロール：admin

このタスクについて

無効な Active Directory ユーザーは、userAccountControl 属性の値を確認することで識別できます。このルールは、userAccountControl の値が変更されるたびに実行され、[ユーザーアカウント制御] が無効な AD アカウントを示している場合はユーザーアカウントを非アクティブ化します。

関連付けられた AD ユーザーが無効になっている場合は、次のスクリプトを使用してユーザーを自動的に非アクティブ化します。

手順

1. [ユーザー] フォームを設定し、[ユーザーアカウント制御] という名前の新しい整数フィールドを作成します。
2. userAccountControl (外部) のマッピングを新しいフィールドに追加します。
3. 次のプロパティを使用して、新しいビジネスルールを作成します。

AD ユーザーのビジネスルールを無効にする

[ビジネスルール] フィールド	値
名前	AD ユーザーを無効化
テーブル	ユーザー [sys_user]
時期	前
条件	current.u_user_account_control.changes()

[スクリプト] フィールドには、以下を含める必要があります。

```
var disabledFlag = 2;
//perform a bitwise comparison on userAccountControl to see if the 2 bit flag is enabled
if (current.u_user_account_control & disabledFlag) {
  gs.log('Disabling user: ' + current.user_name + 'userAccountControl=' +
 current.u_user_account_control);
  current.active='false';
  current.locked_out='true';
}
```

LDAP フィールド値の割り当て

スクリプトを使用して、フィールドマッピングがあるフィールドに値を割り当てることができます。

たとえば、[sys_user.company] フィールドに値を割り当てるには、[会社] フィールドのフィールドマップを作成し、次の変換スクリプトを追加します。

```
company = "Don's Sporting Goods";
```

特定の **LDAP** ユーザーを除外する

LDAP フィルタープロパティを使用して LDAP ユーザーリストを完全にフィルタリングできない場合は、マップスクリプトを使用してユーザーを除外できます。

インポートしないユーザーを識別するロジックを実行した後、user_name フィールドを空の文字列に設定すると、このユーザーはインポートされません。

```
user_name='';
```

除外するユーザーを識別する方法の 1 つは、DistinguusedName 属性で文字列を検索することです。たとえば、このスクリプトは、ユーザー OU にないアカウントを除外します。ユーザー OU が多すぎてターゲット OU LDAP オプションに含めることができない場合は、このスクリプトを使用できます。

```
//vdn is a variable mapped to distinguishedName
gs.include("LDAPUtils");
var vdn = source.getElement(this.distinguishedName);
if (vdn.indexOf('OU=Users')<0) {
  user_name='';
```

```
  gs.log('LDAP Import Skipping User: ' + vdn);
}
```

フィルタリングのより複雑な方法は、正規表現を使用することです。

```
//vcn is a variable mapped to cn
//vdn is a variable mapped to distinguishedName
//c is the regular expression string
gs.include("LDAPUtils");
var vdn = source.getElement(this.distinguishedName);
var vcn = source.getElement(this.cn);
var c = /^[a-z][a-z][a-z][0-9][0-9][0-9]$/;
var nvcn = vcn.toLowerCase();
//test to see if the cn is in the form of 3 letters followed by 3 numbers, only import these
if (c.test(nvcn)) {
 user_name = nvcn;
} else {
 gs.log("LDAP import rejected username: " + vcn + " for DN: " + vdn);
 user_name = "";
}
```

参照フィールドのインポートの選択肢アクションを設定する

LDAP 変換マップは、インポートセットテーブルのフィールドをインシデントやユーザーなどの既存のテーブルのフィールドにマッピングする方法を決定します。

始める前に
必要なロール：admin

このタスクについて
LDAP 変換マップがインポートセットテーブルのフィールドを更新すると、LDAP データに新しいレコードがあるたびに統合によって新しいレコードが自動的に作成されます。LDAP 変換マップが別のテーブルのデータを格納する参照フィールドを更新する場合、アドミニストレーターは新しい LDAP レコードを作成するか、無視するか、または却下するかを選択できます。
たとえば、統合が既存の部門と一致しない新しい部門レコードを受信した場合、インスタンスに新しい部門レコードを作成せずに他のすべての LDAP レコードフィールドを更新できます。変換マップを使用すると、各参照フィールドのレコード作成オプションを設定できます。

手順

1. 移動先 すべて > システム **LDAP** > 変換マップ.
2. [フィールドマップ] 関連リストで、[選択肢アクション] フィールドから次のいずれかのアクションを選択します。
   - 作成 – 一致するレコードが存在しない場合、新しい参照フィールドレコードを作成します。
   - 無視 – 参照フィールド内の新しいレコードを無視し、変換マップ内の他のすべてのフィールドの処理を完了します。
   - 却下 – レコード全体の変換を停止します。

   注: フィールドマップには、参照フィールドの [選択肢アクション] フィールドのみが表示されます。

LDAP マッピングを確認する

LDAP 変換マップを作成したら、LDAP データをリフレッシュして、変換マップが想定どおりに機能することを確認します。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > システム **LDAP** > ロードスケジュール.
2. LDAP インポートジョブをクリックします。
3. [今すぐ実行する] をクリックします。

LDAP 統合をトラブルシューティングする

LDAP サーバーを統合する際に質問がある場合は、以下のアイテムが問題のトラブルシューティングに役立つ可能性があります。

予備チェック

- LDAP が利用できない場合、ユーザーはインスタンスにログインできません。LDAP が停止した場合でもアドミニストレーターがインスタンスにアクセスできるように、アドミン用のローカルアカウントを用意することをお勧めします。
- サービスアカウントが期限切れになっていないか、またはロックアウトされていないかを確認します。
- ユーザー名の形式を確認してください。ユーザー名のみを使用する代わりに、ユーザー名とドメイン (ユーザー名@ドメイン) を使用してみてください。
- ldap_server_config レコードの system_id エントリを変更したことを確認します。更新セットを使用して意図せず system_id を変更すると、system_id はターゲットインスタンスの間違ったノードを指し示し、機能しません。

エラーコード

LDAP ログファイルには、LDAP と Active Directory (AD) の両方の業界標準エラーコードが一覧表示されます。LDAP ログファイルはラッパーファイルに含まれています。LDAP エラーコードは 2 桁の数字ですが、Active Directory のエラーコードは 3 桁の数字です。最も一般的なエラーコードのリストについては、「LDAP エラーコード」を参照してください。

複数のドメインを統合する

同じフォレスト内または完全に信頼されていないドメイン内で複数のドメインを統合できます。ドメインごとに個別の LDAP サーバーレコードを作成することをお勧めします。各 LDAP サーバーレコードは、指定されたドメインのドメインコントローラーを指す必要があります。これは、各ドメインコントローラーへの接続を許可する必要があることを意味します。1 つの LDAP アカウントでの LDAP による複数の AD フォレストはサポートされていません。

複数のドメインに展開する場合は、アプリケーションのユーザー名に一意の LDAP 属性を特定し、結合値をインポートすることが重要です。Active Directory の一般的な一意の結合属性は objectSid です。一意のユーザー名は、LDAP データ設計によって異なります。一般的な一意の属性は email または userPrincipalName です。

受信レコード

参照フィールドで一致する値が欠落している受信 LDAP レコードを統合で処理する方法を設定するには、「LDAP 変換マップ」を参照してください。

一般的な認証エラー

- ユーザーがログインできない (無効な DN)
- 無効な CN
- 無効な接続

自動 **LDAP** 接続テスト

LDAP サーバーへの接続を手動でテストすることも、ServiceNow で自動的に接続をテストすることもできます。

接続が自動的にテストされます。

- ユーザーが [LDAP サーバー] フォームを開くたびに行われます。
- デフォルトで 15 分ごとに実行される LDAP 接続テストのスケジュール済みジョブを使用します。

  このスケジュール済みジョブの実行頻度は変更できます。このスケジュール済みジョブが接続を確立できない場合、新しい 1 回限りのスケジュール済みジョブによって、5 分またはスケジュール済みジョブの [繰り返し間隔] 値の半分のいずれか早い時点で接続テストが再試行されます。

LDAP サーバーへの接続に問題がある場合は、フォームにエラーメッセージが表示されます。MID サーバーの背後にあるサーバーのテスト接続もサポートされています。

LDAP モニターを表示する

LDAP モニターを使用して、LDAP サーバーとリスナーに関する現在の情報を表示できます。

始める前に
必要なロール：admin

このタスクについて
使用可能な状況は、次のとおりです。

- アクティブ
- 非アクティブ
- エラー
- アクティブ (シャットダウン中...)
- エラー (シャットダウン中...)

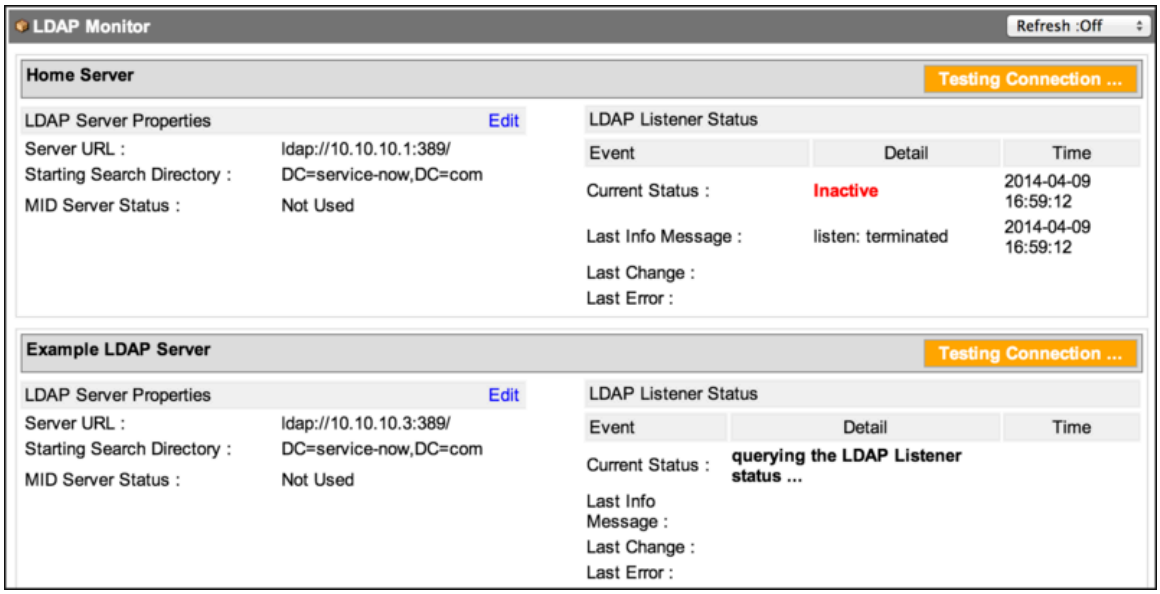
現在のステータスに加えて、モニターには次の情報も表示されます。

- LDAP 変更の待機中、接続エラーなど、リスナーによって検出された最後のメッセージ。
- 新規ユーザー、更新されたユーザーなど、前回の LDAP ユーザーの変更。
- 発生した最後のエラー。

LDAP モニターを表示するには：

手順

移動先 すべて > **LDAP** > システム **LDAP** > **LDAP** モニター.



画面のプロパティとフィールドの説明については、表を参照してください。

LDAP モニター

フィールド	説明
リフレッシュ	リフレッシュレートを設定するには、[LDAP サーバーモニター (LDAP Server Monitor)] ヘッダーバーの [リフレッシュ] フィールドをクリックし、各データのリフレッシュ間隔を秒数で選択します。[なし] を選択してリフレッシュを抑制することもできます。
接続ステータス	サーバー接続インジケーターは、[LDAP リスナー状態] フィールド上部の右側にあります。サーバーが接続されると、ボックスが緑色になり、「接続」と表示されます。サーバーが接続されていない場合、ボックスが赤色になり、「未接続」と表示されます。サーバー接続のテスト中は、ボックスが黄色になり、「接続をテストしています」と表示されます。
LDAP サーバープロパティ	
編集	LDAP サーバーを監視するときは、[LDAP サーバーモニター (LDAP Server Monitor)] 画面で [編集] をクリックしてプロパティを変更できます。
サーバー URL	LDAP サーバーが待機しているサーバー名とサーバーポートの組み合わせ。多くの場合、ポートは次のいずれかに設定されます。 • 389：クリアテキストで LDAP に接続するためのデフォルトのポート • 636：SSL 接続を介して LDAP に接続するための標準ポート 値の例：ldap://10.10.10.3:389/ LDAP サーバーに複数の URL アドレスがある場合があります。これは、別の LDAP サーバーエントリを作成することによってデータをインポートできる複数のディレクトリ構造を確立するわけではありませんが、複数の LDAP サーバーがある場合は単一障害点を回避するために提供します。LDAP URL アドレスはスペース文字で区切られ、有効な接続が確立されるまで各サーバーアドレスが順番に試行されます。
検索開始ディレクトリ	システムがユーザーまたはグループの検索を開始するディレクトリまたは RDN (相対識別名)。 値の例：DC=service-now,DC=com

自動翻訳

フィールド	説明
	このポイントより上のデータはインポートできません。インスタンスは、LDAP 階層内の指定されたディレクトリおよびその下のディレクトリを可視化できます。
MID サーバーステータス	MID サーバーの現在の接続ステータス。
LDAP リスナーステータス	
現在のステータス	リスナーがアクティブかどうかを示します。
最後の情報メッセージ	ユーザーとグループの変更に関連して LDAP サーバーが最後に受信したメッセージと、メッセージが受信された時間を表示します。
前回の変更	LDAP サーバーに最後に加えられた変更と、それが行われた時刻を表示します。
前回エラー	LDAP サーバーで最後に発生したエラーと、その発生時刻が表示されます。

LDAP エラーコード

LDAP ログファイルには、LDAP と Active Directory (AD) の両方の業界標準エラーコードが一覧表示されます。

標準のエラーコード

標準的な **LDAP** エラー

エラーデータコード	テキスト	説明
0	LDAP_SUCCESS	要求されたクライアント操作が正常に完了したことを示します。
2	LDAP_PROTOCOL_ERROR	サーバーがクライアントから無効な要求または誤った形式の要求を受信したことを示します。
3	LDAP_TIMELIMIT_EXCEDED	クライアントまたはサーバーによって指定された操作の時間制限を超えたことを

標準的な **LDAP** エラー (続く)

エラーデータコード	テキスト	説明
		示します。検索操作では、不完全な結果が返されます。
4	LDAP_SIZELIMIT_EXCEEDED	検索操作で、クライアントまたはサーバーによって指定されたサイズ制限を超えたことを示します。不完全な結果が返されます。
5	LDAP_COMPARE_FALSE	エラー状態を示すものではありません。比較操作の結果が false であることを示します。
6	LDAP_COMPARE_TRUE	エラー状態を示すものではありません。比較操作の結果が true であることを示します。
7	LDAP_AUTH_METHOD_NOT_SUPPORTED	バインド操作中に、クライアントが LDAP サーバーでサポートされていない認証方法を要求したことを示します。
8	LDAP_STRONG_AUTH_REQUIRED	次のいずれかを示します。 バインド要求で、LDAP サーバーは強力な認証のみを受け入れます。クライアント要求で、クライアントが強力な認証を必要とする削除などの操作を要求しました。切断の一方的な通知では、LDAP サーバーは、クライアントとサーバー間の通信を保護するために予期しない障害が発生したか、侵害されていることを検出します。
9		予約済み。
10	LDAP_REFERRAL	エラー状態を示すものではありません。LDAPv3 で、サーバーが要求のターゲットエントリを保持していないが、照会フィールド内のサーバーが保持している可能性があることを示します。
11	LDAP_ADMINLIMIT_EXCEEDED	管理権限によって設定された LDAP サーバーの制限を超えたことを示します。
12	LDAP_UNAVAILABLE_CRITICAL_EXTENSION	1 つ以上の重要な拡張が利用できなかったため、LDAP サーバーが要求を満たすことができなかったことを示します。サーバーがコントロールをサポートしていないか、コントロールが操作タイプに適していません。
13	LDAP_CONFIDENTIALITY_REQUIRED	セッションが、セッションの機密性を提供するトランスポートレイヤーセキュリ

標準的な **LDAP** エラー (続く)

エラーデータコード	テキスト	説明
		ティ (TLS) などのプロトコルによって保護されていないことを示します。
14	LDAP_SASL_BIND_IN_PROGRESS	エラー状態を示すのではなく、サーバーがプロセスの次のステップを実行する準備ができていることを示します。クライアントは、プロセスを続行するために同じ SASL メカニズムをサーバーに送信する必要があります。
15		使用されていません。
16	LDAP_NO_SUCH_ATTRIBUTE	変更または比較操作で指定された属性がエントリに存在しないことを示します。
17	LDAP_UNDEFINED_TYPE	変更または追加操作で指定された属性が LDAP サーバーのスキーマに存在しないことを示します。
18	LDAP_INAPPROPRIATE_MATCHING	検索フィルターで指定された一致ルールが属性の構文に定義されたルールと一致しないことを示します。
19	LDAP_CONSTRAINT_VIOLATION	DN の変更、追加、または変更操作で指定された属性値が、属性に課せられた制約に違反していることを示します。制約は、サイズまたはコンテンツ (文字列のみ、バイナリなし) のいずれかです。
20	LDAP_TYPE_OR_VALUE_EXISTS	変更または追加操作で指定された属性値がその属性の値として既に存在することを示します。
21	LDAP_INVALID_SYNTAX	追加、比較、または変更操作で指定された属性値が、属性の認識できない構文または無効な構文であることを示します。
22 〜 31		使用されていません。
32	LDAP_NO_SUCH_OBJECT	ターゲットオブジェクトが見つからないことを示します。このコードは、検索ベースを検索するものの検索フィルターに一致するエントリを見つけられない検索操作や、バインド操作では返されません。
33	LDAP_ALIAS_PROBLEM	エイリアスが逆参照されたときにエラーが発生したことを示します。
34	LDAP_INVALID_DN_SYNTAX	DN の構文が正しくないことを示します (DN 構文が正しいにもか

標準的な **LDAP** エラー (続く)

エラー / データコード	テキスト	説明
		かわらず、LDAP サーバーの構造ルールで操作が許可されていない場合、サーバーはコード 53: LDAP_UNWILLING_TO_PERFORM を返します)。
35	LDAP_IS_LEAF	指定された操作をリーフエントリで実行できないことを示します (このコードは現在 LDAP 仕様に含まれていませんが、この定数用に予約されています)。
36	LDAP_ALIAS_DEREF_PROBLEM	検索操作中に、クライアントにエイリアスオブジェクトの名前を読み込むアクセス権がないか、逆参照が許可されていないことを示します。
37 ～ 47		使用されていません。
48	LDAP_INAPPROPRIATE_AUTH	バインド操作中に、クライアントが正しく使用できない認証方法を使用しようとしていることを示します。たとえば、次のいずれかがこのエラーの原因です。強力な認証情報が必要な場合、クライアントは簡単な認証情報を返します。または、エントリにパスワードが定義されていない場合、クライアントは簡易バインドの DN とパスワードを返します。
49	LDAP_INVALID_CREDENTIALS	バインド操作中に、クライアントが誤った DN またはパスワードを渡した、またはパスワードの有効期限が切れたか、侵入者検出によってアカウントがロックされたか、あるいはその他の同様の理由によりパスワードが正しくなくなったことを示します。詳細については、データコードを参照してください。
49 / 52e	AD_INVALID CREDENTIALS	Active Directory (AD) の AcceptSecurityContext エラーを示します。これは、ユーザー名が有効であるが、パスワードとユーザー認証情報の組み合わせが無効である場合に返されます。これは、LDAP エラーコード 49 に相当する AD です。
49 / 525	USER NOT FOUND	ユーザー名が無効な場合に返される Active Directory (AD) の AcceptSecurityContext データエラーを示します。

標準的な **LDAP** エラー (続く)

エラー / データコード	テキスト	説明
49 / 530	NOT_PERMITTED_TO_LOGON_AT_THIS_TIME	ユーザーが現時点でログオンを許可されていないために発生したログオンの失敗である、Active Directory (AD) の AcceptSecurityContext データエラーを示します。有効なユーザー名と有効なパスワードの認証情報が提示された場合にのみ返されます。
49 / 531	RESTRICTED_TO_SPECIFIC_MACHINES	ユーザーがこのコンピューターからのログオンを許可されていないために発生したログオンの失敗である、Active Directory (AD) の AcceptSecurityContext データエラーを示します。有効なユーザー名と有効なパスワードの認証情報が提示された場合にのみ返されます。
49 / 532	PASSWORD_EXPIRED	ログオンの失敗である Active Directory (AD) の AcceptSecurityContext データを示します。指定されたアカウントのパスワードの有効期限が切れています。有効なユーザー名とパスワードの認証情報が提示された場合にのみ返されます。
49 / 533	ACCOUNT_DISABLED	ログオンの失敗である Active Directory (AD) の AcceptSecurityContext データを示します。アカウントは現在無効です。有効なユーザー名とパスワードの認証情報が提示された場合にのみ返されます。
49 / 568	ERROR_TOO_MANY_CONTEXT_IDS	ログオン試行中に、ユーザーのセキュリティコンテキストで蓄積されたセキュリティ ID が多すぎることを示します。これは、LDAP アドミニストレーターが調査する必要がある特定の LDAP ユーザーオブジェクト/アカウントの問題です。
49 / 701	ACCOUNT_EXPIRED	ログオンの失敗である Active Directory (AD) の AcceptSecurityContext データを示します。ユーザーのアカウントの有効期限が切れています。有効なユーザー名とパスワードの認証情報が提示された場合にのみ返されます。
49 / 773	USER MUST RESET PASSWORD	Active Directory (AD) の AcceptSecurityContext データエラー

標準的な **LDAP** エラー (続く)

エラーデータコード	テキスト	説明
		を示します。最初にログオンする前に、ユーザーのパスワードを変更する必要があります。有効なユーザー名とパスワードの認証情報が提示された場合にのみ返されます。
50	LDAP_INSUFFICIENT_ACCESS	問い合わせユーザーに、要求された操作を実行するための十分な権限がないことを示します。
51	LDAP_BUSY	現時点では LDAP サーバーがビジー状態でクライアント要求を処理できないことを示しますが、クライアントが要求を待機してから再送信した場合、サーバーはそれを処理できる可能性があります。
52	LDAP_UNAVAILABLE	LDAP サーバーがクライアントのバインド要求を処理できないことを示します。通常はシャットダウン中であることが理由です。
52e	AD_INVALID CREDENTIALS	Active Directory (AD) の AcceptSecurityContext エラーを示します。これは、ユーザー名が有効であるが、パスワードとユーザー認証情報の組み合わせが無効である場合に返されます。これは、LDAP エラーコード 49: LDAP_INVALID_CREDENTIALS に相当する AD です。
53	LDAP_UNWILLING_TO_PERFORM	サーバー定義の制限のために LDAP サーバーが要求を処理できないことを示します。このエラーは、エントリの追加要求がサーバーの構造ルールに違反している、ユーザーが変更できない属性が属性の変更要求に指定されている、パスワードの制限によりアクションが実行できない、または接続制限によりアクションが実行できないという理由で返されます。
54	LDAP_LOOP_DETECT	クライアントがエイリアスまたは照会ループを検出したため、この要求を完了できないことを示します。
55 ～ 63		使用されていません。
64	LDAP_NAMING_VIOLATION	DN の追加または変更操作がスキーマの構造ルールに違反していることを示しま

標準的な **LDAP** エラー (続く)

エラーデータコード	テキスト	説明
		す。たとえば、要求はエントリをエイリアスの下に配置します。要求は、格納規則で禁止されているコンテナにエントリを配置します。エントリの RDN で禁止されている属性タイプが使用されています。
65	LDAP_OBJECT_CLASS_VIOLATION	追加、変更、または DN の変更操作がエントリのオブジェクトクラスルールに違反していることを示します。たとえば、追加または変更操作で必須属性の値なしでエントリを追加しようとする要求、追加または変更操作でクラス定義に含まれていない属性の値を使用してエントリを追加しようとする要求、変更操作で必須属性を定義する補助クラスを削除せずに必須属性を削除しようとする要求でこのエラーが返されます。
66	LDAP_NOT_ALLOWED_ON_NONLEAF	要求された操作がリーフエントリでのみ許可されることを示します。たとえば、クライアントによる親エントリでの削除操作の要求、クライアントによる親エントリでの DN の変更操作の要求でことエラーが返されます。
67	LDAP_NOT_ALLOWED_ON_RDN	変更操作で、エントリの相対識別名を形成する属性値を削除しようとしたことを示します。
68	LDAP_ALREADY_EXISTS	追加操作で既に存在するエントリを追加しようとしたか、変更操作で既に存在するエントリの名前に変更しようとしたことを示します。
69	LDAP_NO_OBJECT_CLASS_MODS	変更操作でオブジェクトクラスの構造ルールを変更しようとしたことを示します。
70	LDAP_RESULTS_TOO_LARGE	CLDAP 用に予約されています。
71	LDAP_AFFECTS_MULTIPLE_DSAS	DN の変更操作で LDAP サーバー間でエントリが移動され、複数の LDAP サーバーが必要であることを示します。
72 ～ 79		使用されていません。
80	LDAP_OTHER	不明なエラー状態を示します。これは、他の LDAP エラーコードにマップされ

標準的な **LDAP** エラー (続く)

エラー / データコード	テキスト	説明
		ない NDS エラーコードのデフォルト値です。
775	USER_ACCOUNT_LOCKED	ユーザーアカウントがロックされているため、ユーザーがログインできないことを示します。

カスタマイズ済みエラーコード

カスタマイズ済み **LDAP** エラーコード

エラー / データコード	テキスト
10000	LDAP_ERROR_GENEREL
10001	LDAP_ERROR_MAL_FORMED_URL
10002	LDAP_ERROR_UNAUTHENTICATED_BIND
10300	LDAP_ERROR_COMMUNICATION_EXCEPTION
10301	LDAP_ERROR_SOCKET_TIMEOUT
10302	LDAP_ERROR_CONNECTION_REFUSED
10303	LDAP_ERROR_CONNECTION_RESET
10304	LDAP_ERROR_NO_ROUTE
10305	LDAP_ERROR_UNKNOW_HOST
10400	LDAP_ERROR_SSL_EXCEPTION
10401	LDAP_ERROR_SSL_EMPTY_CERT_STORE
10402	LDAP_ERROR_SSL_CERT_NOT_FOUND
10403	LDAP_ERROR_SSL_CERT_EXPIRED
10500	LDAP_ERROR_INVALID_SEARCH_FILTER_EXCEPTION

LDAP サーバーのダウン時にワンタイムパスワードを送信する

LDAP プロパティは、LDAP サーバーがダウンしているためにユーザーがログインできない場合に、ワンタイムパスワードを送信するために使用できます。別のプロパティを設定して、パスワードの有効期間をコントロールすることもできます。

始める前に
必要なロール：admin
ワンタイムパスワードを受信するには、ユーザープロファイルで通知を有効にする必要があります。通知はメールメッセージのみです。SMS メッセージはサポートされていません。

このタスクについて
どちらのプロパティもデフォルトで有効になっています。パスワードの有効性をコントロールするプロパティのデフォルト値は 10 分です。

手順

1. アプリケーションナビゲーターのフィルターに「sys_properties.list」と入力して、システムプロパティのリストを開きます。
2. *glide.ldap.onetime.password.enabled* プロパティを検索します。
3. プロパティを [true] に設定します。
4. ユーザーのパスワードの有効時間を変更するには、プロパティ *glide.authenticate.onetime.password.validity* を整数の分数に設定します。

LDAP レコードを同期する

アドミニストレーターは、非アクティブ、無効、または削除された LDAP レコードを自身の LDAP レコードと同期できます。

LDAP レコードの同期は、LDAP サーバー上の非アクティブなレコードを検出し、対応する LDAP レコードを更新するプロセスです。非アクティブな LDAP レコードを検出するには、各ユーザーオブジェクトの一貫したデータインジケーターを定義し、LDAP データをインポートし、データインジケーターを評価する必要があります。

データインジケーターは次のようになります。

- [日付] フィールド
- *useraccountcontrol* 属性を使用した、特定の OU のメンバーシップ (*dn* 属性を解析して識別)
- これらのインジケーターの組み合わせ

インポートされたデータは、データを評価および処理できるインポートセットテーブルを介してインスタンスに取り込まれます。

インポートプロセスでは、複数のインポートジョブで LDAP リフレッシュフィルターを使用して、さまざまなタイプのユーザーレコードを分割し、レコードを分離して個別に処理できます。

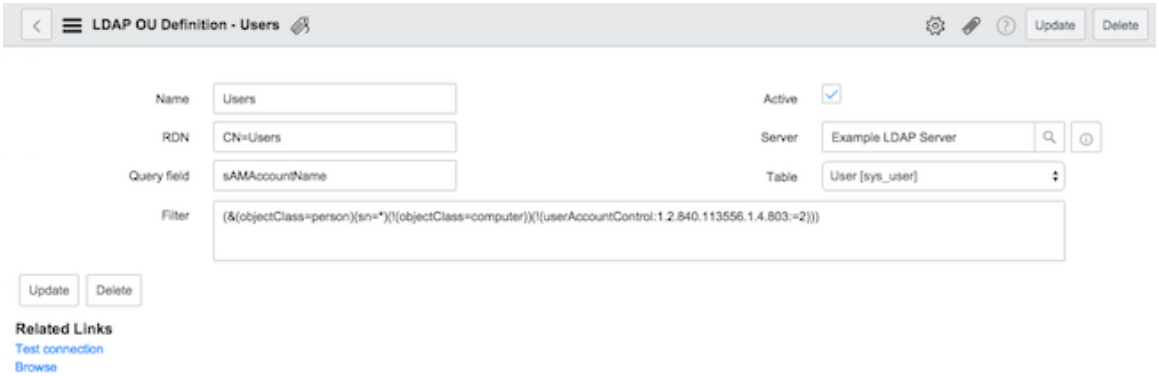
LDAP リフレッシュフィルター

LDAP リフレッシュプロセスのフィルターを使用して、無効なユーザーの挿入を無視する処理を指定できます。

LDAP OU フィルターを緩和して、すべてのデータをインポートセットテーブル (非アクティブなユーザーを含む) に取り込み、無効なユーザーの挿入を無視する処理を指定できます。インスタンスがすぐに利用可能な LDAP サンプルで提供する、サンプルの「ユーザー」OU 定義には、フィルターが含まれています。

このフィルターは、評価対象のインポートセットテーブルに取り込まれるユーザーレコードを定義するため重要です。データのロードは小さくなりますが、このフィルターの制限により、非アクティブなユーザーが除外されるため、非アクティブなユーザーレコードはインポートセットの一時テーブルにインポートされません。非アクティブなユーザーレコードは可視化されないため、レコードインジケーターを評価することはできません。

LDAP OU フィルター



メインの LDAP リフレッシュプロセス内でフィルタリングを使用するには、すべてのユーザーレコードを取り込むようにフィルターを変更します。結果として、すべてのレコードがインポートセットの一時テーブルにロードされ、そこで評価および変換が可能になります。

注: ここで注意が必要です。フィルタリングではすべてのレコードが取り込まれるため、インスタンスに挿入すべきではない古い非アクティブな LDAP アカウントが大量に存在する可能性があります。無効なユーザーに対してユーザーレコードを作成しないでください。

LDAP 抽出

LDAP 抽出プロセスを実装して、無効なユーザーを検出できます。

LDAP ソースからの抽出は、インポートのすべてのレコードに設定できるアクティブフラグを使用して、無効なユーザーをフィルタリングできます。('target.active=false') を指定し、テーブル変換マップレコードの [スクリプト] フィールドに直接コピーします。

利点

この方法のメリットは次のとおりです。

- スクリプティングが簡単
- 既存のユーザーレコードが処理に関与しない
- 非アクティブなユーザーが一時インポートテーブルにロードされない
- パフォーマンスへの影響がない

欠点

この方法のデメリットは次のとおりです。

- 追加のプロセスが作成される
- データソースがアクセスできる場所に抽出セットを配置する必要がある

代替方法

LDAP リフレッシュフィルターでは、複数のインポートジョブを使用して異なるタイプのユーザーレコードを分割し、レコードを分離して個別に処理します。

非アクティブな LDAP ユーザーアカウント

既存の現在のユーザーアカウントが非アクティブであるか、Active Directory (AD) LDAP から削除されていることを検出します。

LDAP 統合の一般的な問題は、Active Directory (AD) で無効化または削除されたユーザーを検出し、インスタンスでそれらを非アクティブ化する方法です。Active Directory LDAP では、通常、

更新時に非アクティブなユーザーを除外するようにフィルターが設定されているため、インスタンスは AD で無効化または削除されたユーザーを認識しません。問題は、既存の現在のユーザーが非アクティブであること、または AD から削除されたことを検出する方法です。

非アクティブなアカウントの検索の詳細については、「userAccountControl フィールドを使用した非アクティブな LDAP アカウントの検索」を参照してください。

注: 推奨されるアプローチは、ユーザーレコードと他のすべてのタイプのレコードを削除することではなく、それらを非アクティブ化にすることです。各レコードは他のレコードにリンクされており、レコードを削除すると他のレコードとの関係がすべて破棄されます。レコードを非アクティブ化すると、これらの関係は維持されます。

userAccountControl フィールドを使用した非アクティブな LDAP アカウントの検索

Active Directory (AD) ユーザーがいつ削除されるか (または非アクティブになるか) を識別します。

始める前に
必要なロール：admin

このタスクについて
1 つの方法は、AD ユーザーのアクティブステータスを追跡し、AD アカウントが非アクティブなときに対応するアカウントを更新するビジネスルールを作成することです。

手順

1. ユーザー [sys_user] テーブルに新しい文字列フィールドを作成して、AD **[userAccountControl]** フィールドの値を追跡します。
   例：u_ad_user_account
2. フィールド値を設定する LDAP 変換スクリプトを作成します。

   Example

   ```
   target.u_ad_user_account = source.userAccountControl
   ```

3. LDAP フィルターを更新して、無効な AD アカウントを表示します。

   Example
   フィルターの例を次に示します。

   ```
   (&(objectClass=person)(sn=*)(!(objectClass=computer))(!
   (userAccountControl:1.2.840.113556.1.4.803:=2)))
   ```

   使用できる交換用フィルターの例を次に示します。

   ```
   (&(objectClass=person)(sn=*)(!(objectClass=computer)))
   ```

4. [u_ad_user_account] フィールドの値が 514 の場合は常に active フィールドを false に設定する onChange ビジネスルールを作成します。
   「514」は非アクティブなアカウントを示します。

LDAP スクリプトの例

次のスクリプトの例では、LDAP サーバーに Active Directory (AD) を使用していることを前提としています。

userAccountControl 属性値スクリプト

この例では、無効なユーザー (514 または 546) に関連付けられた userAccountControl 属性値をソースでテストします。

```
//Deactivate LDAP-disabled users during transform based on 'userAccountControl' attribute
if(source.u_useraccountcontrol == '514' || source.u_useraccountcontrol == '546'){
  target.active=false;
  target.locked_out=true;
}
```

ビットごとのチェックを使用した例を次に示します。

```
if(source.u_useraccountcontrol & 2){
  active = false;
}
```

userAccountControl 属性スクリプト

この例では、userAccountControl 属性を調べますが、特定の値をテストしません。LDAP ユーザーアカウントを再有効化するオプションも含まれています。

```
/*
* Deactivate LDAP-disabled users during transform based on 'userAccountControl' attribute
* Convert the userAccountControl attribute back to a hex value
*/
var ctrl = parseInt(source.u_useraccountcontrol, 10);
ctrl = ctrl.toString(16);

/*
* The only digit we care about is the final one
* A final hex digit value of '2' in 'ctrl' means disabled
*/
if(ctrl.substr(-1) == "2"){

  //Deactivate and lock the user account
  target.active = false;
  target.locked_out = true;

  //Ignore any insert of a disabled record
  if(action == 'insert'){
    ignore = true;
  }
}
/* Optional: Uncomment else block to reactivate and unlock the user account
else {
  target.active = true;
  target.locked_out = ctrl.substr(-2, 1) == "1";
}
*/
```

onBefore 変換マップスクリプト

onBefore 変換マップスクリプトの例を次に示します。スクリプトは、無効なレコードと挿入されるレコードを識別します。無効なユーザーの挿入が発生している場合、操作の変換はレコードを無視します。

```
//Ignore any insert of a disabled record as defined by the 'userAccountControl' attribute
var uc = source.u_useraccountcontrol;
if((uc == '514' || uc == '546') && action == 'insert'){
  ignore = true;
}
```

DN メンバースクリプト

このスクリプトの例では、546 および 514 userAccountControl 値に依存せずに、ユーザーが特定の識別名 (DN) のメンバーであるかどうかをチェックすることで、柔軟性を導入しています。このスクリプトは、「テーブル変換マップ」レコードの [スクリプト] フィールドまたは onBefore 変換マップスクリプトで使用できます。

```
//Deactivate LDAP-disabled users during transform based on OU membership in 'dn'
if(source.u_dn.indexOf('OU=Disabled Accounts') > -1){
  target.active = false;
  target.locked_out = true;
}
```

Active Directory Application Mode (ADAM)

Active Directory Application Mode (ADAM) は、Lightweight Directory Access Protocol (LDAP) 準拠のディレクトリサービスです。

注: このトピックを理解するには、Microsoft Windows サーバーと Active Directory の基本を理解する必要があります。ADAM 用に構成するサーバーの"アドミニストレーター権限も必要です。

これらはサンプル手順です。インストールと環境はさまざまであるため、直接サポートを提供することはできません。Microsoft コンサルタントに相談することをお勧めします。

ADAM は、Windows オペレーティングシステムで簡単にインストールでき、サービスとして実行されます。完全にカスタマイズしてアプリケーションコンポーネントとして配布することも、スタンドアロンの LDAP ディレクトリとして使用することもできます。ADAM は、Active Directory ドメインコントローラーと同じテクノロジー (レプリケーション機能と委任機能を含む) を使用し、独自の管理機能とカスタマイズ機能を備えています。Windows サービスとして実行できます。ADAM は、Windows XP、2000、2003、および 2008 のオペレーティングシステムにインストールできます。ADAM は、Windows Server 2003 R2 および Windows Server 2008 の一部として含まれています。以前のオペレーティングシステムについては、http://www.microsoft.com/downloads http://www.microsoft.com/downloads からダウンロードできます。

セキュリティ

会社のセキュリティポリシーによっては、外部ベンダーやパートナーが Active Directory (AD) ドメインコントローラに直接接続することが禁止されています。特定の AD オブジェクトまたは属性を外部ベンダーまたはパートナーに公開することが禁止されている場合は、AD セキュリティアクセス制御エントリ (ACE または ACL) を使用してオブジェクトおよび属性へのアクセスをブロックできます。セキュリティ要件に応じて、この方法では統合が複雑になる可能性があります。複数のドメインとフォレストを統合することをお勧めします。すべての LDAP インポートと認証を単一のソース経由で行う必要がある場合は、統合ソースとして ADAM を使用できます。Windows 2008 のリリースで、この機能は Light-Weight-Directory Service (LDS) に名前が変更されました。インストールと構成は Windows Server 2003 R2 と同様です。

推奨ナレッジ

このタスクでは、AD、オブジェクトクラス、および属性を理解する必要があります。統合を正常に行うには、現在の AD オブジェクト構造を理解し、Active Directory の委任に精通し、ADAM を使用する方法と目的について戦略を立てる必要があります。AD または ADAM に精通していない場合は、AD アドミニストレーターと協力して新しい ADAM 環境を構成してください。

信託

userProxy オブジェクトを使用する場合、ADAM をホストするコンピューターは、AD アカウントを持つドメインのメンバーであるか、信頼できるドメインのメンバーである必要があります。

内部接続

`userProxy` オブジェクトを使用する場合、ADAM コンピューターは関連するドメインコントローラに接続してプロキシ認証を実行できる必要があります。

関連トピック

http://www.microsoft.com/downloads/en/details.aspx?familyid=9688f8b9-1034-4ef6-a3e5-2a2a57b5c8e4&displaylang=en%7C

ADAM を使用してインスタンスを構成する

初回インストールでは、ADAM ファイルがコンピューターにコピーされ、必要なコンポーネントが登録され、アプリケーションのショートカットが作成されます。

始める前に
必要なロール：admin

このタスクについて
デフォルトでは、アプリケーションファイルはすべて %systemroot%\ADAM にインストールされます。

- Windows Server 2003 R2 - ADAM は、 コントロールパネル > プログラムの追加と削除 > オプションのコンポーネントマネージャー.
- Windows Server 2000 および Windows XP - Microsoft から http://www.microsoft.com/downloads をダウンロードします。

ADAM によってホストされる最初のディレクトリサービスとして機能する最初のインスタンスサービスを作成します。次のいずれかの操作を行います。

手順

- ADAM フォルダーから `adaminstall.exe` を実行します。
- から **ADAM** インスタンスを作成 ショートカットを使用する **ADAM** >プログラム>スタートメニュー フォルダ。

1. [一意のインスタンス] インストールオプションを選択します。

   注: このオプションを使用して、2 番目のサーバーにインスタンスのレプリカをインストールして、フォールトトレラントなシステムを提供できます。

2. フィールドに入力します。

ADAM インスタンス

フィールド	説明
インスタンス名	主に Windows サービス名と表示名を識別するために使用します。
ポート	LDAP および LDAPS リスナーに使用するポート番号を設定します。デフォルトの LDAP ポートは 389、LDAPS は 636 です。これらのポートがサーバーで使用されている場合、セットアップウィザードは新しいポートを選択します。ネットワークアドミニストレーターと協力して、使用するのに最適なポートを決定してください
アプリケー	アプリケーションディレクトリパーティションを作成します。この段階では必要ありませんが、ここで新しいパーティションを作成することをお勧めします。フォレス

フィールド	説明
ションディレクトリパーティション	トまたはドメインと同じ識別名を使用することをお勧めしますが、最上位レベルのドメインを com または local ではなく adam に置き換えます。たとえば、フォレストパーティションが *dc=myCompany,dc=com* の場合、ADAM パーティションを *dc=myCompany,dc=adam* として作成します。
ファイルの場所	ADAM パーティションデータの場所を選択します。
サービスアカウントの選択	インスタンスを実行するサービスアカウントを選択します。スタンドアロンサービスの場合は、デフォルトのネットワークサービスアカウントを使用できます。レプリカを使用する場合は、すべての ADAM インスタンスにアクセスできるアカウントを使用する必要があります。
ADAM アドミン	Windows 統合認証を利用する ADAM ディレクトリの委任。管理用の初期アクセス権はこの方法で付与されます。最初のアカウントに権限が付与されると、このユーザーまたはグループは他の Windows ユーザーまたは ADAM ユーザーに権限を委任します。デフォルトを選択して、現在のユーザーにのみ admin アクセス権を付与することも、必要に応じて別のユーザーまたはグループにアクセス権を付与することもできます。
LDIF ファイルのインポート	インポートするファイル。MS-UserProxy はインポートする最も重要なファイルですが、スキーマにほとんどオーバーヘッドがなく、後でニーズが拡大した場合に拡張する必要がないため、利用可能なすべてのファイルを追加する価値があります。詳細を確認すると、ウィザードが設定を完了します。

関連トピック

http://www.microsoft.com/downloads/en/details.aspx?familyid=9688f8b9-1034-4ef6-a3e5-2a2a57b5c8e4&displaylang=en%7C

ADAM コンソールの設定

ADAM コンソールを設定します。ADAM と Active Directory の間には多くの類似点がありますが、[ユーザーとコンピューター] の管理コンソールがないため、管理は大きく異なります。

始める前に

必要なロール：admin

このタスクについて

一般的な管理のほとんどは、**[ADAM]** スタートメニューから利用可能な ADAM ADSI MG コンソールを使用して行われます。ADAM ADSI コンソールを初めて実行するときは、作成したパーティションに接続する必要があります。

手順

1. 左フレームの **[ADAM ADSI Edit]** アイテムを右クリックします。
2. 新しい接続に名前を付け、インスタンスの作成時に使用した情報でサーバー名とポートのフィールドを更新します。
3. [識別名] または [命名コンテキスト **(naming context)]** を選択し、以前に作成したアプリケーションパーティションの識別名を指定します。

詳細設定オプションの [構成およびスキーマ *(Configuration and Schema)*] パーティションに接続できます。
これで、LostAndFound、NTDS クォータ、およびロールのパーティションとデフォルトのコンテナが表示されます。ロールコンテナがまだ構成されていません。

関連トピック

http://www.microsoft.com/downloads/en/details.aspx?familyid=9688f8b9-1034-4ef6-a3e5-2a2a57b5c8e4&displaylang=en%7C

ADAM のコンテナと組織単位の作成

ADAM に格納されているオブジェクトを、Active Directory の場合と同様に、コンテナと組織単位 (OU) に論理的にグループ化します。

始める前に
必要なロール：admin

手順

1. ルートパーティションを右クリックし、 新規 > オブジェクト > **organizationalUnit**.

   注: 利用可能な他のオブジェクトのリストを表示することもできます。このリストは、LDF ファイルをインポートしたときにインストールされたスキーマ拡張によって異なります。

2. 値の入力を求められたら、Users などの OU の名前を入力します。
   画面に [その他の属性] ボタンが表示されます。
3. ボタンを使用して、追加の属性に値を割り当てます。
   OU とコンテナの場合、追加の値は必要ありません。
   OU を作成すると、新しい OU がルートオブジェクトの子としてリストされます。

関連トピック

http://www.microsoft.com/downloads/en/details.aspx?familyid=9688f8b9-1034-4ef6-a3e5-2a2a57b5c8e4&displaylang=en%7C

ADAM を使用して委任する

OU 構造が作成されたら、限られたユーザーにオブジェクトを適切に保護するための権限委任を定義します。

Active Directory と同様に、権限を付与する一般的な方法は 2 つあります。

- 適切な権限が既に割り当てられているグループにユーザーを追加します。
- ADAM オブジェクトに対する新しい権限を定義します。

このタスクでは、オブジェクトレベルの権限について説明します。グループメンバーシップの詳細については、「グループ管理 (Group Administration)」セクションを参照してください。

ADAM 用の [ユーザーとコンピューター] コンソールがないため、オブジェクトレベルの権限はすべて Active Directory ユーティリティ *DSACLS.exe* を使用して定義されます。このファイルは ADAM プログラムディレクトリにあります。ADAM ユーティリティを実行する場合は、ADAM ツールのコマンドプロンプトを起動することをお勧めします。これにより、ツールのバージョンが適切になります。DSALCS は、オブジェクトのアクセス権を表示および設定するために使用されます。

例：「dsacls \\localhost:50010\dc=myCompany,dc=adam」は、localhost ポート 50010 で実行されているパーティション *dc=myCompany, dc=adam* のルートに割り当てられた権限を表示しま

す。DSACLS は、複雑な委任を作成するために使用される複雑なツールです。使用上の注意については「DSACLS /?」を実行します。

関連トピック

ADAM のコンテナと組織単位の作成

ADAMSync を使用して ADAM に入力する

http://www.microsoft.com/downloads/en/details.aspx?familyid=9688f8b9-1034-4ef6-a3e5-2a2a57b5c8e4&displaylang=en%7C

ADAM オブジェクトを設定する

ADAM オブジェクトには、ユーザーオブジェクト、UserProxy オブジェクト、およびグループオブジェクトが含まれます。

ユーザーオブジェクト

ユーザーは、OU の作成と同様に ADAM ADSI Edit コンソールを使用して作成できます。AD コマンドラインツールを使用してユーザーを管理することもできますが、これについてはこのドキュメントでは説明しません。新しいユーザーオブジェクトに必須の属性は *cn* のみです。これはユーザーの短い名前またはフルネームです。Active Directory ユーザー属性と同様の幅広いオプション属性もあります。ユーザーオブジェクトからプロパティを選択することで、属性の完全なリストにアクセスできます。

UserProxy オブジェクト

ServiceNow LDAP 統合では、関連する AD ユーザーアカウントにリンクするプロキシアカウントを作成する ADAM の *UserProxy* オブジェクトを使用することをお勧めします。これにより、ServiceNow がドメインコントローラに直接接続しなくても、ドメインの AD ユーザー名とパスワードを使用して ADAM でログオン認証情報を認証できます。*UserProxy* オブジェクトは、パスワードを格納せず、リンクされた AD ユーザーオブジェクトの SID を含む *objectSID* 属性を持つ点を除いて、AD および ADAM ユーザーオブジェクトとよく似ています。プロキシは以上のように動作します。*UserProxy* オブジェクトは *ADSIEdit* コンソールまたはコマンドラインツールを使用して作成されますが、この作業には手間がかかります。以下で定義されている自動化プロセスを使用することをお勧めします。

グループオブジェクト

グループは、ADSIEdit コンソールと AD コマンドラインツールを使用して作成されます。グループの概念は AD に似ており、グループとメンバーを ServiceNow に統合するために使用されます。最大の違いは、ADAM グループには ADAM または信頼できる AD ドメインのメンバーを含めることができるということです。

ADAM オブジェクトの作成を自動化する

Active Directory アカウントを ADAM に同期する場合は、 Microsoft *ADAMSync* 移動しますこれは、ServiceNow LDAP 統合のための ADAM の最も一般的な使用法です。

権限の委任について

ADAM には、デフォルトの権限を持つビルトイングループがいくつか含まれています。これらのグループは、*cn=roles,dc=myCompany,dc=adam* コンテナにあります。これらはドメインレベルグループに似ており、現在のパーティション内のオブジェクトに対する権限があります。AD フォレストと同様に、*cn=roles,cn=configuration,dc=myCompany,dc=adam* のデフォルトグループを使用して、より高いレベルの権限を設定することもできます。*ADSIEdit* の構成パーティションに接続する必要があります。デフォルトでは、アドミングループにはセットアップ中に指定されたアカウントが含まれています。このメンバーは構成グループを介して継承されるため、常に表示されるわけ

自動翻訳

ではありません。アドミニストレーターは、すべてのパーティションオブジェクトを完全に制御できます。デフォルトでは、リーダーグループにはメンバーが含まれておらず、パーティション内のすべてのオブジェクトに対する読み取りアクセス権があります。ユーザーグループは、Active Directory と同様に動的なグループです。推移的には、パーティションに作成されたすべての ADAM ユーザーが含まれます。

関連トピック

http://www.microsoft.com/downloads/en/details.aspx?familyid=9688f8b9-1034-4ef6-a3e5-2a2a57b5c8e4&displaylang=en%7C

ADAM セットアップをテストしてトラブルシューティングする

テストに使用される主要なツールは LDP です。これにより、ユーザー認証を完全にテストできます。

オブジェクトと属性のコレクション全体へのアクセスを提供する ADAM ADSI Edit コンソールを使用して、オブジェクト管理のほとんどを実行できます。ADAM サービスの最高レベルの制御とトラブルシューティングでは、インスタンスのセットアップ中に作成された Windows サービスを使用します。サービス名は作成されたインスタンスの名前によって異なります。ADAM サービスを実行するには、このサービスが実行されている必要があります。接続の問題が発生している場合は、ネットワーク構成を確認して、サーバーと ADAM ポートに接続するための適切なネットワークアクセス権があることを確認する必要があります。インストールされている ADAM インスタンスごとに、Windows イベントログが作成されます。これは、ADAM サービスのトラブルシューティングにも最適なツールです。

Windows セキュリティイベントログは、*userProxy* 認証のトラブルシューティングにも役立ちます。すべての *userProxy* ログオン試行はセキュリティログに記録され、リモートクライアントデバイスのアドレス、ログオンしようとしているユーザーの識別名、および結果またはステータスコードを参照します。

関連トピック

http://www.microsoft.com/downloads/en/details.aspx?familyid=9688f8b9-1034-4ef6-a3e5-2a2a57b5c8e4&displaylang=en%7C

ADAM でバックアップして復旧する

すべての ADAM データは、標準のファイルシステムバックアップ方法を使用してバックアップできます。

冗長性

ADAM には、AD と同じテクノロジーに基づくレプリケーションユーティリティが組み込まれています。ADAM パーティションの完全な読み取りおよび書き込みレプリカは、同じまたは別のコンピューターに存在できます。このレプリカをさまざまな方法で使用して、インスタンスとのフォールトトレラントな LDAP 統合を提供できます。1 つは、両方のパーティションをファイアウォール経由でインスタンスに公開し、[LDAP プロパティサーバー (LDAP Properties server)] フィールドの両方のサーバーを定義するという方法です。

関連トピック

Active Directory Application Mode (ADAM)

http://www.microsoft.com/downloads/en/details.aspx?familyid=9688f8b9-1034-4ef6-a3e5-2a2a57b5c8e4&displaylang=en%7C

ADAM で LDAPS を使用する

userProxy オブジェクト認証のデフォルト構成では、LDAPS (セキュア LDAP) 通信を強制します。LDAPS では、ネットワークトラフィックを保護するために SSL 証明書が必要です。

この要件を削除するには、構成パーティションに接続されている *ADSIEdit* コンソールを使用して次の変更を行います。

```
Object: CN=Directory Service, CN=Windows NT, CN=Services, CN=Configuration
Attribute: msDS-Other-Setings
Value: change RequiresSecureProxyBind from 1 (enforced) to 0 (disabled)
```

新しい設定を使用するには、ADAM サービスを再起動します。

セキュアなバインドをサポートし、送信されるユーザーとパスワードの情報を暗号化するには、サーバーと LDAP クライアントに SSL 証明書をインストールする必要があります。ADAM サービスには制限付きで制御された使用法があるため、証明書のコストや認証局 (CA) インフラストラクチャをビルドすることなく、ニーズを満たす自己署名証明書を使用することができます。既に CA がある場合は、証明書を発行できます。ない場合は、自己署名証明書を作成します。

自己署名証明書を作成する

selfssl ユーティリティを使用するには、Internet Information Services (IIS) をインストールする必要があります。このサービスは、証明書を生成した後に削除できます。selfssl.exe ユーティリティは IIS Resource Kit から入手できます。IIS が既にインストールされている場合は、証明書の生成中に現在のサイトが影響を受けないように新しい Web サイトを作成します。selfssl は、新しい自己発行の証明書を有効な Web サイトに一時的に添付する必要があります。

selfssl はコマンドラインツールであり、次の一般的なパラメーターがあります。

selfssl パラメーターの説明

パラメーター	説明
/T	ローカルマシンの「信頼できる証明書」に証明書を追加します
/N:cn	証明書の共通名を設定します。これは、証明書を使用する Web サービスを実行しているサーバーの完全修飾ドメイン名と一致する必要があります
/K	キーサイズの強度をビット単位で設定します
/V	証明書が有効な日数
/S	証明書を添付する Web サイト ID
/P	Web サービスの IP ポート

共通名属性は、インスタンスが ADAM コンピューターに接続するために使用する外部名またはアドレスと一致する必要があります。デフォルトの Web サイト ID である 1 を使用し、selfssl コマンドで定義する必要がない場合を除き、IIS Web サイトのサイト ID を取得する必要があります。myCompany の証明書を生成するサンプルコマンドは次のとおりです。

```
selfssl /N:CN=myCompany.externaldomain.com /K:1024 /V:3650 /S:12345 /P:50001 /T
```

このステートメントは、10 年間有効な証明書を作成します。値を任意の期間に設定します。ただし、古い証明書が期限切れになる前に新しい証明書を生成してインスタンスに送信する必要があることに注意してください。証明書の有効期限をメモすることをお勧めします。

証明書が生成されたら、Web サイトから削除するか、一時サイトを作成した場合は Web サイト全体を削除できます。

関連トピック

http://www.microsoft.com/downloads/en/details.aspx?familyid=9688f8b9-1034-4ef6-a3e5-2a2a57b5c8e4&displaylang=en%7C

ADAM への証明書の割り当て

セキュアなバインドをサポートし、送信されるユーザーとパスワードの情報を暗号化するには、サーバーと LDAP クライアントに SSL 証明書をインストールします。

始める前に
必要なロール：admin

このタスクについて
ADAM サービスには制限付きで制御された使用法があるため、証明書のコストや認証局 (CA) インフラストラクチャをビルドすることなく、ニーズを満たす自己署名証明書を使用することができます。

手順

1. 証明書 MM コンソールを開き、2 つのコンソール接続を作成します。1 つはローカルコンピューター証明書用で、もう 1 つは新しい ADAM サービスのローカルコンピューターサービス証明書用です。
   新しい証明書は、Certificates (ローカルコンピューター)\Personal\Certificates にあります
2. ADAM サービスのコンテナ Certificates – Service (ADAM Service Name)\ADAM_ADAM Service Name\Trusted Root Certificates\Certificates に証明書をコピーし、 Certificates – Service (ADAM Service Name)\ADAM_ADAM Service Name\Personal\Certificates に証明書をコピーします。
3. コピーした証明書の [詳細] タブを開き、有効開始日の日付スタンプをメモして、証明書キーファイルへの読み取りアクセス権を割り当てます。
   C:\Documents and Settings\All Users\Application Data\Microsoft\Crypto\RSA\MachineKeys に移動し、一致するタイムスタンプで証明書を特定します。ADAM を実行しているサービスアカウントに読み取りおよび実行の権限を割り当てます。デフォルトでは、これは [ネットワークサービス] です。
4. 新しい証明書を有効にするには、ADAM サービスを再起動します。

関連トピック

http://www.microsoft.com/downloads/en/details.aspx?familyid=9688f8b9-1034-4ef6-a3e5-2a2a57b5c8e4&displaylang=en%7C

公開鍵証明書のエクスポート

インスタンスを含む LDAPS クライアントには、ADAM への安全な接続を行うために公開鍵証明書が必要です。

始める前に
必要なロール：admin

このタスクについて
上記で使用したサーバー証明書コンソールから、クライアントが使用する公開鍵をエクスポートします。

手順

1. 証明書を選択し、右クリックして [すべてのタスク/エクスポート **(all tasks/export)]** を選択します。
   秘密鍵をエクスポートしないでください。デフォルトの DER エンコードバイナリ X.509 形式を選択し、エクスポートファイル名を指定します。
2. LDAPS を使用してサーバーに接続する LDAP クライアントに公開証明書をインストールします。
   プロンプトが表示されたら、*Trusted Root Certificate Authorities* ストアに証明書を追加します。

関連トピック

http://www.microsoft.com/downloads/en/details.aspx?familyid=9688f8b9-1034-4ef6-a3e5-2a2a57b5c8e4&displaylang=en%7C

Active Directory Application Mode (ADAM) アクセスアカウント

システムには、アプリケーションインスタンスにインポートされた Active Directory Application Mode (ADAM) オブジェクト情報を読み込むためのユーザーアカウントが必要です。

次の方法のいずれかを使用してアカウントを作成します。

- ローカル ADAM ユーザーアカウントを作成し、それにパスワードを割り当て、権限を割り当てます。
- ADAM パーティションの Windows ドメインアカウントに権限を割り当てます。
- *userProxy* アカウントを使用します。

ADAM を LDAP ソースとして使用する場合は、インスタンスの LDAP サーバーの [ログイン識別名] フィールドで ADAM アカウントの完全修飾識別名 (FQDN) を指定する必要があります。

関連トピック

Active Directory Application Mode (ADAM)

http://www.microsoft.com/downloads/en/details.aspx?familyid=9688f8b9-1034-4ef6-a3e5-2a2a57b5c8e4&displaylang=en%7C

LDAPS 接続のテスト

LDAPS 接続をテストします。2 つのコンソール接続があります。1 つはローカルコンピューター証明書用で、もう 1 つは新しい ADAM サービスのローカルコンピューターサービス証明書用です。

始める前に
必要なロール：admin

手順

1. *LDP.exe* を ADAM インストールフォルダー c:\windows\adam から実行します。
   これは標準の Windows LDP クライアントではないため、ADAM バージョンが選択されていることを確認します。
2. [接続/接続する **(Connection/Connect)]** メニューを使用して、新しい接続を開きます。
   サーバー名は、証明書に割り当てられた CN と一致する必要があります。
3. **[LDAPS** ポート **(LDAPS port)]** を入力し、**[SSL]** チェックボックスをオンにします。
   接続に成功すると、一般的なサーバー情報が表示され、エラーはありません。
4. サービスにバインド (ログイン) します。

一般的な LDAP クライアント接続をレプリケートするには、[シンプルバインド] オプションを選択します。[ユーザー] フィールドに有効な ADAM ユーザーまたは *userProxy* 識別名と関連するパスワードを入力します。
「次のように認証されました：…。(Authenticated as:….)」というメッセージが表示された場合は、LDAPS を使用して正常に接続しています。

関連トピック

http://www.microsoft.com/downloads/en/details.aspx?familyid=9688f8b9-1034-4ef6-a3e5-2a2a57b5c8e4&displaylang=en%7C

ADAMSync を使用して **ADAM** に入力する

アドミニストレーターは MS ADAMSync を使用して、MS ADAM を使用する LDAP ディレクトリを設定します。

注:

このドキュメントは、Microsoft Windows Server、Active Directory、および ADAM について少なくとも基本レベルの理解があり、パーティションを備えた機能的な Active Directory Application Mode (ADAM) インスタンスを既に持っていることを前提としています。

これらはサンプル手順です。複雑であり、お使いの環境で実行されているため、直接サポートを提供することはできません。問題が発生した場合は、Microsoft または Microsoft のコンサルタントに相談することをお勧めします。

ADAM がインストールされ、最初のパーティションが作成されたら、そのパーティションにオブジェクトを設定できます。

次のオプションが使用可能です。

- GUI またはスクリプトを使用して手動でオブジェクトを作成します。このオプションは非効率的で時間がかかります。
- Microsoft 統合情報サーバーを使用して Active Directory と統合します。このオプションは、最終的に最も柔軟性と機能性を備えていますが、いくつかの高度な設定が必要です。Active Directory、ADAM、および Exchange の Microsoft グローバルアドレス一覧と互換性のある MIIS の無料バージョンがあります。MIIS を使用した経験がある場合を除き、LDAP 統合専用の新しい環境を実装しないことをお勧めします。
- Microsoft が ADAM に提供する同期ツールである Adamsync を使用します。このオプションについてはここで説明します。

ADAM ユーザーアカウントの定義

ADAM でユーザーアカウントを定義します。ユーザーアカウントの 1 つは接続するインスタンスに使用され、もう 1 つのユーザーアカウントは ADAMSync に使用されます。

始める前に
必要なロール：admin

このタスクについて
これらのアカウントは、ローカルの ADAM ユーザーオブジェクト、UserProxy オブジェクト、または信頼できるドメインの Windows アカウントにすることができます。

ADAM User アカウントには、インスタンスにインポートするディレクトリ構造への読み取り専用アクセスが必要です。これを実現する最善の方法は、cn=roles,dc=myCompany,dc=adam にあるリーダーグループのメンバー属性にアカウントを追加することです。

新しい ADAM ユーザーアカウントはデフォルトで無効になっています。新しいアカウントを有効にしてパスワードを設定する必要があります。

手順

1. ユーザーを有効にするには、属性 msDS-UserAccountDisabled を FALSE に変更します。
2. ユーザーオブジェクトを右クリックし、パスワードをリセットします。
3. Active Directory Application Mode (ADAM) で定義されている LDP を使用して新しいアカウントをテストし、接続できることを確認します。
   次を使用： **LDAP** > ビュー/ツリー オプションを選択し、[ベース DN] を空白のままにして、新しいアカウントを使用してディレクトリ内のオブジェクトを表示できるようにします。[構成]、[スキーマ]、および [ドメイン] パーティションが左ペインに表示されます。[ドメイン] パーティションを走査します。新しいローカル ADAM アカウントを使用している場合は、「子なし (No Children)」と表示されます。これは、オブジェクトへの読み取りアクセス権がないことを意味します。セットアップグループメンバーシップを確認し、再テストします。

   ADAMSync は *ADAMSync User* アカウントを使用して ADAM パーティション内のオブジェクトを管理します。このアカウントは ADAM オブジェクトを作成、更新、削除するため、admin レベルの権限が必要です。

   ADAMSync は *ADAMSync AD* アカウントを使用して、ADAM に同期される AD オブジェクトを読み取ります。

ADAMSync の設定

ADAMSync は Windows Server 2003 R2 に含まれています。別の OS を使用している場合は、ADAMSync をダウンロードしてインストールします。

スキーマを拡張する

ADAMSync をサポートするには、ADAM スキーマを拡張する必要があります。

1. c:\windows\adam から次のコマンドを実行して、ADAMSync スキーマ拡張をインポートします。現在のユーザーにアクセス権がない場合は、 server:port を変更して認証情報を追加する必要があります。詳細については、AdamSyncMetadata.ldf ファイルを参照してください。

```
ldifde -i -f MS-AdamSyncMetadata.LDF -s localhost:50000 -j . -c "cn=Configuration,dc=X"
 #configurationNamingContext
```

2. Windows 2003 の属性をサポートするには、MS-AdamSchemaW2k3.ldf で同じ操作を行います。

```
ldifde -i -u -f MS-AdamSchemaW2K3.LDF -s localhost:50000 -j . -c "cn=Configuration,dc=X"
 #configurationNamingContext
```

推奨されるスキーマの変更

推奨される追加のスキーマ変更を以下に示します。

1. 新しい MMC コンソールを開き、ADAM スキーマスナップインを追加します。
2. ADAM インスタンスに接続します。
3. [クラス] フォルダーを展開し、 [userProxy] クラスを見つけて [プロパティ] を開きます。
4. [属性] タブで次のオプションの属性を確認し、まだ存在しないものを追加します。
   - company
   - department
   - givenNane
   - mail

- physicalDeliveryOfficeName
- sAMAccountName
- sn
- telephoneNumber
- title
- userAccountControl
- userPrincipalName

5. ADAM サービスを再起動して、新しい設定を有効にします。

ADAM 構成ファイルのインストール

Windows のコマンドラインから ADAM 構成ファイルをインストールします。

始める前に
必要なロール：admin

手順

1. 構成ファイルをインストールします。

```
C:\WINDOWS\adam>adamsync /install localhost:50000 MS-AdamSyncConf-SNC.XML
```

2. 同期ファイルを実行してコンソールにログを記録します。

```
C:\WINDOWS\adam>adamsync /sync localhost:50000
"ou=users,dc=service-now,dc=adam" /log -
```

3. ADSIEdit コンソールを使用して結果を確認します。
ADAMSync によって作成された新しいオブジェクトと属性が表示されます。
4. ldap を実行して UserProxy 認証をテストします。
同期プロセスの自動化

同期プロセスを Windows スケジュール設定済みタスクとして設定します。構成ファイルまたはコマンドラインで認証情報を入力するか、アクセスできるアカウントでスケジュール設定済みタスクを実行する必要があります。

特記事項

- 複数の構成ファイルとスケジュール済みジョブを作成して、複数のソースから ADAM を同期できます。

  この例では、アプリケーションログオンとして使用できる sAMAccountName 属性をインポートします。ソースを同期する場合は、ログオン認証情報に使用できる一意の属性値があることを確認する必要があります。sAMAccountName は、複数のドメイン間ではなく、ドメイン内で一意であることが保証されています。

- Microsoft Exchange を使用している場合は、オブジェクトフィルター構成の一部として cn=SystemMailbox* オブジェクトを除外することをお勧めします。

ADAM 構成ファイルの例

ADAMSync の構成はすべて xml ファイルに格納されます。

コメントを含むデフォルトの構成ファイル

ADAMSync のインストールには、 MS-AdamSyncConf.xml と呼ばれるデフォルトの構成ファイルが含まれています。このファイルのコピーを作成して、後で参照する基本の例を用意してください。この例は、コメントが追加されたデフォルトの構成ファイルです。

```
<?xml version="1.0"?>
<doc>
 <configuration>
<!-- Sync File Description -->
<description>MyCompany ADAMSync Configuration</description>
  <security-mode>object</security-mode>;
<!-- source-ad-name = fqdn of the domain controller -->;
  <source-ad-name>;fully.qualified.domain.name.of.domain.controller</source-ad-name>;
<!-- source-ad-partition = root AD domain partition -->;
  <source-ad-partition>;dc=myCompany,dc=com</source-ad-partition>;
<!-- source-ad-account = use this to specify an account to connect to AD -->;
<!-- if not used, the current user will be used  -->;
  <source-ad-account>;</source-ad-account>;
  <account-domain>;</account-domain>;
<!-- target-dn = target ADAM OU -->;
  <target-dn>;ou=servicenow users,dc=myCompany,dc=adam</target-dn>;
  <query>;
<!-- base-dn = should be the root AD partition if you want all users -->;
   <base-dn>;dc=myCompany,dc=com</base-dn>;
<!-- object-filter = standard ldap query format, this will grab all users -->;
<!-- need to review results to see if you should modify this filter -->;
   <object-filter>;(objectCategory=person)</object-filter>;
   <attributes>;
<!-- include=userproxy requires objectSID to link back to the AD account -->;
    <include>;objectSID</include>;
    <include>;givenName</include>;
    <include>;sn</include>;
    <include>;description</include>;
    <include>;title</include>;
    <include>;company</include>;
    <include>;department</include>;
    <include>;mail</include>;
    <include>;physicalDeliveryOfficeName</include>;
    <include>;telephoneNumber</include>;
    <include>;sAMAccountName</include>;
   </attributes>;
  </query>;
<!-- map for user-to-userproxy object types -->;
  <user-proxy>;
    <source-object-class>;user</source-object-class>;
    <target-object-class>;userProxy</target-object-class>;
  </user-proxy>;
  <schedule>;
   <aging>;
    <frequency>;0</frequency>;
    <num-objects>;0</num-objects>;
   </aging>;
   <schtasks-cmd>;</schtasks-cmd>;
  </schedule>;
 </configuration>;
 <synchronizer-state>;
```

自動翻訳

```
 <dirsync-cookie>;</dirsync-cookie>;
 <status>;</status>;
 <authoritative-adam-instance>;</authoritative-adam-instance>;
 <configuration-file-guid>;</configuration-file-guid>;
 <last-sync-attempt-time>;</last-sync-attempt-time>;
 <last-sync-success-time>;</last-sync-success-time>;
 <last-sync-error-time>;</last-sync-error-time>;
 <last-sync-error-string>;</last-sync-error-string>;
 <consecutive-sync-failures>;</consecutive-sync-failures>;
 <user-credentials>;</user-credentials>;
 <runs-since-last-object-update>;</runs-since-last-object-update>;
 <runs-since-last-full-sync>;</runs-since-last-full-sync>;
 </synchronizer-state>;
</doc>;
```

LDAP フィルター構成ファイル

構成ファイルの object-filter 値には、任意のレベルのフィルタリングを指定できます。標準の演算子の代わりに、次の xml エスケープ文字を使用して、標準の LDAP クエリ構文を使用します。

- AND =「&」& に置き換える
- OR =「|」(縦線) | に置き換える
- NOT =「!」! に置き換える

構成ファイルの参照

サンプルとして参照できる実際の構成ファイルを次に示します。

```
<?xml version="1.0"?>;
<doc>;
 <configuration>;
<description>;SNCTest ADAMSync Configuration</description>;
  <security-mode>;object</security-mode>;
  <source-ad-name>;domaincontroller.service-now.com</source-ad-name>;
  <source-ad-partition>;dc=service-now,dc=com</source-ad-partition>;
  <source-ad-account>;</source-ad-account>;
  <account-domain>;</account-domain>;
  <target-dn>;ou=servicenow users,dc=service-now,dc=adam</target-dn>;
  <query>;
   <base-dn>;dc=service-now,dc=com</base-dn>;
   <object-filter>;(objectCategory=person)</object-filter>;
   <attributes>;
    <include>;objectSID</include>;
    <include>;givenName</include>;
    <include>;sn</include>;
    <include>;description</include>;
    <include>;title</include>;
    <include>;company</include>;
    <include>;department</include>;
    <include>;mail</include>;
    <include>;physicalDeliveryOfficeName</include>;
    <include>;telephoneNumber</include>;
    <include>;userAccountControl</include>;
   </attributes>;
  </query>;
  <user-proxy>;
```

```
  <source-object-class>;user</source-object-class>;
  <target-object-class>;userProxy</target-object-class>;
 </user-proxy>;
 <schedule>;
  <aging>;
   <frequency>;0</frequency>;
   <num-objects>;0</num-objects>;
  </aging>;
  <schtasks-cmd>;</schtasks-cmd>;
 </schedule>;
</configuration>;
<synchronizer-state>;
 <dirsync-cookie>;</dirsync-cookie>;
 <status>;</status>;
 <authoritative-adam-instance>;</authoritative-adam-instance>;
 <configuration-file-guid>;</configuration-file-guid>;
 <last-sync-attempt-time>;</last-sync-attempt-time>;
 <last-sync-success-time>;</last-sync-success-time>;
 <last-sync-error-time>;</last-sync-error-time>;
 <last-sync-error-string>;</last-sync-error-string>;
 <consecutive-sync-failures>;</consecutive-sync-failures>;
 <user-credentials>;</user-credentials>;
 <runs-since-last-object-update>;</runs-since-last-object-update>;
 <runs-since-last-full-sync>;</runs-since-last-full-sync>;
</synchronizer-state>;
</doc>;
```

セキュアな **LDAPS** 通信のための **Microsoft Active Directory** を設定する

証明書のペアを使用して、Microsoft Active Directory (AD) LDAPS 通信を有効にします。

注: これらの手順は、Windows 2003 R2 Standard Edition を使用して設計およびテストされており、Windows 2003 のすべてのバージョンで機能します。

セキュア LDAP (LDAPS) 通信は、サーバーとクライアント間のデータを暗号化する点で SSL (HTTPS) 通信に似ています。これを実現するために、サーバーとクライアントは証明書のペアを使用して共通の情報を共有します。サーバーは秘密鍵証明書を保持し、クライアントは公開鍵証明書を保持します。これらの証明書は、Microsoft Active Directory (AD) LDAPS 通信を有効にするために必要です。

Active Directory の LDAPS を設定するには、次のことを行う必要があります。

- Active Directory ドメインが設定され、インスタンスがファイアウォール経由で Active Directory サーバーに接続できることを確認します。
- ドメインコントローラー (DC) の証明書を発行できる認証局 (CA) があることを確認します。CA インフラストラクチャをまだ使用していない場合は、2 つのオプションがあります。
  - 証明書を発行するためのスタンドアロン CA のセットアップ
  - サードパーティの証明書の要求
- 既に CA がある場合は、内部 CA から証明書を生成できます。

すべての証明書には有効期限があり、証明書のプロパティで確認できます。証明書の有効期限が切れると、すべての LDAPS トラフィックが失敗し、ユーザーはインスタンスにログインできなくなります。これを解決するには、新しい証明書を発行してインスタンスにインストールする必要があります。

Microsoft CA 証明書のデフォルトの有効期限は 1 年です。外部 CA 証明書は通常 1 年単位で購入します。証明書の有効期限が切れたときに注意するか、アプリケーションの有効期限通知機能(システム **LDAP** > 証明書) を開きます。古い証明書の有効期限が切れる前に、新しい証明書があることを確認してください。これにより、古い証明書の有効期限が切れる前に新しい証明書をインストールしてテストする時間が確保できます。

Active Directory のスタンドアロン認証局を設定する

SSL アクセス用に Microsoft Active Directory を設定する最初のステップは、スタンドアロンの認証局 (CA) を設定することです。

始める前に
必要なロール：admin

このタスクについて
必要なサービス (IIS と CA) はどちらも証明書の発行後に無効にできるため、追加のリソース使用率について心配する必要はありません。

手順

1. Internet Information Server (IIS) をインストールします。
2. スタンドアロンモードで認証局サービスをインストールします。
3. 証明書サービス Web アプリケーションがインストールされ、アクティブになっていることを確認します。

次のタスク
IIS マネージャーコンソールを使用して、ローカルコンピューターを展開し、*[Web* サイト] を選択します。[既定の **Web** サイト] のステータスは [実行中] である必要があります。また、[既定の **Web** サイト] の下に *CertSrv* アプリケーションが表示されます。サイトが実行されていないか、アプリケーションが存在しない場合は、続行する前に問題を解決する必要があります。

内部認証局からの証明書を生成する

SSL アクセス用に Microsoft Active Directory を設定する場合は、内部証明書を生成して外部証明書を要求する必要があります。

始める前に
必要なロール：admin

このタスクについて
これらの手順は、Microsoft CA サービスに適用されます。別の内部 CA プラットフォームを使用している場合は、ローカル CA アドミニストレーターにサポートを依頼してください。

手順

1. 証明書を作成するドメインコントローラー (DC) から、`http://localhost/certsrv` を参照するか、CA サーバー名がリモートサーバー上にある場合は指定します。
2. [ようこそ] ページで、[証明書の要求] をクリックし、[証明書の要求の詳細設定] を選択します。
3. [証明書の要求の詳細設定] ページで、[作成] を選択して、この CA に要求を送信します。
4. [証明書の要求の詳細設定] を次のように入力します。

[証明書の要求の詳細設定] フィールド

フィールド	エントリ
名前	証明書を要求している DC の完全修飾ドメイン名 (FQDN)。
メール	証明書の責任者のメールアドレス。
会社	会社名。
キーオプション設定	
新しいキーセットの作成	それを選択します。
CSR	Microsoft RSA SChannel 暗号化プロバイダー。
主な使用法	交換。
キーサイズ	1024 をお勧めします。インスタンスは最大 2048 をサポートします。
自動キーコンテナー名	それを選択します。
ローカルコンピューター証明書ストアへの証明書の保存	それを選択します。

5. [送信] をクリックします。
   [要求 **ID]** が記載されたページが表示されます。この ID をメモします。
6. 処理待ちの要求を処理するには、次の手順を実行します。
   a. 認証局管理コンソールを開きます。
   b. サーバーノードを展開し、[要求処理待ち] を選択します。
   c. 送信した要求の要求 ID を見つけて右クリックし、[すべてのタスク/要求を承認して証明書を発行するために発行する *(All Tasks/Issue to approve the request and issue the certificate)]* を選択します。
7. 発行された証明書を取得するには、次の手順を実行します。
   a. 要求を行った DC から、http://localhost/certsrv を参照するか、リモートサーバー上にある場合は CA サーバー名を指定します。
   b. [処理待ちの証明書の要求の状態] を選択します。
   c. 新しい証明書へのリンクを選択します。
   d. [この証明書のインストール] のリンクを選択します。

次のタスク

サードパーティの証明書を要求する必要があります。外部 CA からの証明書は年間 30 ドルで購入できます。外部 CA からの証明書を要求する詳細な手順については、Microsoft の記事『321051』を参照してください。受領、インストール、およびテスト後、エクスポート手順に従います。

LDAPS 接続のローカルテスト

SSL アクセス用に Microsoft Active Directory を設定する場合は、内部証明書とサードパーティの証明書をインストールした後に LDAPS 接続をテストします。

始める前に

必要なロール：admin

手順

1. Windows サポートツールがドメインコントローラー (DC) にインストールされていることを確認します。
   Support Tools のセットアップ (suptools.msi) は、Windows Server CD の `\Support\Tools` ディレクトリにあります。
2. 移動先 開始 > すべてのプログラム > **Windows** サポートツール > コマンドプロンプト.
   コマンドラインで「`ldp`」と入力してツールを開始します。
3. *ldp*ウィンドウから、 接続 > コネクト をクリックし、ローカル FQDN とポート番号 (636) を指定します。
   また、*[SSL]* を選択します。
   成功すると、Active Directory SSL 接続に関連する情報がウィンドウに表示されます。接続に失敗した場合は、システムを再起動してこの手順を繰り返します。

公開鍵証明書のエクスポートによる **LDAP** 証明書の信頼

SSL アクセス用に Microsoft Active Directory を設定する場合は、公開鍵証明書をエクスポートしてアプリケーションにインポートします。

始める前に
必要なロール：admin

このタスクについて
認証局が信頼できるサードパーティベンダーでない場合は、発行元 CA の証明書をエクスポートして、信頼できるようにする必要があります。また、関連性別に LDAP サーバー証明書を信頼することもできます。MS 証明書サービスユーザーは、エクスポートに使用するコンソールで証明書を表示することで証明書パスを表示できます。[証明書パス] タブを選択します。チェーン内のすべての証明書をエクスポートする必要があります。[証明書パス] で名前を探すと、LDAP 証明書と同じフォルダーで CA 証明書を見つけることができます。インスタンスにインポートするすべての証明書を送信します。

手順

1. 現在または新しい MMC コンソールから、証明書 (ローカルコンピューター) スナップインを追加します。
2. `Personal/Certificate` フォルダーを開きます。
3. 新しい証明書を見つけます。
   [発行先] 列には、ドメインコントローラーの FQDN が表示されます。
4. 証明書を右クリックして、[すべてのタスク/エクスポート *(All Tasks/Export)]* を選択します。
5. DER または Base-64 形式にエクスポートします。
   ファイルに「MyCompany.cer」という形式の名前を付けます。これは、ドメインコントローラーと安全に通信するためにインスタンスで使用する必要がある公開鍵証明書です。
6. 証明書をインスタンスに送信する前に、LDAPS をローカルでテストします。

次のタスク
この手順を完了したら、公開鍵証明書をアプリケーションにインポートします。
証明書をアプリケーションにアップロードするには、「LDAP X.509 SSL 証明書をインストールする」を参照してください。

LDAP グローバルカタログの使用

DC にはグローバルカタログ (GC) ロールを付与できます。グローバルカタログ (GC) ロールは、フォレスト内のすべてのドメインのすべてのオブジェクトの部分的な表現で構成される LDAP 準拠のディレクトリです。

アドミニストレーターは、次のいずれかのホスティング方法を使用して、ライトウェイトディレクトリアクセスプロトコル (LDAP) ディレクトリ情報をホストするように Active Directory を設定します。

- LDAP ディレクトリ情報をホストする一般的な方法は、ポート 389 または 636 でデフォルトの LDAP または LDAPS (セキュア LDAP) を使用することです。これらの標準 LDAP ポートは常にドメインコントローラー (DC) 上に存在し、ほとんど変更されません。このディレクトリパーティションにアクセスすると、DC でホストされているドメイン内のすべてのオブジェクトにアクセスできます。この方法を使用して他のドメインからオブジェクトにアクセスする方法はありません。
- DC にはグローバルカタログ (GC) ロールを付与することもできます。グローバルカタログ (GC) ロールは、フォレスト内のすべてのドメインのすべてのオブジェクトの部分的な表現で構成される LDAP 準拠のディレクトリです。この LDAP ディレクトリにはポート 3268 でアクセスでき、LDAPS ではポート 3269 でアクセスできます。LDAPS とデフォルトの LDAP ポートの証明書要件は同じです。

グローバルカタログ **LDAP** 依存関係

- インスタンスが接続するドメインコントローラーでは、グローバルカタログロールが有効になっている必要があります。
- ファイアウォールルールでは、ポート 3268 (LDAP) または 3269 (LDAPS) でドメインコントローラーへの受信トラフィックを許可する必要があります。

特記事項

- すべての属性が GC パーティションにレプリケートされるわけではありません。名、姓、メール、電話番号、説明、住所などの一般的な属性が含まれています。他の属性を GC に追加できますが、フォレストレプリケーショントラフィックへの影響を最小限に抑えるために制限する必要があります。
- 標準の LDAP 統合では、通常、sAMAccountName をインスタンスの UserID として使用し、LDAP インポートマップの結合キーとして使用します。これは、ドメイン内で一意であることが保証されているためです。ドメインのフォレスト全体を表示する場合、この属性は一意ではなくなります。新しい一意の属性を、UserID および結合キーとして識別する必要があります。これらは同じ属性である必要はなく、フォレストの設計に応じて異なる場合があります。Active Directory アドミニストレーターに相談してください。通常、userPinicpaalName はドメイン間で一意の属性です。これはログインするのに分かりやすい名前ではありませんが、インポート時に一意の識別子として使用できます。UserID に使用される一般的な属性はメールアドレスです。これらの決定は、LDAP プロパティと LDAP マッピングに影響します。
- LDAP インポートマップの結合キーに使用される値は一意であり、インポートされるすべてのオブジェクトに存在する必要があります。一意でないか、存在しない場合は、誤ったレコードが変更されて更新されます。
- 既に LDAP 統合があり、それを GC に変更する場合は、インポート結合キーを変更します。結合キーを変更する前に、新しいキー値をインポートする必要があります。
- 統合が壊れるような変更を LDAP に加えた場合、まずそれらの変更を元に戻す必要があります。その後、試行内容に関する完全な情報を カスタマーサービス & サポート に連絡してください。

OpenLDAP のマイナーなスキーマ変更

back-bdb (Berkley バックエンド) を使用する OpenLDAP 2.3 システムでは、アドミニストレーターがスキーマに軽微な変更を加えて統合を促進します。

⚠ **警告:** ここに記載されているカスタマイズは、特定のインスタンスで使用するために開発されたものであり、Now Support ではサポートされていません。この方法は現状のまま提供され、実装の前に完全にテストする必要があります。このカスタマイズに関するすべての質問およびコメントは、コミュニティフォーラム に投稿してください。

自動翻訳

OpenLDAP 2.3 では、back-bdb は不等式インデックス (順序) を制限付きでサポートしています。これは、generalizedTime および ChangeSequenceNumber 構文に対してのみ実装されます。サブ文字列をサポートする構文ではサポートされません。不等式を含む検索フィルターは、プレゼンスインデックスを使用して処理されます。

既にインデックス化されているものやスキーマに存在するものを変更するのではなく、この目的のためにカスタム属性を作成することをお勧めします (*servnowid* など)。

OpenLDAP スキーマの変更

OpenLDAP スキーマを変更します。以下の手順では、ある顧客から提供された OpenLDAP 2.3 へのスキーマ変更によってインスタンスと統合する方法について詳しく説明します。

始める前に
必要なロール：admin

このタスクについて

⚠ **警告:** ここに記載されているカスタマイズは、特定のインスタンスで使用するために開発されたものであり、Now Support ではサポートされていません。この方法は現状のまま提供され、実装の前に完全にテストする必要があります。このカスタマイズに関するすべての質問およびコメントは、コミュニティフォーラム に投稿してください。

インスタンスとの統合のために OpenLDAP スキーマを変更するには：

手順

1. カスタム属性を作成します。

   Example

```
attribute ( 1.3.6.1.4.1.3403000.2.1.8

    NAME 'servnowid'
 ORDERING caseIgnoreOrderingMatch
 EQUALITY caseIgnoreMatch
 SYNTAX '1.3.6.1.4.1.1466.115.121.1.15' )
```

2. 選択したオブジェクトクラス OID に属性を含めます。

   Example

```
objectclass ( 1.3.6.1.4.1.3403000.2.2.1
    NAME 'BcfUserIdentifiers' SUP top AUXILIARY
 MAY ( uniqid $ unixid $ servnowid ) )
```

   OpenLDAP 2.3 では、サーバー構成を動的に変更できますが、拡張できるのはスキーマのみです。既存のスキーマを変更または削除することはできません。動的構成でこの属性に別のオブジェクトクラスを作成する代わりに、静的構成ファイル slapd.conf を使用します。

3. slapd.conf で、メインデータベースバックエンドの bdb セクションに新しい属性のインデックスを追加します。

   Example

```
database bdb (configs here) ....

index servnowid pres

(other indexes here) .....
```

4. ルートとして、slapindex を実行してこの属性をインデックス化し、検索フィルターで使用できるようにします。
   slapindex を開始する前に、OpenLDAP デーモンが実行されていないか、読み取り専用モードであることを確認してください。

LDAP レコードの削除

デフォルトでは、LDAP から削除された後、インスタンスはエントリを削除しません。

エントリ (レコードとも呼ばれます) を削除すると、履歴全体と削除されたエントリへの参照も削除されます。

たとえば、構成アイテム (CI)、SLA 契約、ソフトウェアライセンス、発注書 (PO)、およびサービスカタログエントリにはすべて部門への参照があり、部門が削除されると、その部門へのすべての参照がクリアされます。また、ユーザーを削除すると、そのユーザーの実行履歴がすべて失われます。

組織のニーズに応じて、テーブル内のすべてのレコードの削除 するか保持するかを決定します。

同時セッションの制限

すべてのノードにわたって、インスタンスのユーザーまたはロールの同時インタラクティブセッションの数を制限できます。

探索	アクティブ化
	
同時セッション制限の機能とビジネス価値について説明します。	同時セッション制限を有効にする方法について理解します。

自動翻訳

セット

ユーザーまたはロールに同時セッション制限を設定します。

無効化

同時セッション制限を無効にする方法について学びます。

自動翻訳

同時セッション制限の詳細

すべてのノードにわたって、インスタンスのユーザーまたはロールの同時インタラクティブセッションの数を制限できます。

同時インタラクティブセッションは、ユーザーが ServiceNow インスタンスごとにアクティブにできるセッションの数を参照します。アクティブなインスタンスセッションは、特定の ServiceNow インスタンスに新たにログインするたびに発生します。デフォルトでは、ユーザーが保持できるアクティブなインスタンスセッションの数に制限はありません。

Jakarta リリースでは、ユーザーごとのアクティブな同時セッション数を制限できます。アクティブなセッションの最大数に達した後にユーザーがログインすると、最も古いアクティブセッションが終了し、新しいインタラクティブセッションがアクティブになります。ユーザーがブラウザーを介してクローズ済みセッションにアクセスしようとすると、ユーザーはログインページにリダイレクトされます。

注: セッションの上限を有効にするには、**Limit concurrent sessions** プラグインをアクティブ化する必要があります。制限は、glide.authenticate.max.concurrent.interactive.sessions プロパティによって設定されます。上限値は、制限プロパティがアクティブなすべてのユーザーまたはロールに適用されます。ユーザーまたはユーザーに接続されているロールは、開始するセッションの制限に対して limit_concurrent_sessions 値を true に設定する必要があります。Jakarta リリースの場合、この機能はネイティブモバイルアプリまたは非インタラクティブメカニズムによって作成されたセッションをサポートしません。

最大同時セッション数が 1 に設定されている場合の一般的なユースケース：

1. ユーザーは、Chrome を介して最初の ServiceNow インスタンスにアクセスします。
2. ユーザーが正常にログインすると、ServiceNow によってユーザーにセッション 1 (S1) が作成されます。
3. ユーザーは、Firefox を介して ServiceNow インスタンスへの別のアクセスを開始することにします。
4. ユーザーが正常にログインすると、ServiceNow によってユーザーにセッション 2 (S2) が作成されます。
5. ユーザーの同時セッション数の上限は 1 であるため、S2 セッションが作成されると S1 が無効になります。
6. ユーザーが Chrome から S1 ServiceNow インスタンスに再度アクセスすると、S1 が無効であるため、ユーザーはログインページにリダイレクトされます。

同時セッション制限は、すべての ServiceNow 認証メカニズム (SAML、LDAP、およびローカルデータベース認証) で機能します。また、マルチファクター認証とすべてのインタラクティブ ServiceNow 認証メカニズムでも機能します。セッションのソースは、[タイプ] 列の **sys_user_session** テーブルを介して表示できます。値は次のとおりです。

- Web ブラウザー
- モバイルブラウザー
- ServiceNow モバイルアプリ
- 非インタラクティブ (SOAP、WSDL、OAuth)

Limit Concurrent Sessions プラグインのアクティブ化と設定

admin ロールを持っている場合は、Limit Concurrent Sessions プラグイン (com.glide.limit.concurrent.sessions) をアクティブ化できます。

始める前に

必要なロール：admin

このタスクについて

手順

1. 移動先 すべて > システム定義 > プラグイン.
2. 同時セッション制限プラグインを見つけてクリックします。
3. [システムプラグイン] フォームでプラグインの詳細を確認し、[有効化/アップグレード] 関連リンクをクリックします。
4. [アクティブ化] をクリックします。
5. この機能を有効にして同時セッションの上限を設定するには、[プラグインファイル] タブに移動し、次のプロパティを見つけて設定値を変更します。

オプション	説明
glide.authenticate.limit.concurrent.interactive.sessions	値を **True** に設定することで、同時セッション制限機能を有効にできます。デフォルトでは、このプロパティは **False**に設定されています。つまり、ユーザーがアクティブにできるインタラクティブセッションの数に制限がないことを意味します。

オプション	説明
	注: この機能を無効にするには、 プロパティを **False** に設定します。
glide.authenticate.max.concurrent.interactive.sessions	すべてのノードにわたって、インスタンス上で同時にアクティブにできるインタラクティブセッションの最大数を設定できます。

6. オプション: 必要に応じて、次のプロパティを変更することもできます。

オプション	説明
glide.authenticate.session.types.to.limit.concurrency	このプロパティはセッションタイプを制限します。デフォルトでは、Web ブラウザセッションにのみ制限があります。セッションタイプは次のとおりです。 ○ Web ブラウザ (1) ○ モバイルブラウザ (2) ○ ServiceNow モバイルアプリ (3) ○ 非インタラクティブ (10) Web ブラウザの場合は「1」、モバイルブラウザの場合は「2」、またはその両方の場合は「1,2」に値を設定できます。 注: Web およびモバイルブラウザセッションでのみ制限を設定できます。ServiceNow モバイルアプリまたは非インタラクティブセッションから発生するセッションには制限がありません。
glide.authenticate.limit.concurrent.sessions.at.cluster.level	このプロパティは、 ServiceNow インスタンスのすべてのノードにわたって同時セッションを制限するのではなく、ノードごとに同時セッション数を制限します。デフォルトでは、値は true に設定されており、すべてのノードでユーザーセッションが制限されます。プロパティが false に設定されている場合、そのノードのセッションのみが制限の対象になり、他のノードのセッションは対象になりません。

7. [更新] をクリックして設定を有効にします。

次のタスク
ユーザーまたはロール別の同時セッション制限の設定。

関連トピック
プラグインのリスト (Zurich)

ユーザーまたはロール別の同時セッション制限の設定

特定のユーザーまたは特定のロールに同時セッション制限を設定できます。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > ユーザー管理 > ユーザー または ユーザー管理 > ロール.
2. 同時セッション制限を設定するユーザーまたはロールを選択し、[同時セッションを制限] チェックボックスをオンにして、[更新] をクリックします。
   ユーザーまたはロールには、一度にオープンできる同時セッション数の制限があります。

ユーザーまたはロール別の同時セッション制限の無効化

特定のユーザーまたは特定のロールに対して同時セッション制限を無効にできます。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > ユーザー管理 > ユーザー または ユーザー管理 > ロール.
2. 同時セッション制限を無効にするユーザーまたはロールを選択し、[同時セッションを制限] チェックボックスをオフにして、[更新] をクリックします。
   ユーザーまたはロールは、一度にオープンできる同時セッション数の制限対象ではなくなりました。

ローカル認証

ServiceNow® ローカル認証を使用して、ローカルデバイスでのユーザーのログインを保護します。

ローカル認証は、ローカルデバイスでユーザーの本人確認を行う方法です。ユーザーが持つ固有の特性または認証情報を使用して、ユーザーのログインを検証します。

ServiceNow® のローカル認証機能を使用して、ローカルデバイス上の ServiceNow® インスタンスに安全にアクセスします。

ログインと認証のセキュリティ

インスタンスへのアクセスを制御するログインセキュリティオプションを設定します。

ログインセキュリティの設定

高セキュリティを構成する方法について説明します。

ログインシナリオの定義

ログインシナリオを定義します。

パスワードリセット

ログインパスワードのリセットについて学びます。	

ログインと認証のセキュリティの詳細

インスタンスへのアクセスを制御するログインセキュリティオプションを設定します。

セキュリティオプション

ユーザーのログインと認証セキュリティのいくつかの側面を制御できます。

機能	説明	関連トピック
ログインおよびログアウトのコントロール	ログイン時にユーザーに表示するランディングページを指定するなど、ユーザーのログインおよびログアウトプロセスのいくつかの側面を制御します。また、ユーザーのログアウト方法を制御します。	• ログインシナリオの定義 • ログアウト確認プロンプトを設定する • インストレーションイグジット • 失敗したログイン試行のロックアウトを指定する
認証セキュリティ	パスワードリセットプロセスと [記憶する] オプションなどの機能を制御します。インスタンスへのアクセスに IP アドレスベースのコントロールを使用することができます。また、Single Sign-on Digest Authentication で使用するノンスを実装することもできます。	• パスワードポリシーの設定 • パスワードリセット • 記憶する • IP 範囲ベースの認証 • ノンスを実装する

ログインシナリオの定義

ログイン後にすべてのユーザーを同じページに誘導することができます。

始める前に
必要なロール：admin

このタスクについて
ユーザーが http://{instance_name}.service-now.com/ にアクセスするなど、インスタンスに直接ログオンすると、システムは次のことを行います。

1. プロパティ *glide.entry.page.script* の値にアクセスします。プロパティのデフォルト値は、*CMSEntryPage* という名前のスクリプトインクルードから派生します。
2. エントリページでログインが必要な場合、ユーザーをインスタンスのログインページに誘導します。
3. ログインルールがある場合は、ユーザーに適用します。

ログイン後にすべてのユーザーを同じページに誘導するようにするには、次の手順を実行します。

手順

1. 移動先 すべて > コンテンツ管理 > 設定 > 環境設定ページ.
2. [ログインページ] フィールドの値を選択するか、必要に応じて新しいページを作成します。

   このページがサイトのデフォルトページでない場合は、常にここにリダイレクトされます。サイトのデフォルトページの場合は、ログインルールが適用されます。この値が null の場合、システムは navpage.do をエントリページとして使用します。ここにログインページを入力しないでください。入力すると、ユーザーは 2 回ログインする必要があります。

   インスタンスにログインしてレコードにアクセスする場合：

   ユーザーがインスタンスにログインして、http://{instance}.service-now.com/incident.do?sys_id={sys_id} などのグローバルに一意な識別子 (sys_id) でレコードにアクセスすると、システムは以下を実行します。

   a. まだログインしていない場合は、ユーザーをログインページに誘導します。
   b. アクセスが許可されている場合、ユーザーを適切なレコードに誘導します。ユーザーがレコードへのアクセス権を持っていない場合は、アクセスを拒否するメッセージが表示されます。

   サービスポータルまたは CMS サイトにログインする場合：

   ユーザーがサービスポータルまたは CMS サイト (http://<instance>.service-now.com/site-name/page.do など) にログオンすると、システムは以下を実行します。

   - サービスポータルまたは CMS サイトフォームの [ログインページ] フィールドに値がある場合、ユーザーはそのログインページに誘導され、ログインルールがある場合はそれが適用されます。
   - ログインページが指定されていない場合は、サービスポータルまたは CMS サイトフォームの [ホームページ] フィールドの値に誘導されます。

   サービスポータルまたは CMS サイトにログインしてレコードにアクセスする場合：

   ユーザーがレコードにアクセスするためにサービスポータルまたは CMS サイト (http://{instance}.service-now.com/ess/incident_detail.do?sysparm_document_key=incident,{sys_id} など) にログオンすると、同じ手順が実行され、最後にレコードに誘導されます。ユーザーがレコードへのアクセス権を持っていない場合は、アクセスを拒否するメッセージが表示されます。

ログインと従業員セルフサービスポータル

ユーザーが従業員セルフサービスポータルにログインしようとしても、ユーザーがアクセスしようとしている最初のページの追跡を続けます。

次のシナリオを考えてみましょう。

例 1：

1. ユーザーはログインしておらず、特定の SYS ID を URL 内で使用してレコードにアクセスしようとします。
2. ユーザーはログインページにリダイレクトされます。
3. ユーザーはログインするのではなく、従業員セルフサービス (/ess) ポータルなどの別のサイトにアクセスしようとします。
4. ユーザーはログインページに再度リダイレクトされます。
5. ユーザーはログインすると、従業員セルフサービスポータルではなく、ユーザーが最初にアクセスしようとしたレコードにリダイレクトされます。

例 2：

1. ユーザーはログインしておらず、従業員セルフサービス (/ess) ポータルから特定の SYS ID を URL 内で使用してレコードにアクセスしようとします。
2. ユーザーはログインページにリダイレクトされます。
3. ログインするのではなく、ユーザーは従業員セルフサービスポータルから別のレコードにアクセスしようとします。
4. ユーザーはログインページに再度リダイレクトされます。
5. ユーザーはログインすると、2 番目のレコードではなく最初のレコードにリダイレクトされます。

ログインランディングページの指定

デフォルトでは、ログイン時にホームページが表示されます。システムプロパティまたはコンテンツ管理システムを使用して、別のログインランディングページを指定することができます。

始める前に
必要なロール：admin

このタスクについて
すべてのユーザーのログインランディングページを指定するには、sys_properties テーブルのプロパティ値を変更します。

手順

1. ナビゲーションフィルターで、「sys_properties.list」と入力します。
2. **glide.login.home** システムプロパティを見つけます。
3. [値] フィールドに、ログイン時にすべてのユーザーに表示されるページ名を入力します。

   <page name>.do を使用すると、URL の http://"instance".service-now.com/ の部分を省略できます。システム内のページ名またはページの URL を指定するために、リンクをポイントすることができます。可能なページは、welcome.do と incident.doです。

   ダッシュボードのランディングページを指定するには、プロパティを $dashboards.do?dashboard=<SYS_ID> に設定します。<SYS_ID> をダッシュボードの sys_id に置き換えます。

   ユーザーをサービスポータルに誘導するには、プロパティを /sp に設定します。

> **注:** このプロパティはシステム全体に適用されるため、設定はすべてのユーザーに影響します。ロールのないユーザー専用のログインを設定するには、同じ手順を適用し、**glide.entry.loggedin.page_ess** プロパティを使用します。

コンテンツ管理システムでログインランディングページを指定することもできます。

失敗したログイン試行のロックアウトを指定する

システムで提供されている非アクティブなスクリプト アクションを使用すると、ユーザー アカウントをロックする前のログイン試行の失敗回数を指定でき、ログイン成功後にカウントをリセットすることができます。

始める前に
必要なロール：admin

手順

移動先 すべて > システムポリシー > スクリプトアクション をクリックして、スクリプトを表示またはアクティブ化します。

> **注:** Kingston リリース以降の zBoot 以後では、「**SNC** ユーザーのロックアウトの確認と自動ロック解除」および「**SNC** ユーザーのクリア」のスクリプトアクションがアクティブ化されています。
>
> 失敗したログインの試行に影響するプロパティの詳細については、インスタンスセキュリティ強化設定の「失敗したログインの試行の管理 (インスタンスセキュリティ強化)」を参照してください。

スクリプトアクション	説明
SNC ユーザーのロックアウトの確認と自動ロック解除	• `glide.user.max_unlock_attempts` プロパティの値を使用して、ログイン試行の失敗回数の制限を設定します。 • `glide.user.unlock_timeout_in_mins` プロパティで指定された時間が経過すると、ユーザーアカウントのロックが解除されます。値を指定しない場合は、デフォルトの 15 分後にユーザーアカウントのロックが解除されます。
SNC ユーザーのロックアウトの確認	ログイン試行の失敗回数を追跡し、指定した回数のログイン試行の失敗後にユーザーアカウントをロックします (デフォルトは 5 回)。
SNC ユーザーのクリア	ログイン成功後にユーザーレコードを更新します。ログイン試行の失敗回数をリセットし、最終ログイン日を更新します。

次のタスク
ユーザーがログインを試みると、そのたびにアクションがイベントログに記録されます。ログイン試行の失敗のログを表示できます。

1. 移動先 システムポリシー > イベントログ.
2. [名前] フィールドでフィルターを適用し、**[login.failed]** を探します。試行されたログインの名前、日付、およびログイン試行が行われた IP アドレスを表示できます。

UI ページを公開または非公開にする

ユーザーがログインせずにページを表示できるようにするには、ページを公開にします。

始める前に
必要なロール：admin

このタスクについて
ほとんどのページは、ログインしたユーザーにのみ表示されます。限られた数のページ (ようこそページ、フロントページ、ログインページとログアウトページなど) は、ユーザーがログインしなくても表示できるように公開されています。

> ⚠ **警告:** 多くの機能にはいくつかのベースシステムの公開ページが必要です。ベースシステムの公開ページを無効にしないでください。

手順

1. アプリケーションナビゲーターフィルターに「csm_table_map.list」と入力します。
2. [新規] をクリックします。
3. sys_public テーブルで、次の値のレコードを作成します。

フィールド	説明
ページ	ページ名例：$sp
有効	選択すると、ページが公開されます。ページを非公開にする場合は、[アクティブ] オプションの選択を解除します。

4. [保存] をクリックします。
   アクティブを true に設定するとページが公開されるため、<instance_name>/sp または <instance_name>/$sp.do によってページにアクセスできます。

パスワードの複雑さの要件

ServiceNow® インスタンスのパスワードは複雑さの要件を満たす必要があります。

探索

パスワードの複雑さの要件の特徴とビジネス価値について学びます。

有効化

パスワードの複雑さの要件を有効にする方法を理解します。

構成

パスワードの複雑さの要件を設定します。

参照 - サポートされていないパスワード文字

自動翻訳

サポートされていないパスワード文字を把握します。

パスワードの複雑さの要件の確認

ServiceNow® インスタンスのパスワードは複雑さの要件を満たす必要があります。

パスワードの複雑さに関する要件ポリシー設定は、パスワードが一連の強力なパスワードガイドラインを満たす必要があるかどうかを決定します。

パスワードの複雑さの要件は、パスワードが変更または作成されるときに適用されます。

パスワードの複雑さの要件は、以下の条件に基づいて遵守され、機能します。

- *glide.apply.password_policy.on_login* プロパティが有効になっている場合、ログイン時にユーザーに対してパスワードポリシーチェックが適用されます。ログイン時に、ユーザーはパスワードポリシーに従い、インスタンスのパスワードを変更する必要があります。
- パスワードポリシー要件は、現代の言語すべての文字を含む Basic Multilingual Plane (BMP) に基づいています。ServiceNow インスタンスには、約 10,000 文字の BMP が付属しています。
- 許容される BMP 内のパスワードをインスタンスに設定できます。これらの BMP に準拠しないパスワードは許可されません。

必要に応じて、少なくとも 90 日ごとにパスワードを変更するように要求できます。

要件と禁止文字

安全なネットワーク環境を有効にするには、文字、数字、および記号を組み合わせた強力なパスワードを使用する必要があります。このような組み合わせを使用することで、権限のないユーザーが主に手動または自動化された方法で脆弱なパスワードを推測することが起こりにくくなります。

インスタンスのパスワードは、次の要件を満たしている必要があります。

- 最小 8 文字
- 最大 100 文字
- 小文字と大文字を含む
- 特殊文字を含む
- 数字を含む

次のような、一連のよくある不適切なパスワードを禁止できます。

- 「123456」、「qwerty」、「!@#$%^」、「aaaaa」などの予測可能な繰り返しシーケンス
- 従業員名またはユーザー名
- 関連するブランド名または製品名
- 本社、都市、国などの場所名
- 会社固有の内部用語または略語
- 絵文字

注: パスワードに使用できないユーザー、ブランド、または会社固有の文字は、[パスワードポリシー] または [パスワードを除外] ページで設定できます。

インスタンスでのパスワードポリシーの有効化

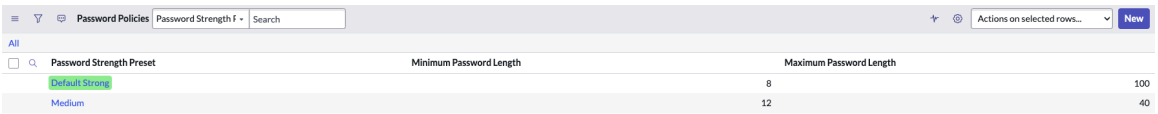
ログイン時にパスワードポリシーコントロールを実装します。パスワードがパスワードポリシー基準を満たしていない場合、ユーザーにパスワードの変更を強制します。

始める前に
必要なロール：admin

このタスクについて
Password Policy プラグイン (com.glide.password_policy) はデフォルトで有効になっています。ユーザーがパスワードを変更またはリセットすると、ポリシーが有効になります。
[パスワード強度のプリセット] フィールドは、自動的に [デフォルトは強] に設定されます。新しい基準を追加する場合は、次の手順を実行できます。

ValidatePasswordStronger インストレーションイグジットまたはパスワードリセット認証情報ストアの *pwd_cred_store* プロパティを使用してインスタンスをカスタマイズした場合は、パスワードポリシーのプロパティを参照して、インスタンスのパスワードポリシーを実装する方法を確認してください。

注: 次のように、インスタンスのアクティブなパスワードポリシーが強調表示されます。



パスワードポリシーを変更するには、次の場所に移動します。 すべて > パスワードリセット > 認証情報ストアをクリックし、認証情報を選択して、[パスワードポリシー] フィールドを必要なポリシー入力に変更します。

手順
1. 移動先 すべて > パスワードのポリシー > パスワードのポリシー.
2. [**New**] をクリックします。
   [パスワードポリシー] フォームが表示されます。
3. パスワードポリシーの [名前] を指定します。
4. [パスワードポリシー基準] セクションで、[パスワード強度のプリセット] フィールドから次のいずれかのプリセットを選択します。

パスワード強度のプリセット	説明
デフォルト	必要なパスワード文字のフィールドを次のように自動入力します。 ○ [大文字の最小文字数] を 1 に設定します。 ○ [小文字の最小数] を 1 に設定します。 ○ [数値の最小文字数] を 1 に設定します。 ○ [特殊文字の最小文字数] を 0 に設定します。 パスワードの長さは最小 8 文字、最大 100 文字です。
中	必要なパスワード文字のフィールドを次のように自動入力します。

パスワード強度のプリセット	説明
	○ [大文字の最小文字数] を 1 に設定します。 ○ [小文字の最小数] を 1 に設定します。 ○ [数値の最小文字数] を 1 に設定します。 ○ [特殊文字の最小文字数] を 1 に設定します。 パスワードの長さは最小 12 文字、最大 40 文字です。
高	必要なパスワード文字のフィールドを次のように自動入力します。 ○ [大文字の最小文字数] を 1 に設定します。 ○ [小文字の最小数] を 2 に設定します。 ○ [数値の最小文字数] を 1 に設定します。 ○ [特殊文字の最小文字数] を 3 に設定します。 パスワードの長さは最小 8 文字、最大 100 文字です。
デフォルトは強	必要なパスワード文字のフィールドを次のように自動入力します。 ○ [大文字の最小文字数] を 1 に設定します。 ○ [小文字の最小数] を 1 に設定します。 ○ [数値の最小文字数] を 1 に設定します。 ○ [特殊文字の最小文字数] を 1 に設定します。 パスワードの長さは最小 8 文字、最大 100 文字です。
カスタム	必要なパスワード文字のフィールドを次のように自動入力します。 ○ [大文字の最小文字数] を 1 に設定します。 ○ [小文字の最小数] を 1 に設定します。 ○ [数値の最小文字数] を 1 に設定します。 ○ [特殊文字の最小文字数] を 1 に設定します。 パスワードの長さは最小 8 文字、最大 100 文字です。 表示されるパスワードポリシーのスクリプトをカスタマイズすることもできます。
詳細	[詳細] を選択すると、[パスワードのルールスクリプト] と [パスワードの安全性スクリプト] が表示されます。要件に基づいて、これらのスクリプトをカスタマイズできます。

注: パスワードポリシーは、選択したプリセットに基づいて適用されます。

5. フォームの残りのフィールドに入力します。

[パスワードのポリシー] フォーム

フィールド	説明
パスワードの最小長	パスワードの最小長。このオプションは、[詳細] を除くすべてのプリセットで表示されます。このフィールドは最低 8 ～ 10 文字に設定します。
パスワードの最大長	パスワードの最大長。このオプションは、[詳細] を除くすべてのプリセットで表示されます。このフィールドは最大 100 文字に設定します。
大文字の最小文字数	パスワードの大文字の最小数 (0 ～ 10)。
小文字の最小数	パスワードの小文字の最小数 (0 ～ 10)。
数字の最小文字数	パスワードの数字の最小数 (0 ～ 10)。
特殊文字の最小文字数	パスワードの特殊文字の最小数 (0 ～ 10)。
含まれる特殊文字	区切り文字のない特殊文字セットの使用を制限できるようにします。 たとえば「$,!」と入力すると、ユーザーはパスワードの特殊文字として「$」と「!」のみを使用できます。他の特殊文字は使用できません。他の特殊文字を含むパスワードは使用できません。
除外される特殊文字	区切り文字のない特殊文字セットの使用を制限できるようにします。 たとえば「@$!」と入力すると、ユーザーはパスワードの特殊文字として「@」、「$」、「!」を使用できなくなります。 **注:** このオプションは、*glide.password_policy.use_excluded_specia* プロパティが有効になっている場合に使用できます。
ユーザーデータを禁止	認証関連のユーザーデータを禁止するオプション
シーケンス長のしきい値	パスワードのシーケンス長
繰り返しの長さのしきい値	パスワードの繰り返しの長さ。

フィールド	説明
	注: ○ シーケンス長しきい値と繰り返し長さのしきい値は、どちらも最大 8 文字を設定できます。これらのフィールドを使用すると、「123456」、「qwerty」、「!@#$%^」、「aaaaa」などの予測可能な繰り返しシーケンスを持つ脆弱なパスワードの組み合わせを制限できます。 ○ [パスワード強度のプリセット] が [デフォルトは強] に設定されている場合、シーケンス長のしきい値と繰り返し長さのしきい値は、両方とも 4 文字に設定されます。
パスワードをテスト	このフィールドに実際のパスワードを指定します。

6. [パスワードをテスト] をクリックします。
7. パスワードが有効であることを確認したら、[送信] をクリックしてパスワードを送信します。

注: 送信する前に必ずパスワードをテストしてください。

パスワードポリシープロパティ

パスワードポリシープロパティを使用すると、パスワードポリシーを管理し、リストパスワードを除外し、ログイン時にパスワードポリシーを適用できます。

移動先 パスワードのポリシー > プロパティ をクリックして、パスワードポリシーのプロパティを表示および編集します。

プロパティ	説明
glide.enable.password_policy	インスタンスでパスワードポリシーを有効にします。ユーザーがパスワードを変更またはリセットすると、ポリシーが有効になります。このプロパティは自動的に **true** に設定されます。

プロパティ	説明
	注: • *ValidatePasswordStronger* インストレーションイグジットまたはパスワードリセット認証情報ストア [pwd_cred_store] を介してインスタンスがカスタマイズされている場合は、このプロパティを作成してシステムプロパティに追加する必要があります。 • Orlando より前のリリースでは、*ValidatePasswordStronger* インストレーションイグジットを使用してインスタンスをカスタマイズした場合、パスワードポリシーを機能させるにはパスワードポリシープロパティを作成する必要がありました。 • Orlando リリース以降は、インストレーションイグジットのカスタマイズはありません。パスワードポリシープロパティはデフォルトで機能します。これらのプロパティは手動でオフにすることができます。
glide.enable.blacklist_password	特定のパスワードの使用を禁止します。アドミニストレーターは、[除外されたパスワード] テーブルにパスワードを挿入できます。このプロパティは自動的に **true** に設定されます。
glide.apply.password_policy.on_login	既存のパスワードが現在のパスワードポリシーに準拠していない場合、次回ログイン時にユーザーにパスワードの変更を強制します。 このプロパティは自動的に **false** に設定されます。値を **true** に設定すると、ログイン時にパスワードポリシーが適用されます。 注: このプロパティを有効にすると、新しいパスワードポリシーに準拠していないユーザーが大量にパスワードの変更を強制される可能性があります。
glide.password_policy.user_excluded_special_char	[パスワードポリシー] フォームで [除外される特殊文字] オプションの使用を有効化します。
glide.validate.sys_user.password.field	admin が sys_user フォームまたはリストビューを編集するときに、パスワードポリシーに対してユーザーパスワードを検証できるようにします。
glide.password.policy..generate.password.field.disabled	sys_user フォームのパスワード設定ポップアップで、[パスワード] フィールドを無効にします。
glide.user.show.password.field	sys_user フォームの [パスワード] フィールドを有効にします。

プロパティ	説明
glide.password_policy.debug	パスワードポリシーのデバッグログを有効にします。

パスワードポリシーの設定

パスワードポリシー基準により、パスワードを保護して、パスワードの複雑さの最小要件を遵守することができます。

始める前に

必要なロール：admin

このタスクについて

Password Policy (com.glide.password_policy) プラグインはデフォルトで有効になっています。ユーザーがパスワードを変更またはリセットすると、ポリシーが有効になります。ValidatePasswordStronger インストレーションイグジットまたはパスワードリセット認証情報ストア [pwd_cred_store] を使用してインスタンスをカスタマイズした場合は、「パスワードポリシープロパティ」を参照してください。

手順

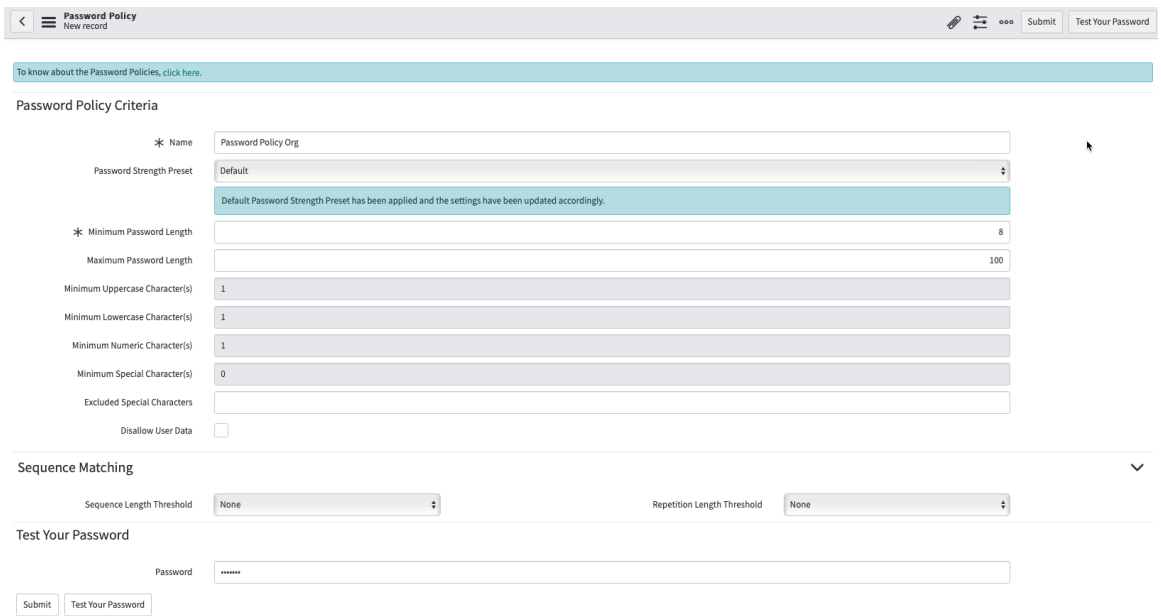
1. 移動先 すべて > パスワードのポリシー > パスワードのポリシー.

> **注:** [デフォルトは強] のプリセットが、デフォルトのパスワード受け入れ条件として有効になっています。場合により、新しい基準を追加する場合は、次の手順を実行できます。

2. [新規] をクリックします。

[パスワードポリシー] の [新規レコード] ページには次のセクションがあります。これらを指定してパスワードを設定します。

- パスワードポリシー基準
- シーケンスマッチング
- パスワードをテスト



3. パスワードポリシーの [名前] を指定します。

4. [パスワードポリシー基準] セクションで、[パスワード強度のプリセット] からプリセットを選択します。選択可能なプリセットは次のとおりです。

パスワード強度のプリセットとその説明

パスワード強度のプリセット	説明
デフォルト	[デフォルト] を選択すると、必要なパスワード文字を次のように自動入力します。 ○ 大文字を最低 1 文字 ○ 小文字の最小文字数 1 ○ 数字を最低 1 文字
中	[中] を選択すると、次の条件に基づいて、必要なパスワード文字を自動入力します。 ○ 大文字を最低 1 文字 ○ 小文字の最小文字数 1 ○ 数字の最小文字数 1 ○ 特殊文字の最小文字数 1
高	[高] を選択すると、パスワード文字を次のように自動入力します。 ○ 大文字を最低 1 文字 ○ 大文字の最小文字数 2 ○ 数字の最小文字数 1 ○ 特殊文字の最小文字数 3
デフォルトは強	[デフォルトは強] を選択すると、次の条件に基づいて、必要なパスワード文字を自動入力します。 ○ 大文字を最低 1 文字 ○ 小文字の最小文字数 1 ○ 数字の最小文字数 1 ○ 特殊文字を最低 1 文字
カスタム	[カスタム] を選択すると、次の条件に基づいて、必要なパスワード文字を自動入力します。 ○ 大文字を最低 1 文字 ○ 小文字の最小文字数 1 ○ 数字の最小文字数 1 ○ 特殊文字を最低 1 文字 表示されるパスワードポリシーのスクリプトをカスタマイズすることもできます。
詳細	[詳細] を選択すると、[パスワードのルールスクリプト] と [パスワードの安全性スクリプト]

パスワード強度のプリセット	説明
	が表示されます。要件に基づいて、これらのスクリプトをカスタマイズできます。

注: パスワードポリシーは、選択したプリセットに基づいて適用されます。

5. テーブルに記載されているフィールドを次のように指定します。

[パスワードのポリシー] フォーム

フィールド	説明
パスワードの最小長	パスワードの最小長。このオプションは、[詳細] を除くすべてのプリセットで表示されます。 注: [パスワードの最小長] は必須フィールドであり、最低でも 8 ～ 10 文字に設定することをお勧めします。
パスワードの最大長	パスワードの最大長。このオプションは、[詳細] を除くすべてのプリセットで表示されます。 注: [パスワードの最大長] はオプションのフィールドです。最大 100 文字に設定することをお勧めします。
大文字の最小文字数	パスワードの大文字の最小数 (0 ～ 10)。
小文字の最小数	パスワードの小文字の最小数 (0 ～ 10)。
数字の最小文字数	パスワードの数字の最小数 (0 ～ 10)。
特殊文字の最小文字数	パスワードの特殊文字の最小数 (0 ～ 10)。
含まれる特殊文字	区切り文字のない特殊文字セットの使用を制限できるようにします。たとえば「$,!」と入力すると、ユーザーはパスワードの特殊文字として「$」と「!」のみを使用できます。他の特殊文字は使用できず、他の特殊文字を含むパスワードは使用できません。
除外される特殊文字	区切り文字のない特殊文字セットの使用を制限できるようにします。たとえば「@$!」と入力すると、ユーザーはパスワードの特殊文字として「@」、「$」、「!」を使用できなくなります。 注: このオプションは、glide.password_policy.use_excluded_special_char プロパティが有効になっている場合に使用できます。
ユーザーデータを禁止	ユーザーデータを禁止することが可能です。

6. [シーケンスマッチング] セクションで、[シーケンス長のしきい値] と [繰り返しの長さのしきい値] を指定します。

   注:

   - シーケンス長しきい値と繰り返し長さのしきい値は、どちらも最大 8 文字を設定できます。これらのフィールドを使用すると、「123456」、「qwerty」、「!@#$%^」、「aaaaa」などの予測可能な繰り返しシーケンスを持つ脆弱なパスワードの組み合わせを制限できます。
   - [デフォルトは強] の場合、シーケンス長のしきい値と繰り返し長さのしきい値は、両方とも 4 文字で選択されます。

7. [パスワードをテスト] セクションで、パスワードを指定します。
8. [パスワードをテスト] をクリックします。
9. パスワードが有効であれば、[送信] をクリックしてパスワードを送信します。

   注: 送信する前に必ずパスワードをテストしてください。

ユーザーのパスワードの設定

構成されているパスワードポリシーに基づいて、インスタンスのユーザーのパスワードを設定します。

始める前に

最初のログイン用のパスワードを設定するために作成されたユーザー。詳細については、「ユーザーの作成」を参照してください。

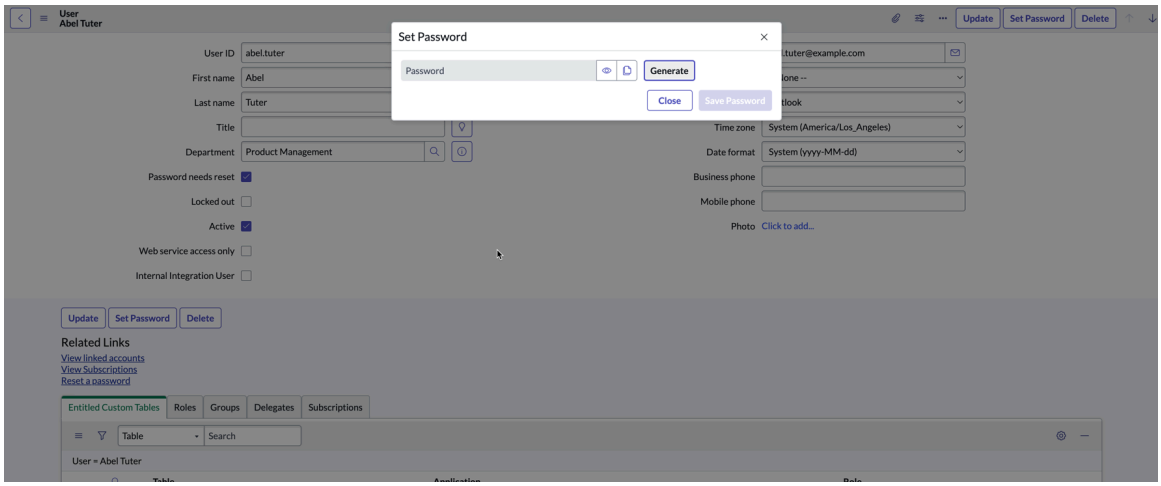
ユーザーフォームの [パスワード] フィールドに直接入力するには、[**sys_user** フォームでのパスワードの表示を有効にする **(Enable to show the password field on the sys_user Form)**] (glide.user.show.password.field) を有効にします。プロパティの詳細については、「パスワードポリシープロパティ」を参照してください。

必要なロール：admin

手順

1. 移動先 すべて > ユーザー管理 > ユーザー.
2. [ユーザー (Users)] ページのリストからユーザーを選択します。
3. パスワードポリシーに基づいてパスワードを設定するには、[パスワードを設定] をクリックします。

   [パスワードを設定] ポップアップが表示されます。



4. [パスワードを設定] で、次の操作を実行します。

a. [生成] をクリックしてパスワードを生成します。

b. [表示] アイコンをクリックしてパスワードを表示します。

c. [コピー] アイコンをクリックして、ユーザーと共有するパスワードをコピーします。

5. [パスワードを保存] をクリックします。

ユーザーにパスワードが設定されます。また、[パスワード休止 (Password needs rest)] チェックボックスが自動的に有効になります。

ユーザーは、最初のログイン時は同じパスワードを使用し、ログインしたらアドミニストレーターが設定したパスワードポリシーに従ってパスワードを変更する必要があります。

インスタンスのパスワードポリシーによるパスワードの除外

[除外されたパスワード] テーブルにパスワードを追加して、インスタンスのユーザーが特定のパスワードを使用できないようにします。

始める前に
必要なロール：admin

このタスクについて
必要に応じていくつでもパスワードを禁止できます。いくつかの例を次に示します。

- 「123456」、「qwerty」、「!@#$%^」、「aaaaa」などの予測可能な繰り返しシーケンス
- 従業員名またはユーザー名
- 関連するブランド名または製品名
- 本社、都市、国などの場所名
- 会社固有の内部用語または略語
- 絵文字

手順

1. 移動先 すべて > パスワードのポリシー > 除外されたパスワード.
2. 除外リストにパスワードを追加するには、[新規] をクリックしてパスワードを入力します。
3. 移動先 パスワードのポリシー > 除外リストの管理 を使用してユーザーのパスワードを管理し、ユーザーがインスタンスに対して間違ったパスワードを使用するのを防ぎます。

[除外リストの管理] には、一般的に使用される約 5000 のパスワードがあります。

サポートされていないパスワード文字

サポートされていないパスワード文字があります。ServiceNow パスワードの複雑さの要件に基づいて、ユーザーはこれらの文字を使用できなくなります。

安全なネットワーク環境を有効にするには、文字、数字、および記号を組み合わせた強力なパスワードを使用する必要があります。このような組み合わせを使用することで、権限のないユーザーが主に手動または自動化された方法で脆弱なパスワードを推測することが起こりにくくなります。

- パスワードポリシー要件は、現代の言語すべての文字を含む Basic Multilingual Plane (BMP) に基づいています。ServiceNow インスタンスには、約 10,000 文字の BMP が付属しています。
- この許容 BMP 内のパスワード文字は、インスタンスに設定できます。この BMP 内にないパスワード文字は許可されません。

注: パスワードに使用できないユーザーまたは会社固有の文字は、除外パスワードリストで設定できます。

パスワードポリシーのプロパティの詳細については、「パスワードポリシープロパティ」を参照してください。

パスワードリセット

デフォルトのセルフサービス パスワードリセット プロセスでは、ユーザーはサービスデスク担当者から支援を受けずにパスワードをリセットできます。

Example: デフォルトのセルフサービスパスワードリセットフロー

1. ユーザーがパスワードを忘れた場合、ユーザーはログイン画面で [パスワードを忘れた場合] リンクをクリックします。
2. パスワードリセット アプリケーションが起動します。[識別] ページで、ユーザーは [ユーザー名] を入力して自分自身を識別します。
3. [検証] ページで、ユーザーは、自分がユーザー名に関連付けられた人物であることを証明します。この例では、ユーザーはユーザープロファイルに関連付けられたメール アドレスを入力します。admin は別の検証方法を設定することができます。または、ユーザーだけが回答できる個人的な質問などの追加の検証を義務付けることができます。
4. [リセット] ページで、メールで手順を確認するようユーザーに指示します。
5. ユーザーはメールを開き、パスワードをリセットするための [ここ] リンクをクリックします。リンクは、ユーザーが指定した期間有効です (**password_reset.request.expiry** プロパティを使用します)。
6. [パスワードのリセット] ページで、パスワードをリセットするための案内がユーザーに表示されます。

デフォルトのセルフサービス パスワードリセット プロセス (com.glideapp.password_reset) では次のものを定義します。

- ユーザーが [パスワードを忘れた場合] をクリックしたときのリダイレクト先を指定する URI。デフォルトでは、この値は **/$pwd_reset.do?sysparm_url=ss_default** です。これは *glide.security.password_reset.uri* プロパティで使用されている値と同じです。以前のリリースでは、この値は **/reset_password.do** に設定されていました。
- [パスワードリセット **URL** を有効にする] オプション。ユーザーが [パスワードを忘れた場合] オプションをクリックした後に、パスワードをリセットするためのリンクを含むメールを受信することを指定します。
- パーソナル データ - 3 段階のパスワードリセットフローを指定するメールアドレス入力検証フロー。

このフォームにアクセスしてフィールドを設定する方法については、「パスワードリセットプロセスの設定」を参照してください。

注:

- この機能は、ローカルで認証されたユーザー (ユーザーレコードで指定されたユーザー名とパスワードを入力したユーザー) の場合に有効です。SSO ソリューションまたは LDAP 統合を使用してインスタンスにログインしているユーザーは、セルフサービス パスワードリセット プロセスの例を使用してパスワードをリセットすることはできません。
- エンドユーザーは通知設定を有効にして設定する必要があります。「サブスクリプションベースの通知」を参照してください。アドミニストレーターは、エンドユーザーに送信されるメールを変更することができます。

パスワードリセット 通知メールテキストの変更

セルフサービスの パスワードリセット プロセスのユーザーは、パスワードのリセットを要求するとメール通知を受け取ります。メールのテキストや通知のその他の要素を変更できます。

始める前に
必要なロール：admin

このタスクについて
このプロセスは、ユーザーがサブスクリプションベースの通知 である場合にのみ該当します。

手順

1. 移動先 すべて > システム通知 > 通知.
2. [パスワードリセット **URL]** 通知を選択します。
3. [内容 **(What it will contain)]** セクションでメールのテキストを変更します。
   通知のその他の側面の設定に関する詳細については、「メール通知を作成する」を参照してください。

パスワードリセット プロパティの設定

エンド ユーザーのパスワードリセットのエクスペリエンスを設定するプロパティを指定できます。

始める前に
必要なロール：password_reset_admin

このタスクについて
プロパティに入力できる値の範囲に制限はありませんが、1 以上の正の整数値のみを使用することをお勧めします。プロパティの上限範囲を決めるときは、ユーザーがどのようなタスクを実行するのかを考慮してください。

たとえば、ユーザーの本人確認の試行を 100 回許可することは通常は望まれません。一般的に許可する試行回数の値は 3 回です。同様に、登録プロセスを完了するために、時間をかけてセキュリティの質問を 30 個選択して回答するようユーザーに強制することも、通常は望まれません。一般的に使用するセキュリティの質問の数は 5 ～ 7 個です。

手順

1. 移動先 すべて > パスワードリセット > プロパティ.
   パスワードリセットプロパティの詳細については、次を参照してください。 パスワードリセット グローバルプロパティ .
2. 必要に応じて設定を更新し、[保存] をクリックします。

記憶する

[記憶する] チェックボックスがログイン時に選択されている場合、ユーザーのコンピューターに cookie が保存されます。この cookie は、その後のアクセス時にユーザーを自動的に認証します。

ユーザーがログアウトすると、cookie は破棄されます。[記憶する] チェックボックスのデフォルト値は 1 つのプロパティで制御されます。チェックボックスがログインページに表示されるかどうかは別のプロパティで制御されます。

2 つのプロパティ *glide.ui.user_cookie.life_span_in_days* および *glide.ui.user_cookie.max_life_span_in_days* は、glide_user システムによって生成された cookie の有効期限の値を制御します。ユーザーが「記憶する」が有効になっているインスタンスにアクセスすると、Cookie の有効期限は最大 (glide.ui.user_cookie.max_life_span_in_days) 有効期限に達するまでになります。

注: これらのプロパティの詳細については、「インスタンスセキュリティ強化設定」の以下のトピックを参照してください。

- 絶対的なセッションタイムアウト時間を最小化する (セキュリティセンター 1.3 で更新)
- セッションウィンドウのタイムアウト時間を最小化する (セキュリティセンター 1.3 で更新)

[記憶する] チェックボックスのデフォルト値を変更する

[記憶する] チェックボックスのデフォルト値を変更することができます。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > システムプロパティ > **UI** プロパティ.
2. *Default value of "Remember me" checkbox on login page* プロパティ (*glide.ui.remember.me.default*) を探します。
3. [記憶する] チェックボックスのデフォルト値を [いいえ] に設定するには、このプロパティのチェックボックスをオフにします。
4. [記憶する] チェックボックスのデフォルト値を [はい] に復元するには、プロパティのチェックボックスを選択します。

[記憶する] チェックボックスを削除する

[記憶する] チェックボックスを削除して、ユーザーがこの機能にアクセスできないようにすることができます。

始める前に
必要なロール：security_admin

注: このプロパティの詳細については、「インスタンスセキュリティ強化設定」の「[記憶する] の削除」を参照してください。

手順

1. ロールを security_admin に昇格させます。
2. 移動先 システムプロパティ > **UI** プロパティ.
3. *Remove "Remember me" checkbox from login page* プロパティ (*glide.ui.forgetme*) を探します。
4. プロパティのチェックボックスを選択します。

この設定により、 [記憶する] チェックボックスが削除され、既存の cookie が無効になり、記憶する機能が完全に無効になります。

5. [記憶する] チェックボックスをログインページに復元するには、プロパティのチェックボックスをクリアします。

ログアウト確認プロンプトを設定する

ログアウト確認プロンプトを有効にして、ユーザーが誤ってログアウトしないようにすることができます。

始める前に

必要なロール：admin

このタスクについて

注: 次の手順は、最新かつ最も一般的に使用されている コア UI より前の UI バージョンでのみ機能します。

手順

1. 移動先 すべて > システムプロパティ > システム.
2. ログアウト要求を確認するメッセージをユーザーに表示するプロパティを見つけて、チェックボックスをオンにします。
3. ユーザーが [ログアウト] ボタンをクリックすると、確認ダイアログボックスが表示されます。

ノンスを実装する

シングルサインオン Digest Authentication で使用するノンスを実装することができます。

Single Sign-on の暗号化されていないトークン方式または暗号化されたトークン方式でノンスを使用するには、これらの手順をいくつかの小さな変更でのみ適用します。

注: ノンスはログイン要求にのみ使用され、他のタイプの要求には使用されません。ログイン後にノンス値を受信した場合、ノンスは消費されません。

利点

ノンスを使用すると、悪意のあるユーザーがシステムにログインするためにリプレイ攻撃を実行できなくなります。

ノンスプロセスフロー

ダイジェストトークンのシングルサインオンを実装し、ノンスのセキュリティを追加する場合は、特定のプロセスフローに従います。

1. ユーザーが顧客のポータルにログインします。
2. 顧客は必要な SSO パラメーターを生成し、末尾にランダムなノンスを追加します。たとえば、顧客がクエリー文字列を使用して認証応答を転送している場合は、次のようになります。

```
SM_USER=itil&DE_USER=V1QuWMmxSfBgfRS099X0cAjKo5Q=&NONCE=1407743018
```

インスタンスはこの要求を受信し、認証変数を取得します。認証応答の整合性を検証する前に、インスタンスは内部テーブル (u_authentication_nonce) に対してノンスを確認し、そのノンスがまだ存在しないことを確認します。そのテーブルにノンスが存在しない場合は、ノンスがテーブルに追加され、認証プロセスが許可されます。そのノンスの値がすでにテーブルに存在する場合、認証の試行はキャンセルされ、failed_missing_requirement というエラーコードが返されます。これにより、通常、ユーザーはログインページに戻ります。

ノンスを実装する

暗号化ノンスを認証ヘッダーに追加して、1 回のみ使用できるようにします。

- *glide.authenticate.header.nonce_key* という名前のシステムプロパティを作成し、その値を NONCE や NCE など、ノンスに使用している変数名に設定します。
- u_authentication_nonce という名前の新しいテーブルを作成します。u_nonce という名前のテーブルにフィールドを追加します。
- 検索項目 システムプロパティ > インストレーションイグジット ExternalAuthentication (*glide.authenticate.external_property*) をオーバーライドする DigestSingleSignOnNonce という項目を作成します。
- 新しく作成した DigestSingleSignOnNonce のスクリプト部分に次のコードを追加します。

```
gs.include("PrototypeServer");

var DigestSingleSignOnNonce = Class.create();
DigestSingleSignOnNonce.prototype = {

 process : function() {

  var headerKey = GlideProperties.get("glide.authenticate.header.key", "SM_USER");
  var headerDigestKey = GlideProperties.get("glide.authenticate.header.encrypted_key",
"DIGEST");
  var headerNonceKey = GlideProperties.get("glide.authenticate.header.nonce_key", "NCE");
  var fieldName = GlideProperties.get("glide.authenticate.header.value", "user_name");
  var fkey = GlideProperties.get("glide.authenticate.secret_key");

  // Look in the Headers
  var data = request.getHeader(headerKey);
  var encdata = request.getHeader(headerDigestKey);
  var nonce = request.getHeader(headerNonceKey);

  // If not, then check the URL Parameters
  if (data == null || encdata == null || nonce == null) {
   data = request.getParameter(headerKey);
   encdata = request.getParameter(headerDigestKey);
   nonce = request.getParameter(headerNonceKey);
  }

  // then maybe its a cookie
  if (data == null || encdata == null || nonce == null) {
   var cookies = request.getCookies();
   data = GlideCookieMan.getCookieValue(cookies, headerKey);
   encdata = GlideCookieMan.getCookieValue(cookies, headerDigestKey);
   nonce = GlideCookieMan.getCookieValue(cookies, headerNonceKey);
  }

  // if found run encryption
  if (data != null && encdata != null && nonce != null) {
   try {

    // Replace all spaces with plus(+)'s, converted in url
    encdata = encdata.replaceAll(' ', '+');

    // ----- Encrypt the username|nonce
```

```
  var key = this.getDigest( data + "|" + nonce, fkey);

  // Check for match of received encoded data
  // and your encoding of user name
  if (encdata == key) {
   var ugr = new GlideRecord("sys_user");
   ugr.initialize();
   if (!ugr.isValidField(fieldName)) {
    GlideLog.warn("External authorization is set to use field: '"+ fieldName + "' which doesn't
exist");
    return "failed_missing_requirement";
   }
   ugr.addQuery(fieldName, data);
   ugr.query();
   if (!ugr.next()) {
    var userLoad = GlideUser.getUser(data);
    if (userLoad == null)
     return "failed_authentication";

    ugr.initialize();
    ugr.addQuery(fieldName, data);
    ugr.query();
    if (!ugr.next())
     return "failed_authentication";

   }

   if (this.processNonce(nonce)){
    var userName = ugr.getValue("user_name");
    return userName;
   }
   else return "failed_missing_requirement";
   }
  else {

   return "failed_authentication";
  }
 } catch(e) {
  gs.log(e);
  return "failed_authentication";
 }
 // Encoded data didn't match recieved Encoded data
} else {

 return "failed_missing_requirement";
}
},

getDigest : function( data, fkey ) {
 try {
  // default to something JDK 1.4 has
  var MAC_ALG = "HmacSHA1";
  return SncAuthentication.encode(data, fkey, MAC_ALG);

 } catch (e) {
  gs.log(e.toString());
  throw 'failed_missing_requirement';
```

自動翻訳

```
 }
} ,

processNonce : function( sentNonce ) {
 var ngr = new GlideRecord("u_authentication_nonce");

 ngr.addQuery("u_nonce", sentNonce);
 ngr.query();
 if (ngr.next()) {
  gs.log("This SSO entry has already been processed! (Nonce: " + sentNonce + ")");
  return false;
 }
 var ngrNew = new GlideRecord("u_authentication_nonce");
 ngrNew.initialize();
 ngrNew.u_nonce = sentNonce;
 ngrNew.insert();
 gs.log("Inserted new nonce: " + sentNonce);
 return true;
}
};
```

- 新しいインストレーションイグジットを保存したら、DigestSingleSignOn インストレーションイグジットに移動し、Active=false に設定されていることを確認します。

これで、インスタンスがノンスを実装するように設定されているはずです。

多要素認証

マルチファクター認証 (MFA) をアクティブ化、使用、構成する方法について説明します。

探索

MFA の機能とビジネス価値について説明します。

構成

自動翻訳

MFA をアクティブ化する方法を理解します。

使用方法

MFA を使用します。

シングルサインオン (SSO) による MFA

シングルサインオン (SSO) を使用して MFA を構成する方法を理解します。

マルチファクター認証の適用

Yokohama リリースから ServiceNow への非 SSO ログインに対する MFA の適用。

MFA は、ログイン時に ID を証明するための証拠を 2 つ以上提出するようユーザーに要求することでセキュリティを強化します。MFA は、フィッシングやアカウントの乗っ取りなどの脅威からアカウントを保護します。

MFA は、さまざまな ID 乗っ取り関連の攻撃に対する重要なセキュリティツールです。ServiceNow では、Yokohama へのアップグレード後にデフォルトで MFA が適用され、非 SSO ログイン (ユーザー名とパスワードまたは LDAP ベースの認証のみでログインするユーザー) に MFA を必須にすることで、セキュリティ体制を強化し、侵害のリスクを軽減します。

MFA の適用は、デフォルトで Yokohama からアクティブ化されているか、Yokohama にアップグレードされている MFA ポリシーを介して実行されます。

注:

- このポリシーは、ローカル (ユーザー名とパスワード) または LDAP ベースの認証を実行するすべての 非**snc_external** ユーザーに対して有効になります。
- インスタンスアドミンは、MFA コンテキストポリシー、ポリシー基準、またはポリシー条件を変更することで、適用スコープを変更できます。

MFA の適用の詳細については、こちらを参照してください。

- すべての本番インスタンスと非本番インスタンスに適用されます。
- すべての 非**snc_external** ユーザーと 非 **SSO** ログインに適用されます。
- ベーシック認証および OAuth との統合 リソース所有者のパスワード認証情報の付与には、Yokohama からの MFA は必要ありません。

施行による変更の詳細については、「 マルチファクター認証の適用による変更」を参照してください。

MFA の詳細については、「 マルチファクター認証の詳細」を参照してください。

関連トピック

マルチファクター認証の構成

マルチファクター認証の使用

マルチファクター認証の適用プロパティ

マルチファクター認証の適用のトラブルシューティング

よく寄せられる質問 - マルチファクター認証の適用

マルチファクター認証の適用による変更

MFA の適用によって予想される変更に関する情報。

予想される MFA 適用の変更は次のとおりです。

- 統合への影響はありません。
- 既に MFA を登録しているユーザーは、引き続きローカルログイン時に MFA のチャレンジを受けます。
- Yokohama へのアップグレード後、ユーザーがまだ MFA を設定していない場合は、最初は一定期間 MFA の適用対象外となり、その間に MFA を登録できます。
- ユーザーが自己登録に失敗した場合は、免除期間 (デフォルトは 30 日) の完了時に MFA が強制されるため、MFA を設定する必要があります。
- 90 日後は、デフォルトで MFA が適用されます。90歳以降の初回ログインには自己登録期間はありません。アドミニストレーターはこの期間を設定できます。

アドミニストレーターは、以下に基づいて MFA の適用を準備できます。

- MFA コンテキストポリシー構成を確認し、ビジネス要件に従ってポリシー条件を調整します。
- 他に除外する必要があるユーザーがいる場合は、デフォルトで指定された免除グループを追加します。
- MFA 適用プロパティを確認し、要件に基づいて調整します。詳細については、「マルチファクター認証の適用プロパティ」を参照してください。

Example: MFA 適用から 30 日後の MFA 適用シナリオ

- **Acme Corp** のシナリオ **1**:インスタンスで、ユーザーがアクティブな MFA ポリシーを持っていない場合。Sarah がローカル認証を使用してインスタンスにアクセスするとします。ログイン時に Yokohama リリースにアップグレードすると、MFA を登録するメッセージが表示されます。このセットアップの完了までの期間は 30 日です。Sarah がセットアップを完了しない場合、30 日が経過すると、アカウントへのログインに MFA が必要になり、MFA が設定されるまでアクセスできなくなります。

- **Acme Corp** のシナリオ **2**:インスタンスでは、Anita は既にローカル認証とともに MFA になっています。Anita は、30 日間の自己登録期間なしで MFA を引き続き要求します。
- **Acme Corp** のシナリオ **3**:インスタンスでは、Olivia は認証にシングルサインオン (SSO) を使用します。ログインエクスペリエンスへの影響はなく、Olivia は MFA を強制されません。
- **Globex Corp** のシナリオ **4**:インスタンスで、ユーザーが会社の信頼できるネットワーク外でのすべてのローカルログイン試行に対して MFA を要求する MFA ポリシーを既に持っていたとします。Yokohama 以降のリリースにアップグレードしても、ユーザーログインの MFA 強制動作は変わりません。

マルチファクター認証の適用プロパティ

Yokohama から MFA を適用するためのプロパティを設定し、Yokohama にアップグレードします。

MFA 適用プロパティを設定するには、 **sys_properties.list** に移動して MFA 適用プロパティを検証または変更します。

MFA 適用プロパティ

フィールド	説明
glide.authenticate.multifactor.setup.bypass.count	ユーザーが多要素認証の設定をバイパスできる回数 (バイパス可能な最大数は 3 です。これを超える回数は 3 として扱われます)。
glide.authenticate.multifactor.self_enrollment_period	ユーザーに MFA を自分で登録するオプションが与えられる日数を示します。日数を過ぎると、ユーザーに自動的に MFA の要求が表示されます。90 を超える値は、90 日としてのみ扱われます。
glide.authenticate.multifactor.enforcement_max_period	新規ユーザーに MFA を自分で登録するオプションが与えられる最大日数を示します。この日数を過ぎると、SSO 以外のログインを実行するすべての新規ユーザーに自動的に MFA の要求が表示されます。270 を超える値は、270 日としてのみ扱われます。
glide.authenticate.multifactor.enforcement.debug	MFA 適用デバッグロガー。MFA ログインのフローのデバッグに役立ちます。 **注:** このプロパティはインスタンスには存在しません。必要に応じて、プロパティを作成して有効にする必要があります。
glide.authenticate.hybrid_user_tracking.enabled	ハイブリッドユーザーを追跡するためのプロパティ。対応するsys_userレコードで「Web サービスへのアクセスのみ」とマークされていないが、ユーザー名とパスワードを使用して統合 (API ログインなど) を実行するユーザーアカウントは、このプロパティを有効にすると [ユーザーログイン情報] テーブルで追跡されます。 **注:** このプロパティはインスタンスには存在しません。必要に応じて、プロパティを作成して有効にする必要があります。

MFA 適用プロパティ (続く)

フィールド	説明
glide.authenticate.hybrid_user_tracking.debug	ハイブリッドユーザー API ログインを追跡するためのデバッグロガー。横 注: このプロパティはインスタンスには存在しません。必要に応じて、プロパティを作成して有効にする必要があります。

マルチファクター認証の適用のトラブルシューティング

MFA の適用によるトラブルシューティング情報

ServiceNow は、Yokohama リリースのアップグレード後にデフォルトで MFA を適用し、非 SSO ログイン (ユーザー名とパスワードまたは LDAP ベースの認証のみでログインを実行するユーザー) に MFA を必須にすることで、セキュリティ体制を強化し、侵害のリスクを軽減します。

MFA の適用は、デフォルトで Yokohama からアクティブ化されているか、Yokohama にアップグレードされている MFA ポリシーを介して実行されます。MFA の動作に変更があった場合に実行できるトラブルシューティングタスクの一部を次に示します。

- トラブルシューティングツールを使用したデバッグ
- ログの場所とデバッグプロパティに移動します
- MFA の使用中のユーザーエクスペリエンスに基づいて MFA シナリオを理解します
- 以前のリリースからのアップグレードによる MFA の問題を理解する

MFA のデバッグ

次のツールのいずれかまたは組み合わせを使用して、デバッグ情報を理解してください。

- **Splunk** - デバッグログを表示します。
- システムログまたはノードログ。
- MFA のデバッグログを分析するための **HAR** ログ。

ログの場所とデバッグプロパティ

ログの詳細については、次の場所に移動してください。

- システムログについては、次に移動します: すべて > システムログ > システムログ.
- ノードログの場合は、 すべて > システムログ > ユーティリティ > ノードログファイルブラウザ.

システムデバッグログとインスタンスノードログは、デバッグのために必要です。有効にする必要があるデバッグプロパティは次のとおりです。

- glide.webauthn.debug.enabled
- glide.log.default_log_debug
- glide.authenticate.policy.debug
- glide.authenticate.hybrid_user_tracking.debug

シナリオに基づく **MFA** の問題

シナリオ 1:ユーザーが第 2 要素を使用してログインできない

ユーザーの MFA をリセットし、次のテーブルから古いユーザーレコードを削除します。

- user_multifactor_auth
- sys_user_public_credential
- sys_user_multi_factor_setup

シナリオ 2:アドミニストレーターが第 2 要素を使用してログインできない

アドミンアクセス権を持つ別のユーザーが、ブロックされたアドミンユーザーの MFA をリセットできます。それでも問題が解決しない場合は、 ServiceNow サポートに連絡してください。

シナリオ 3:MFA セットアップまたは検証中にエラーが発生した

「関連するエラーコード/警告:6桁の確認コードが正しくありません。正しいコードでもう一度お試しください。

次の手順を実行します。

- TOTP 認証アプリの場合、認証システムの MFA デバイスとインスタンスの日時が同期していない場合 (±30 秒)、TOTP コードは受け入れられません。デバイスとインスタンスの日時を確認します。
- メールの場合は、ユーザーレベルの通知、送信メール設定、およびユーザーを sys_user テーブルで正しく構成します。
- SMS の場合は、Twillio またはその他の SMS サービスプロバイダー統合を正しく構成し、アクティブに設定します。ユーザーの携帯電話番号が sys_user テーブルで正しく構成されているかどうかを確認します。

よく寄せられる質問 - マルチファクター認証の適用

MFA の適用が期限となる一部の FAQ の詳細。

MFA は、さまざまな ID 乗っ取り関連の攻撃に対する重要なセキュリティツールです。ServiceNow では、Yokohama へのアップグレード後にデフォルトで MFA が適用され、非 SSO ログイン (ユーザー名とパスワードまたは LDAP ベースの認証のみでログインを実行するユーザー) に MFA が必須になります。

MFA の適用は、デフォルトで Yokohama からアクティブ化されているか、Yokohama にアップグレードされている MFA ポリシーを介して実行されます。MFA の適用に関連する質問の詳細については、次のリンクを参照してください。

- MFA 強制の例外
- MFA の適用要件 – 内容と理由
- MFA 適用範囲
- MFA 適用タイムライン
- MFA メトリクス
- MFA タイプ
- MFA リセット

MFA の適用要件 – 内容と理由

MFA の適用とそれが重要である理由に関連する FAQ。

1. MFA とは何ですか?

   マルチファクター認証 (MFA) は、アカウントまたはシステムにアクセスする前に、2 つ以上の形式の検証を提供する必要があるセキュリティプロセスです。詳細については、「マルチファクター認証の詳細」を参照してください。

2. MFA の適用が義務付けられているのはなぜですか?

   MFA は、アカウントとデータのセキュリティを保護するために義務付けられています。サイバー脅威は絶えず変化しており、パスワードだけでは不正アクセスに対する十分な保護を提供できなくなっています。

   - MFA が有効になっている場合、攻撃者がパスワードを知っていても、攻撃者は 2 番目の形式の検証が必要です。この追加レイヤーにより、ほとんどの不正な試行が大幅にブロックされ、情報の安全性が向上します。
   - MFA をデフォルトとして設定することで、セキュリティ侵害のリスクを最小限に抑え、アカウントを自動的に保護します。つまり、セキュリティに関する追加の決定を行うことなく、安心感を高めることができます。

3. MFA を有効にすることが重要なのはなぜですか?

   MFA を有効にすると、アカウントのセキュリティが強化されます。パスワードはデータ侵害で漏洩する可能性があるため、パスワードだけでは不十分です。MFA では、誰かがあなたのパスワードを知っていても、2 番目の検証手順を経ないとアカウントにアクセスできません。

4. ServiceNow に MFA が必要な理由

   ServiceNow は、これらの脅威からユーザーを保護するために MFA を義務付けています。これは、不正アクセスを減らすためのシンプルかつ効果的な方法です。MFA を要求することで、すべてのアカウントに強力な保護レイヤーが用意され、あなたとすべてのユーザーのセキュリティリスクが軽減されます。

5. 既存の顧客の MFA 要件は何ですか?

   インスタンスを Yokohama 以降のリリースにアップグレードする既存のお客様の場合:

   - インスタンスで 適応認証 – マルチファクター認証コンテキスト がまだオンになっていない場合は、デフォルトの MFA ポリシーとして自動的に有効になります。
   - ローカル認証または LDAP 認証を使用してログインするすべての内部ユーザー (snc_external ロールを持たないユーザー) は、最初のログイン成功から 30 日以内に MFA を設定する必要があります。この間は通常どおりログインできますが、ログイン時に MFA を登録するメッセージが表示されます。
   - 30 日が経過すると、デフォルトで MFA が要求され、ユーザーは MFA セットアップを完了しないとログインできなくなります。

6. 新規顧客の MFA 要件は何ですか?

   Yokohama リリース以降を使用しているインスタンスでは、すべての内部ユーザーに対して MFA がデフォルトで有効になっています。また、 `snc_external` ロールを持たず、ローカル認証または LDAP 認証でログインしているユーザーにも適用されます。ユーザーは、最初のログインから MFA を設定して使用する必要があります。

MFA 適用範囲

MFA の適用範囲とその重要性に関する FAQ。

1. MFA が必要なユーザー、ログイン、環境タイプはどれか?

   Yokohama リリース以降、次のシナリオで新しいデフォルトの安全な MFA ポリシー MFA が適用されます。

   - **snc_external**ロールを持つユーザーを除くすべてのユーザー。
   - ユーザー名とパスワードに基づくローカル認証または Lightweight Directory Access Protocol (LDAP) 認証を実行するすべてのユーザー。
   - 本番インスタンス、準本番インスタンス、テストインスタンスを含め、アップグレード前にアクティブな MFA ポリシーがまだなかったすべての顧客インスタンス。

   インスタンスアドミンは、MFA コンテキストポリシー、ポリシー基準、またはポリシー条件を変更することで、適用スコープを変更できます。

2. シングルサインオン (SSO) ログインに MFA は必要ですか?

   いいえ。デフォルトの安全な MFA ポリシーでは、SSO (SAML、OIDC、証明書ベースの認証) ログインに MFA は必要ありません。

   お客様は、シングルサインオン (SSO) プロバイダー (ID プロバイダーまたは IdP) と協力して、IdP 側でマルチファクター認証 (MFA) を適用できます。IdP 側で MFA を適用できない場合は、「シングルサインオンによるマルチファクター認証」に記載されている手順に従って、SSO ログインに対して ServiceNow プラットフォームの MFA を有効にすることもできます。

3. 外部ユーザーには MFA が必要ですか?

   いいえ。デフォルトの安全な MFA ポリシーでは、snc_external ロールを持つユーザーに MFA は必要ありません。

   - アドミニストレーターは、MFA ポリシー条件を更新することで、この動作を変更し、外部ユーザーに MFA を適用できます。
   - Yokohama 以降のリリースへのアップグレード前に既に MFA を受けていた外部ユーザーは、引き続き MFA を利用できます。
   - 外部ユーザーは自分のプロファイルにアクセスして、MFA を自分で登録できます。

4. モバイルアプリのログインに MFA は必要ですか?

   要。MFA ポリシーは、Web アプリとモバイルアプリの両方で、ユーザー名とパスワードベースの非 SSO ログインでログインする場合に適用されます。

5. 非本番環境とテスト環境に MFA は必要ですか?

   要。Yokohama 以降のバージョンにアップグレードする前にインスタンスにアクティブな MFA ポリシーが存在していない場合、MFA は本番環境、非本番環境、開発環境、テスト環境を含むすべての顧客インスタンスに適用されます。

6. 開発者インスタンスに MFA は必要ですか?

   要。MFA は、Yokohama 以降のリリースバージョンのすべての開発者インスタンスに適用されます。

7. API 認証には MFA が必要ですか?

   いいえ。Yokohama 以降のリリースでは、MFA はユーザー名とパスワードベースのインタラクティブなユーザーログインにのみ必要です。これは、MFA を必要とせずに機能するベーシック認証を使用した API 認証を意味します。OAuth や mTLS などの代替のセキュア API 認証方法を使用することをお勧めします。詳細はこちら。

a. クローン

b. 更新セットの取得

c. RPA

8. MFA によるクローンセットアッププロセスへの影響はありますか?

   いいえ。クローンセットアッププロセスは引き続きユーザー名とパスワードで動作し、MFA は必要ありません。

9. MFA による更新セットの取得に影響はありますか?

   いいえ。更新セットの取得はユーザー名とパスワードで引き続き機能し、MFA は必要ありません。

10. ServiceNowインスタンスにアクセスする RPA ボットに影響はありますか?

    はい。RPA ボットがユーザー名とパスワードの対話型ログインを使用して ServiceNow インスタンスにアクセスする場合は、MFA を実行する必要があります。アドミニストレーターは、RPA ボットアカウントの MFA を緩和する場合、RPA ボットアカウントを **MFA** 適用対象外ユーザーグループ に追加できます。

11. OAuth ベースの統合には MFA が必要ですか?

    OAuth リソース所有者のパスワード認証情報 (ROPC) は、MFA を必要とせずにユーザー名とパスワードを使用して機能します。認証コード権限許可タイプの場合、OAuth 同意を与える前に、ユーザーログインフローの一部として MFA が必要です。

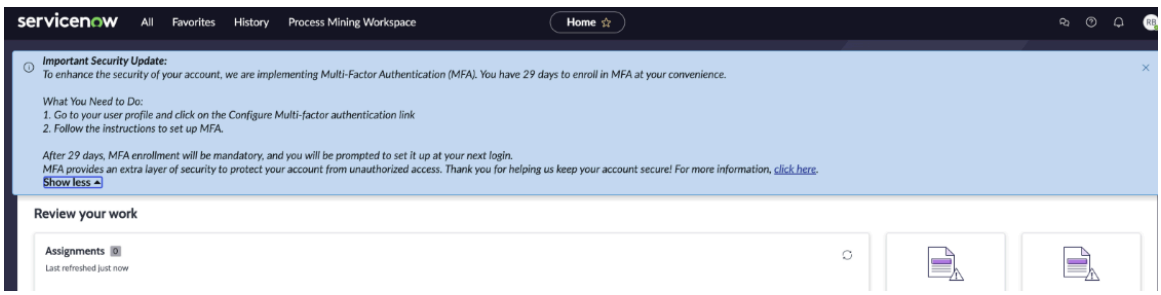
MFA 適用タイムライン

MFA の適用タイムラインとその重要性に関連する FAQ。

1. MFA はいつ強制されますか?

   MFA ポリシーによると、MFA セットアップを完了していない対象ユーザーには、30 日間の自己登録期間があります。この動作は、システムプロパティ glide.authenticate.multifactor.self_enrolment_period を使用して制御されます。プロパティのデフォルト値は 30 日です。最大 90 日間まで更新できます。

   ローカル認証または LDAP 認証を使用してログインするすべての内部ユーザー (snc_external ロールを持たないユーザー) は、最初のログイン成功から 30 日以内に MFA を設定する必要があります。この間は通常どおりログインできますが、ログイン時に MFA を登録するメッセージが表示されます。



   Yokohama 以降のリリースにアップグレードしてから 90 日後、内部ユーザー (snc_external ロールを持たないユーザー) がローカル認証または LDAP 認証を使用して初めてログインした場合、すぐに MFA を使用する必要があります。30 日間の MFA 自己登録期間がない。この期間は、システムプロパティ glide.authenticate.multifactor.enforcement.max_relaxation_period によって管理されます。このプロパティの最大値は 270 日です。

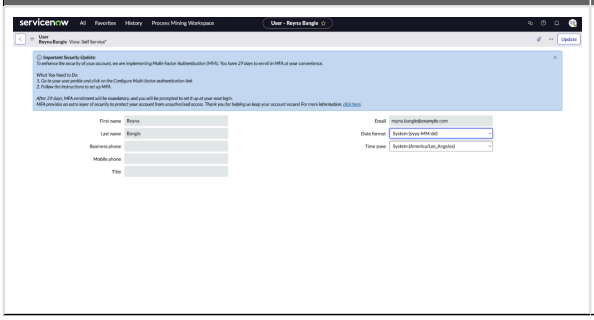
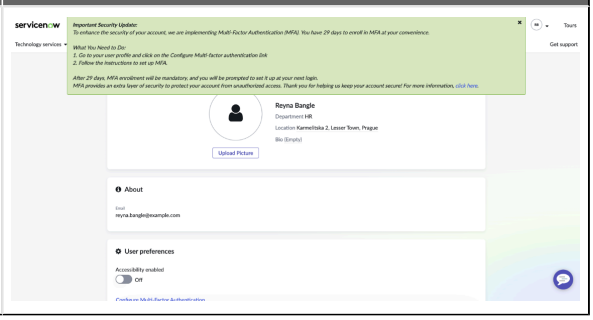
2. MFA の適用タイムラインをどのように調整できますか?

- プロパティ glide.authenticate.multifactor.self_enrolment_periodの値を更新することで、アドミニストレーターは自己登録期間を縮小または拡大できます。プロパティ値を 0 に設定します。ユーザーは、Yokohama 以降のリリースにアップグレードした後、ローカルログインまたは LDAP ログインで最初のログインを試行した後、MFA セットアップを完了する必要があります。自己登録期間の最大期間は 90 日間です。90 より大きいプロパティ値は 90 として扱われます。
- プロパティ glide.authenticate.multifactor.enforcement.max_relaxation_period の値を更新することで、アドミニストレーターは Yokohama 以降のリリースへのアップグレード後何日後に MFA 自己登録期間を取得するかを決定できます。

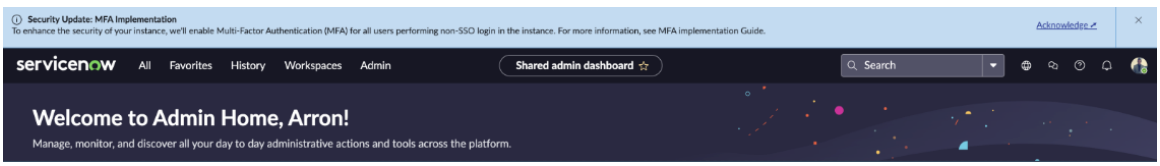
3. この今後の変更についてエンドユーザーにどのように通知されますか?

MFA が強制されるローカル認証または LDAP 認証を実行するエンドユーザーは、ログイン後に情報メッセージが表示されます。ユーザーが自分のプロファイルにアクセスしたときにも、同じメッセージが表示されます。

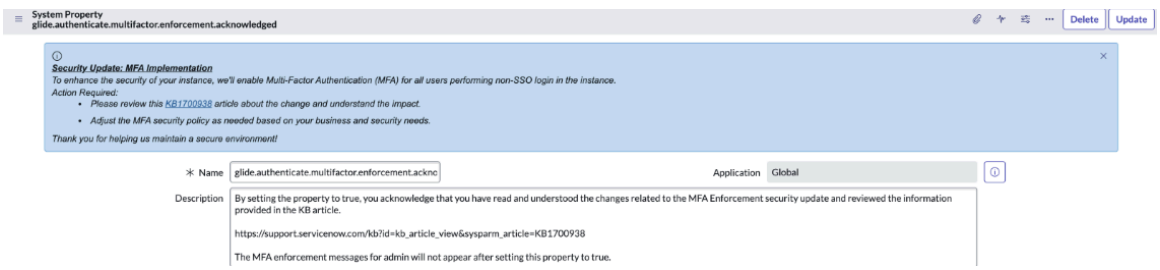
適用メッセージ

ユーザープロファイル	従業員サービスセンター **(ESC)** 上
	

このメッセージは、SSO ログインを実行しているアドミン以外のユーザーには表示されません。ログインに使用された認証方法に関係なく、ログインが成功すると、admin ロールには異なる情報メッセージが表示されます。



このメッセージは、アドミンの 1 人が glide.authenticate.multifactor.enforcement.acknowledged プロパティ値を true に設定して更新を確認するまで表示され続けます。



4. エンドユーザーがログインしたときに、MFA セットアップの完了に関するメッセージをオフにする方法は?

アドミニストレーターは、 glide.authenticate.multifactor.enforcement.show_user_info_message システムプロパティの値を false に更新して、ログイン後にエンドユーザーに表示される MFA 登録情報メッセージをオフにすることができます。

自動翻訳

5. MFA の適用に関してアドミニストレーターに表示されるメッセージをオフにする方法は?

   ログイン後にアドミンユーザーに表示される MFA の適用に関する情報メッセージは、アドミンの 1 人が glide.authenticate.multifactor.enforcement.acknowledged システムプロパティの値を true に更新して確認すると表示されなくなります。

6. インスタンス内の組織のセキュリティニーズに基づいて、適応認証を使用して定義された MFA ポリシーが既に存在します。ポリシーは義務の影響を受けますか?

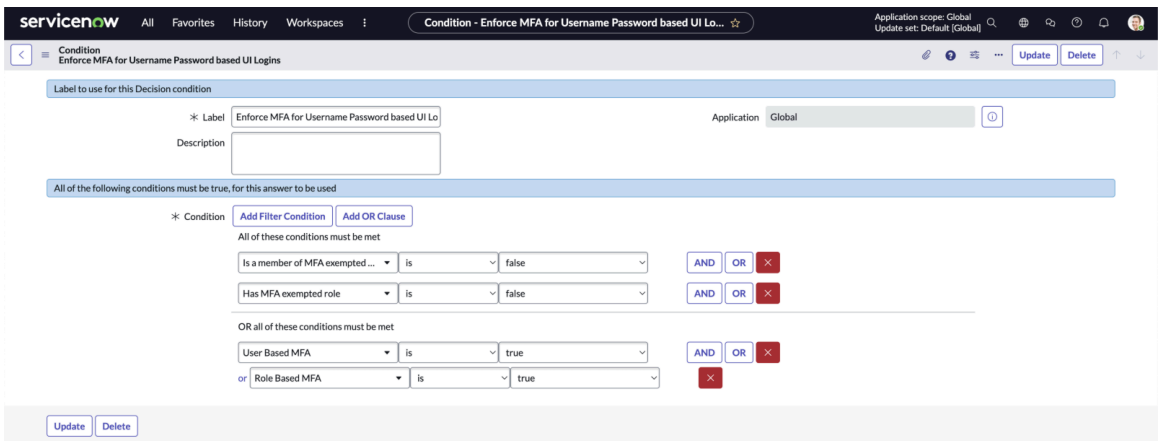
   いいえ。インスタンスに既にアクティブな適応認証 - MFA コンテキストポリシーがある場合、新しいデフォルトの安全な MFA ポリシーは適用されません。インスタンスで MFA プロパティが有効になっている (glide.authenticate.multifactor) が、MFA ポリシーがアクティブではない場合、ユーザー名とパスワードベースのローカルログインまたは LDAP ログインを実行するすべての内部ユーザー (snc_external ロールを持たないユーザー) に MFA を適用するための、デフォルトの安全な MFA ポリシーが有効になります。

MFA 強制の例外

MFA 強制の例外に関連する FAQ と、それが重要である理由。

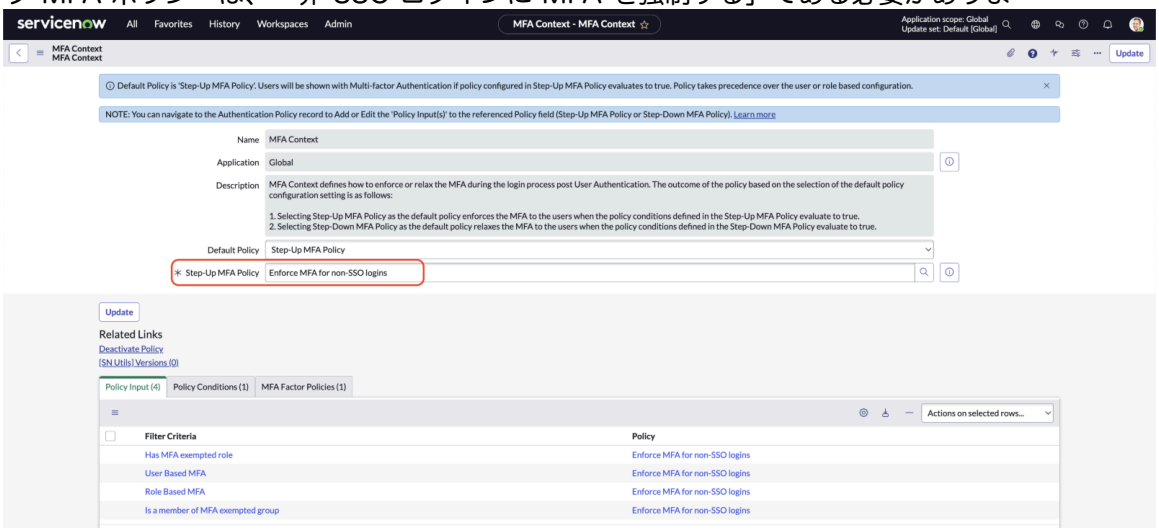
1. 特定のユーザーに対してMFAの義務を緩和するにはどうすればよいでしょうか?

   Yokohama リリースでは、新しいユーザーグループである MFA 適用対象外ユーザーグループレコードが追加されています。デフォルトの条件に基づいて、MFA ポリシーが追加され、このグループのメンバーであるすべてのユーザーに MFA が適用されます。



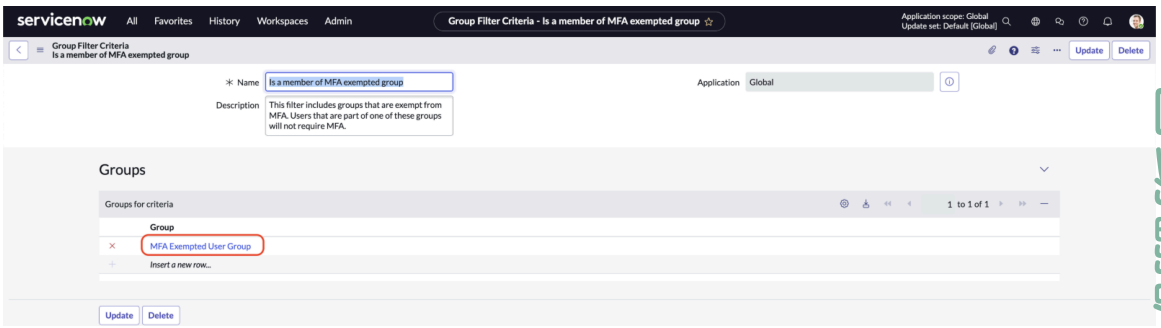
   特定のユーザーの MFA を緩和するには、次の手順に従います。

- **MFA** コンテキストに移動します。MFA コンテキストレコードに関連付けられたステップアップ MFA ポリシーは、「非 SSO ログインに MFA を強制する」である必要がありま



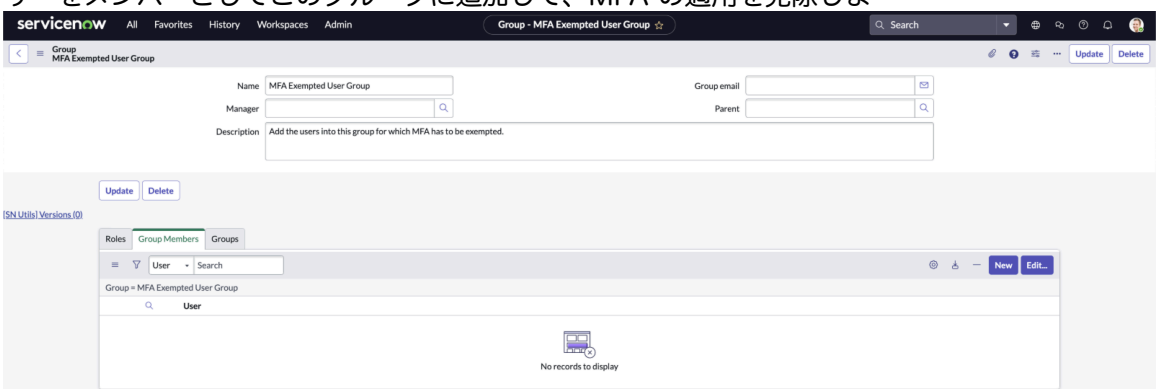
す。

- [ポリシーの入力] 関連リストで、[**MFA** 適用対象外グループのメンバーである] フィルター基準レコードを選択します。
- **MFA** 適用対象外ユーザーグループを選択しま



す。

- ユーザーをメンバーとしてこのグループに追加して、MFA の適用を免除しま

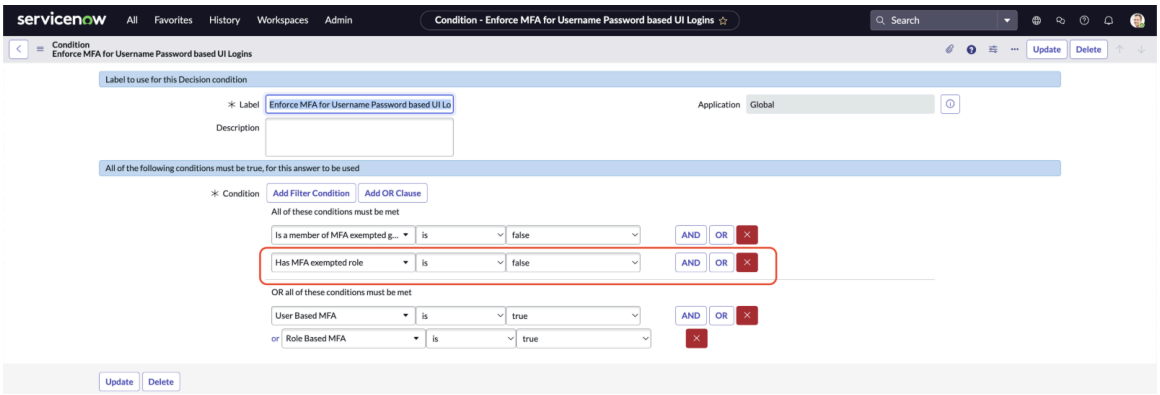


す。

> **注:** MFA コンテキストに別のポリシーが関連付けられている場合は、「MFA 適用対象外グループのメンバー」フィルター基準をポリシーに追加し、このグループのユーザーを MFA の適用から除外するようにポリシー条件を変更できます。

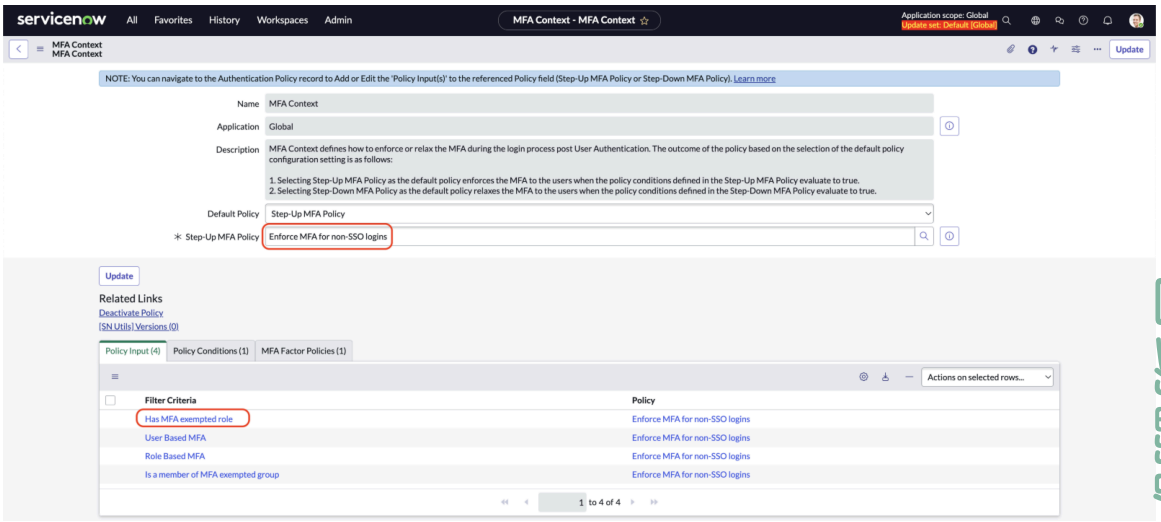
2. 特定の役割についてMFAの義務をどのように緩和できますか?

Yokohama リリースでは、空の新しいロール [**MFA** 適用対象外ロールあり] フィルター基準が追加されています。MFA ポリシーには、ロールが免除されたロール基準の一部であるユーザーを MFA の適用から除外するための条件が追加されています。



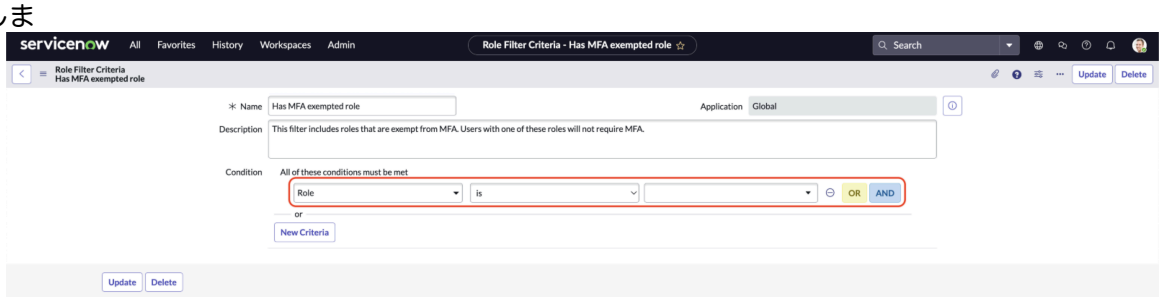
特定のロールの MFA を緩和するには、次の手順に従います。

- **MFA** コンテキストに移動します。MFA コンテキストレコードに関連付けられたステップアップ MFA ポリシーは、「 非 **SSO** ログインに **MFA** を強制する」である必要がありま



す。

- [ポリシーの入力] 関連リストで、[**MFA** 適用対象外ロールフィルター基準あり] レコードを選択しま



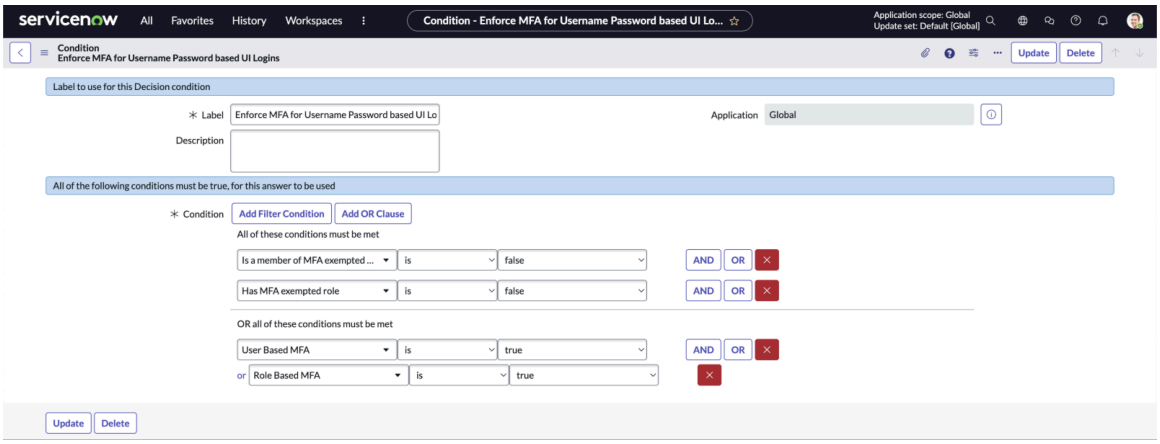
す。

- 条件に追加するロールを追加します。OR 演算子を使用して複数のロールを追加できます。

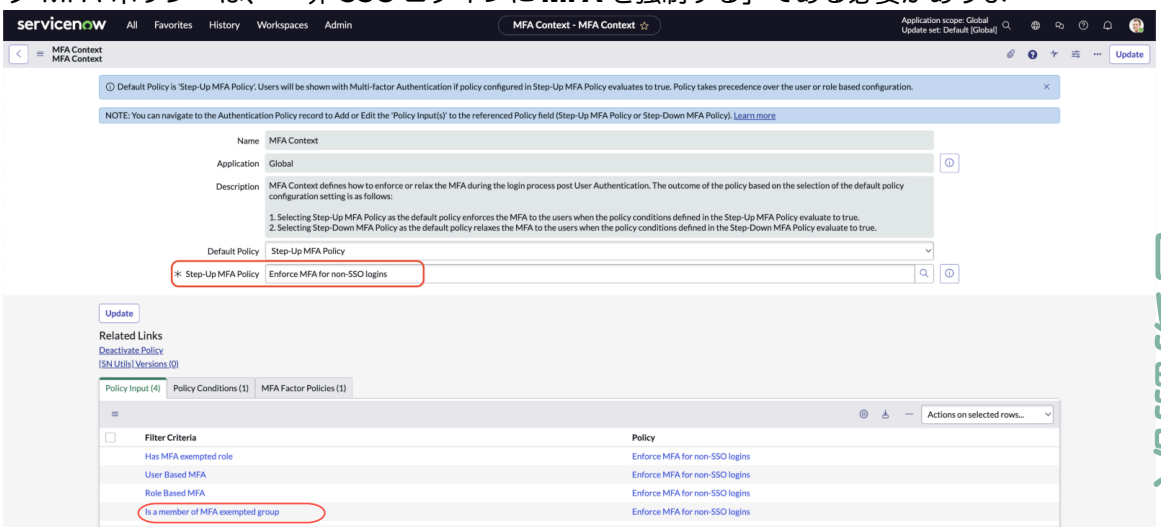
> **注:** MFA コンテキストに関連付けられた別のポリシーがある場合は、[**MFA** 適用対象外ロールあり] フィルター基準をポリシーに追加できます。ポリシー条件を変更して、適用対象外ロールを持つユーザーを MFA の適用から除外します。

3. 特定のグループに対してMFAの義務を緩和するにはどうすればよいですか?

Yokohama リリースでは、 ユーザーグループ **MFA** 適用対象外ユーザーグループ が追加されています。MFA ポリシーに追加されたデフォルトの条件に基づいて、このグループのメンバーであるユーザーまたはグループは MFA で強制されません。

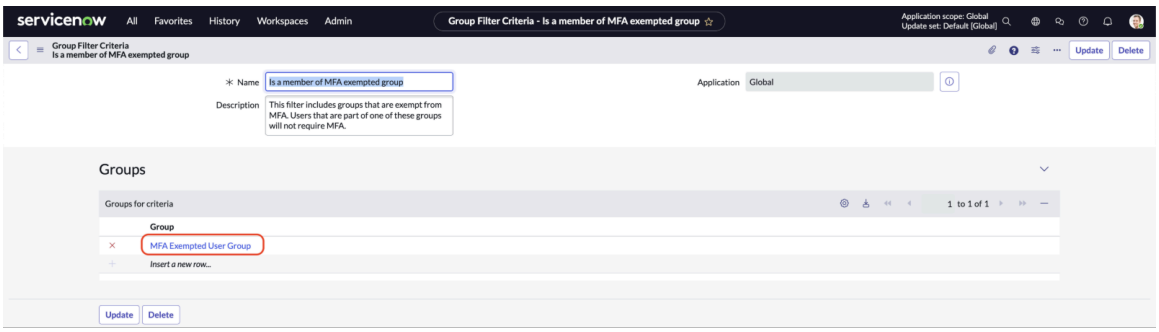


特定のグループの MFA を緩和するには、次の手順に従います。

- **MFA** コンテキストに移動します。MFA コンテキストレコードに関連付けられたステップアップ MFA ポリシーは、「 非 **SSO** ログインに **MFA** を強制する」である必要がありま 

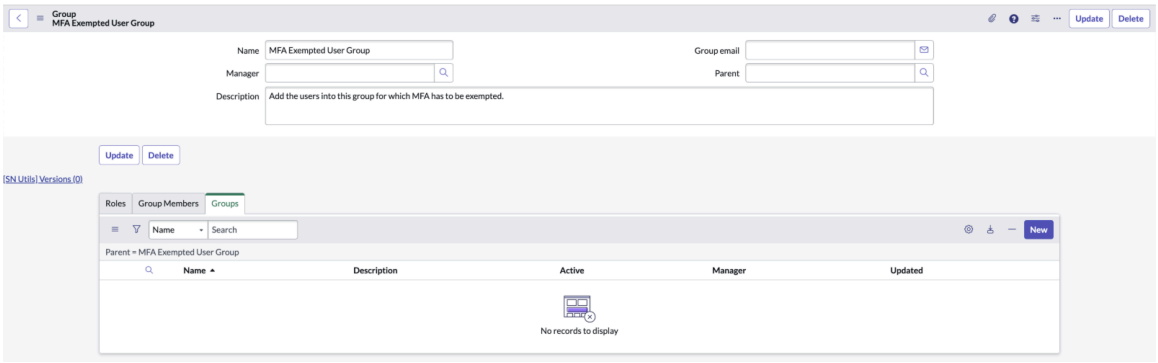
す。
- [ポリシーの入力] 関連リストで、[**MFA** 適用対象外グループのメンバーである] フィルター基準レコードを選択します。

- **MFA** 適用対象外ユーザーグループを選択しま



す。

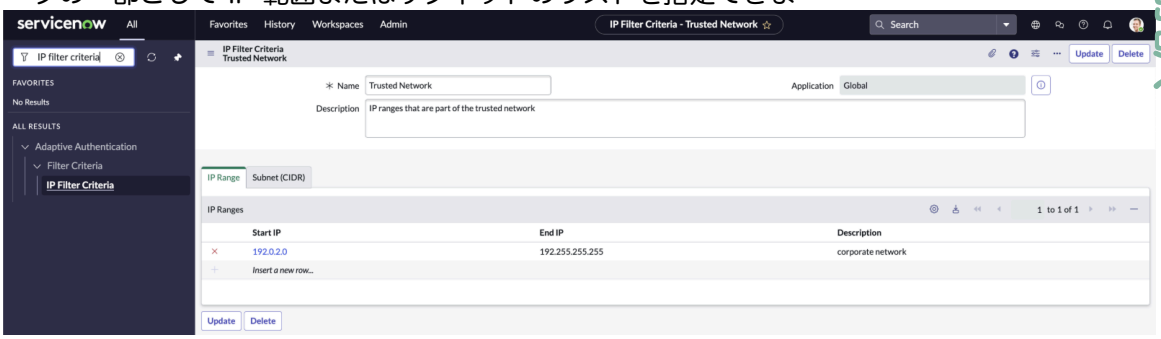
- MFA の適用を除外するグループをこのグループに追加しま



す。

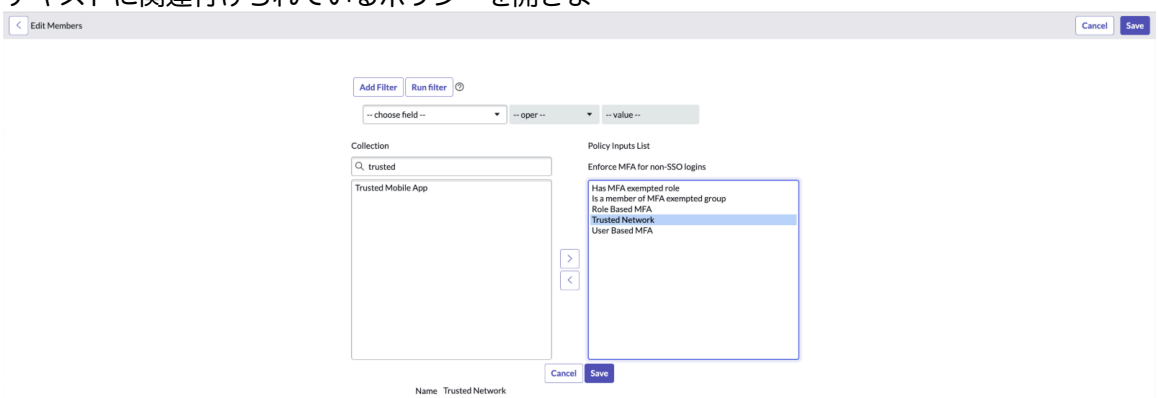
4. 信頼できるネットワークに対するMFAの義務をどのように緩和できますか?

- 移動先 適応認証 > フィルター基準 > **IP** フィルター基準.
- 信頼できるネットワークを指定する基準を作成します。信頼できるネットワークの一部として IP 範囲またはサブネットのリストを指定できま



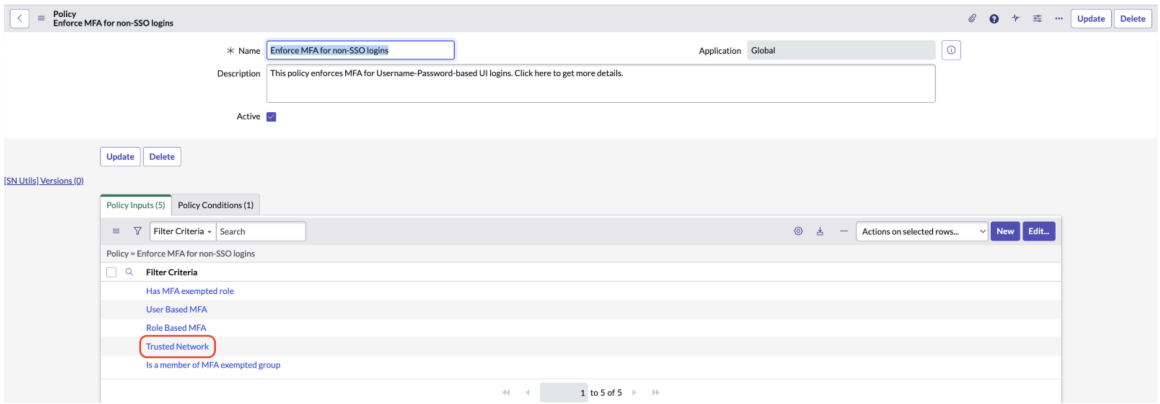
す。

- 移動先 適応認証 > 認証ポリシーのコンテキスト > **MFA** コンテキスト.
- コンテキストに関連付けられているポリシーを開きま

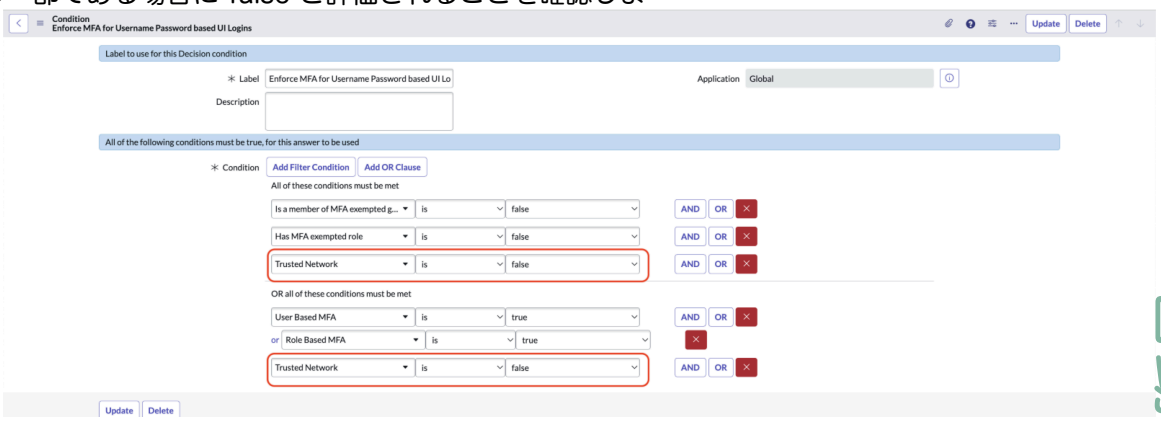


す。

- 編集を選択して、作成した **IP** フィルター基準 を [ポリシー入力] 関連リストに追加しま す。



- ポリシー条件を変更して、ユーザーが信頼できるネットワークの一部である場合に false と評価されることを確認しま す。



> **注:** MFA コンテキストに関連付けられた別のポリシーがある場合は、ステップ 1 の一部として作成された IP フィルター基準をポリシーに追加し、信頼済みネットワークでの MFA の適用を除外するようにポリシー条件を変更できます。

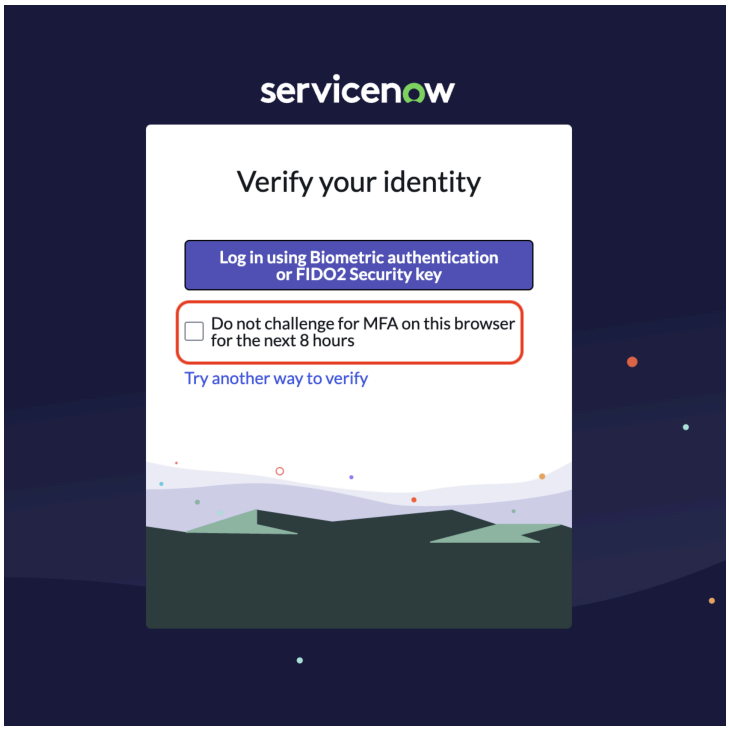
5. 信頼できる場所に対する MFA の義務をどのように緩和できますか?

Zero Trust – Location Based Access (追加のサブスクリプションが必要) プラグインで利用可能な場所フィルター基準を使用できます。

6. 頻繁なMFAの適用を制御する方法は?

Zero Trust – Location-Based Access プラグインで利用可能な場所のフィルター基準を使用します (追加のサブスクリプションが必要です)。

MFA 検証ページには、ブラウザーを記憶するためのチェックボックスがあります。記憶されたブラウザーに MFA が適用されない:



- このシステムプロパティで指定された期間。glide.authenticate.multifactor.browser.fingerprint.validity。プロパティのデフォルト値は 8 時間です。この期間は、最大 24 時間延長できます。同様に、glide.authenticate.multifactor.remember.browser.default システムプロパティを使用して、チェックボックスのデフォルト値を true に設定できます。
- 移動先 多要素認証 > プロパティ をクリックし、これら 4 つのプロパティを調整して、記憶されたブラウザー機能を制御します。

7. ユーザーが共有するアカウントに対して MFA はどのように機能しますか?

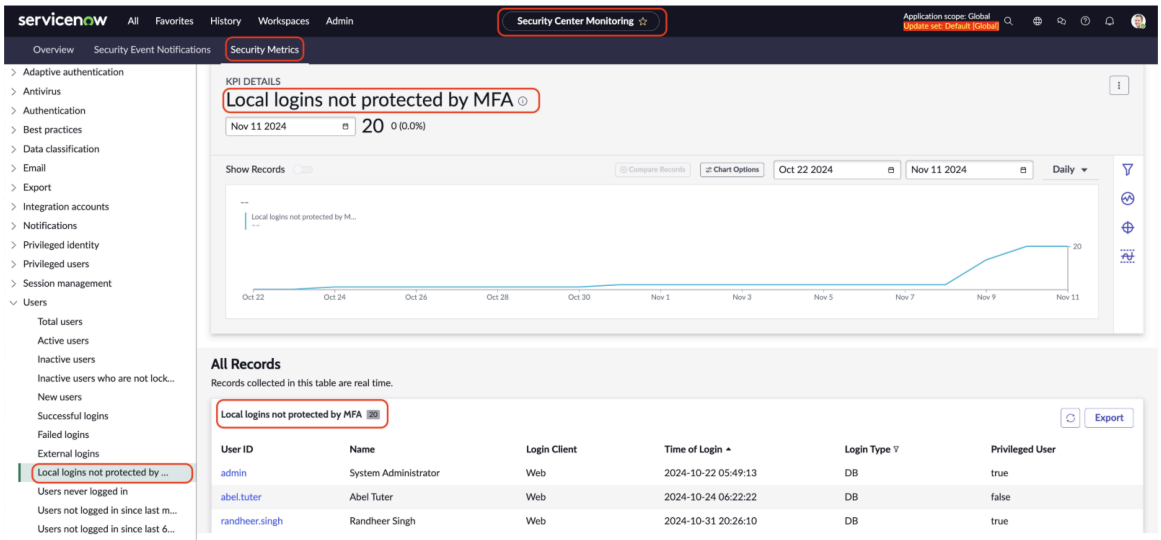
複数のユーザーによって共有される単一のアカウントはセキュリティ上のリスクです。アカウントを複数のユーザーと共有することはお勧めしません。

MFA メトリクス

MFA メトリクスの理解に関連する FAQ。

1. MFA が行われていないローカルログインの数を確認するにはどうすればよいですか?

次に移動できます セキュリティセンター > セキュリティコンソール > セキュリティメトリクス. [ユーザーのメトリクス] で、[MFA で保護されていないローカルログイン] をクリックします。



2. MFA セットアップを完了する必要があり、ローカルログインを実行しているユーザーの数を確認するにはどうすればよいですか?

次に移動できます セキュリティセンター > セキュリティコンソール > セキュリティメトリクス. [ユーザーのメトリクス] で、[MFA で保護されていないローカルログイン] をクリックします。

MFA タイプ

MFA タイプとそれが重要である理由に関連する FAQ です。

1. ServiceNow を使用した MFA で利用可能な検証方法のタイプは何ですか?

ServiceNow ベースシステムは、次の検証方法をサポートしています。

a. パスキー

b. Google Authenticator、Okta verify、Microsoft Authenticator、Authy、DUO などの TOTP 認証アプリ

c. Windows Hello、Apple Touch ID、Face ID、Android 指紋センサーなどの生体認証システム (FIDO2)。

d. YubiKey、Thetis などのハードウェアセキュリティキー (FIDO2)

e. メール ワンタイムパスワード (OTP)

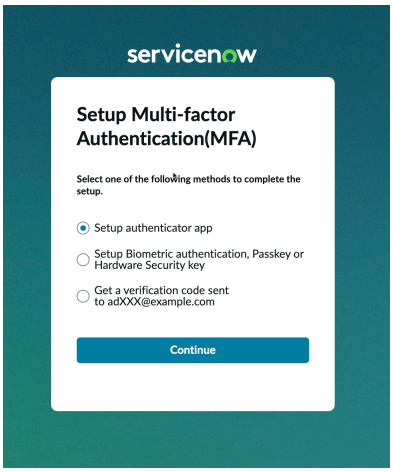
f. SMS OTP:SMS OTP ベースの MFA を有効にするには、SMS com.snc.authentication.sms_mfaplugin を使用したマルチファクター認証のインストールと要素の構成が必要です。

2. ユーザーは複数の MFA 要素または検証方法を設定できますか?

はい、ユーザープロファイルにアクセスして、複数の MFA 要素に登録できます。たとえば、生体認証装置を備えたラップトップを登録し、パスキーで携帯電話を使用し、認証アプリをセットアップできます。

3. MFA セットアップを完了するために、ユーザーはどのような手順を実行する必要がありますか?

ユーザーは、次のいずれかの MFA オプションを実行できます。



MFA セットアップの詳細については、 多要素認証 ドキュメントを参照してください。

4. SMS とメールの OTP ベースの MFA を特定のユーザーに制限することはできますか?

   アドミニストレーターは、メールおよび SMS OTP ベースの MFA 要素の MFA 要素ポリシーを設定して、これらの要素を特定のユーザーグループまたはロールに制限できます。

5. ユーザーは、認証アプリを設定できる携帯電話を持っていません。これらのユーザーが MFA を有効にするにはどうすればよいですか?

   Xanadu リリース以降は、携帯電話で認証アプリをセットアップしなくても、生体認証装置、パスキー、FIDO2 ハードウェアセキュリティキー、メール OTP ベースの MFA を使用できます。

6. エンドユーザーとして MFA を設定する方法は?

   MFA セットアップの詳細については、 マルチファクター認証の使用 ドキュメントを参照してください。

MFA リセット

MFA リセットに関する FAQ と、その重要性

1. 認証アプリのセットアップが失われました。どうすればリセットできますか?

   ユーザーが MFA によってロックアウトされないように、複数の要素に登録することをお勧めします。たとえば、TOTP 認証アプリ、パスキー、メール OTP ベースの MFA を使用できます。

   ユーザーが登録済みの MFA 要素を使用できない場合、アドミニストレーターは次の手順に従って MFA をリセットできます。

   a. 認証アプリのセットアップをクリアしています:
      - 移動先 すべて > 多要素認証 > ユーザー多要素設定.
      - テーブルでユーザーを検索します。
      - ユーザーに関連付けられたレコードを削除します

   b. FIDO2 認証システムとパスキーをクリアする:
      - 移動先 すべて > 多要素認証 > **Web** 認証 > ユーザーの公開認証情報.
      - テーブル内のユーザーの検索
      - ユーザーに関連付けられたレコードを削除します

   c. ユーザーに関連付けられている他のマルチファクターセットアップをクリアしています

- 移動先 すべて > 多要素認証 > ユーザー多要素設定.
- テーブル内のユーザーの検索
- ユーザーに関連付けられたレコードを削除します

アドミンを含む誰もインスタンスにアクセスできない場合。アドミンは、 Now Support で利用可能なカタログアイテムを使用して、セルフサービス MFA リセットを実行できます。

2. MFAによる管理者のロックアウトを回避するには?

   アドミニストレーターは、インスタンスへのアクセスがブロックされないように、複数の MFA 要素を登録します。

   アドミンを含む誰もインスタンスにアクセスできない場合。アドミンは、 Now Support で利用可能なカタログアイテムを使用して、セルフサービス MFA リセットを実行できます。

マルチファクター認証の詳細

マルチファクター認証 (MFA) は、ユーザーに基本認証情報以外の情報の提供を要求する認証方法です。

MFA は、ユーザーがサービスまたはアカウントにアクセスするために 2 つ以上の異なる検証要素を提供することを要求するセキュリティプロセスです。これにより、パスワードだけでなく、サービスに保護のセキュリティレイヤーが追加され、権限のない個人がアクセスすることが困難になります。

MFA は、複数の要素を必要とすることでセキュリティを大幅に強化し、フィッシングや個人情報の盗難などのさまざまなサイバー脅威から保護するのに役立ちます。MFA の仕組みに関するインサイトを次に示します。

- 第 **1** 要素:ユーザー名とパスワードを使用してログインするユーザー。
- 第 **2** 要素:ユーザーは、ユーザーに関する第 2 要素 (認証アプリやセキュリティキーなどの本人確認方法) の入力を求められます。

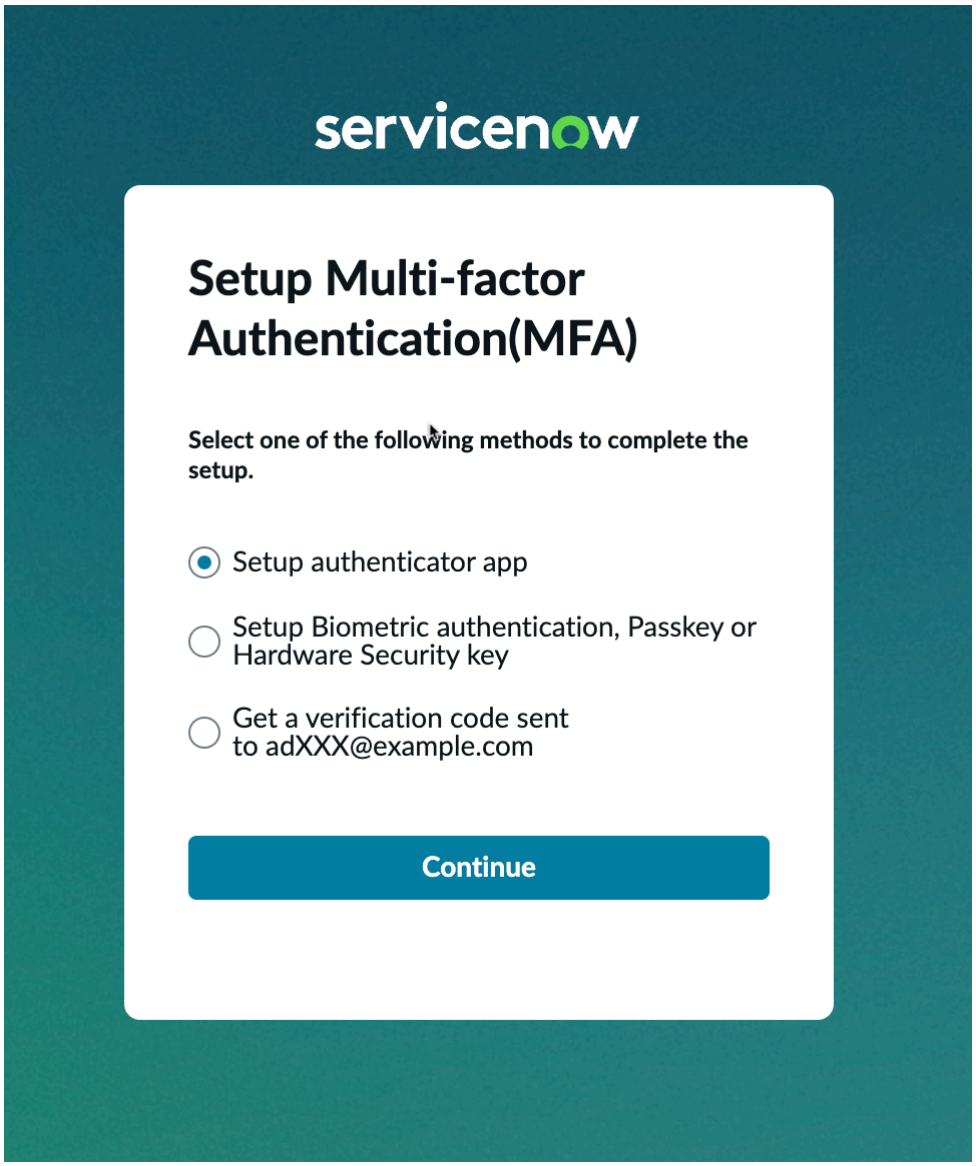
さらに、これらの要因は、通常、一般的なセキュリティの脅威に対する保護レベルに基づいて、セキュリティで保護されたものと保護されていないものに分類できます。

- 担保付き要素:
  - FIDO(Fast Identity Online):この要素は、ハードウェアトークンまたは生体認証方法を使用し、ユーザーがIDを確認するための物理デバイスまたは一意の生体認証特性を持っていることを確認することで、高レベルのセキュリティを提供します。
  - TOTP (Time-Based One-time Password): このファクターは、短期間 (通常は 30 秒) 有効なワンタイムパスワードを生成します。通常、モバイルアプリを介して配信され、ユーザーに特定のデバイスとアプリへのアクセスを要求することで、セキュリティのレイヤーが追加されます。
- 安全性の低い要因:
  - 電子メール： この要素は、ユーザーのメールアドレスに検証コードまたはリンクを送信します。便利ですが、メールアカウントが侵害される可能性があるため、安全性は低くなります。
  - SMS: この要素は、ユーザーの電話番号にテキストメッセージで検証コードを送信します。また、SIMスワップやその他の携帯電話の脆弱性の可能性があるため、安全性も低下します。

セキュリティを強化するには、メールやSMSなどの安全性の低い要素よりも、FIDOやTOTPなどの安全な要素の使用を優先することをお勧めします。

注:

- ServiceNow では、MFA はデフォルトでアクティブ化されています。
- MFA は、 *glide.authenticate.multifactor* プロパティを使用して有効にします。このプロパティを無効にする場合は、MFA を無効にするビジネス上の根拠を提供する必要があります。



ServiceNowの MFA は、認証アプリ、Fast IDentity Online 2 (FIDO2)、パスキー、時間ベースのワンタイムパスワード (OTP) などの検証方法をサポートしています。利用可能な検証方法の詳細は次のとおりです。

- 認証アプリ: 一意の一時的な確認コードを生成するアプリ。例:Okta、Google Authenticator、Microsoft Authenticator など
- **FIDO2**: 公開キー暗号化を使用してユーザー ID を検証する物理デバイス。例:ハードウェアキー(YubiKey)、生体認証スキャナー(AppleのTouch ID)。
- パスキー:生体認証センサー、PIN、またはパターンを使用してデバイスのロックを解除し、パスキーを使用してログインします。
- **OTP**:短時間のみ有効な一意のパスワードを生成するための秘密キーと現在時刻。例:SMS (OTP) とメール (OTP)。

MFA は、以下と併用できます。

- ローカルデータベース認証 (ネイティブ ServiceNow 認証) または ライトウェイトディレクトリアクセスプロトコル統合
- SSO、SAML、または SSO OIDC。詳細については、「複数プロバイダーのシングルサインオン (SSO)」を参照してください。

関連トピック

マルチファクター認証の検証方法

マルチファクター認証システムプロパティ

マルチファクター認証の構成

マルチファクター認証 (MFA) を構成して、 ServiceNowを使用するときのユーザーのセキュリティ体制を改善します。

まず、MFA を実装するには、ビジネスとユーザーのニーズに基づいて MFA 検証方法 を決定します。

ニーズを特定した後、 ServiceNow の MFA を使用して、ユーザーに適した方法またはその組み合わせを選択できます。さらに、さまざまなマルチファクター認証プロパティを使用して MFA を有効化、無効化、および構成し、認証基準に基づいてユーザーの MFA エクスペリエンスをさらに強化することもできます。

MFA の構成に関連する重要なトピックの一部を次に示します。

- マルチファクター認証の検証方法
- マルチファクター認証システムプロパティ
- マルチファクター認証基準

マルチファクター認証コンテキスト

マルチファクター認証 (MFA) ポリシーコンテキストでは、ポリシーを使用して、ログインプロセス中に MFA を適用する方法とタイミングを定義します。

MFA コンテキストレコード

MFA ポリシーコンテキストは、ユーザーがログイン時に 2 番目の認証フォームを提供する必要があるかどうかを定義します。このコンテキストは、認証後および認証前のポリシーとしてインスタンスへのアクセスを拒否するわけではありません。このコンテキストで選択したポリシーは、マルチファクター認証のユーザーまたはロールベースの構成よりも優先されます。

MFA コンテキストにアクセスするには、 すべて > 多要素認証 > **MFA** コンテキスト.

認証後ポリシーコンテキストレコードのフィールドを使用して、インスタンスでのポリシーの使用方法を定義します。

注:

- デフォルトのポリシーがステップアップ **MFA** ポリシーの場合、ステップアップ **MFA** ポリシーで構成されたポリシーが true と評価されると、ユーザーにはマルチファクター認証が表示されます。ポリシーは、ユーザーまたはロールベースの構成よりも優先されます。
- SSO ログインを使用する MFA は、glide.authenticate.mfa.with.multisso.enabled プロパティが true に設定されている場合にのみ使用できます。
- [認証ポリシー] レコードに移動して、参照されているポリシーフィールド ([ステップアップ **MFA** ポリシー] または [ステップダウン **MFA** ポリシー]) に「ポリシー入力」を追加または編集します。
- MFA コンテキストポリシーは、ユーザーのログインにのみ適用されます。API 認証、基本認証、および OAuth リソース所有者のパスワード認証情報の付与には適用されません。

MFA コンテキストフォーム

フィールド	説明
名前	ポリシーコンテキストの名前このフィールドは静的であり、変更することはできません。
説明	コンテキストの説明
デフォルトポリシー	ポリシーを評価するときの、このコンテキストのデフォルト動作を定義します。次のオプションのいずれかを選択します。 ステップアップ MFA ポリシー [ステップアップ **MFA** ポリシー] フィールドで定義されたポリシー条件が true と評価された場合に MFA がユーザーに強制されます。 ステップダウン MFA ポリシー デフォルトで MFA を適用します。[ステップダウン **MFA** ポリシー] フィールドで定義されたポリシー条件が true と評価された場合にかぎり、MFA はユーザーに強制されません。
ステップアップ MFA ポリシー	このコンテキストで使用されるポリシー。このフィールドは、[デフォルトポリシー] フィールドが [ステップアップ **MFA** ポリシー] に設定されている場合にのみ表示されます。
ステップダウン MFA ポリシー	このコンテキストで使用されるポリシー。このフィールドは、[デフォルトポリシー] フィールドが [ステップダウン **MFA** ポリシー] に設定されている場合にのみ表示されます。

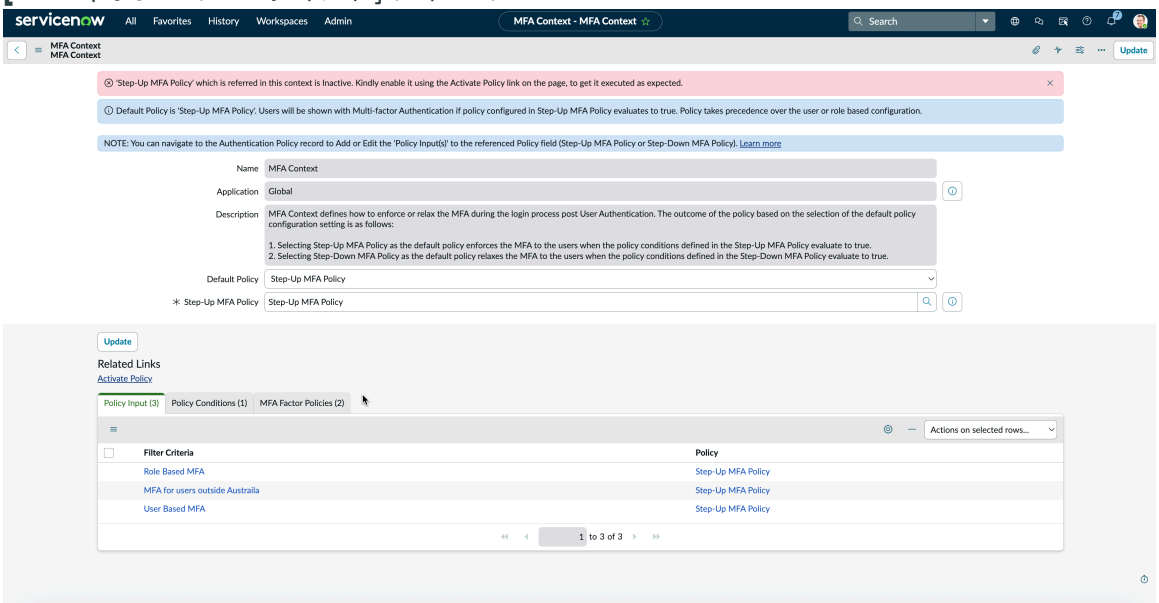
ポリシーの入力と条件

[**Policy Input** (ポリシー入力)] タブと [**Policy Conditions** (ポリシー条件)] タブには、[**Step-Up MFA Policy** (ステップアップ **MFA** ポリシー)] または [**Step-Down MFA Policy** (ステップダウン **MFA** ポリシー)] フィールドで選択されたポリシーの入力と条件が表示されます。これらのタブは参照できますが、ポリシーの入力や条件を変更するために使用することはできません。ポリシー設定を変更するには、[ステップアップ **MFA** ポリシー] または [ステップダウン **MFA** ポリシー] フィールドの横にある参照アイコン (ⓘ) を使用してポリシーに移動します。

注: ポリシー条件はここから作成できますが、ポリシーページから新しいポリシー条件を追加することをお勧めします。

この例は、ステップアップ MFA ポリシーを使用して構成された MFA コンテキストレコードを示しています。このデフォルトポリシーは、ポリシーで定義されている条件が true と評価された場合にのみ MFA が適用されることを意味します。コンテキストは、[ステップアップ **MFA** ポリシー] と呼ばれるポリシーを使用します。そのポリシーには、[ポリシー入力] タブと [ポリシー条件] タブに表示される一連の入力と条件があります。

[MFA ポリシーのコンテキスト] フォーム



MFA 要素ポリシー

MFA 要素ポリシーは、組織のセキュリティ体制の重要なコンポーネントであり、パスワード以外の追加の検証手順を適用できます。これらのポリシーは、ユーザーがアクセスするために採用する必要がある認証方法を定義し、柔軟でカスタマイズ可能な認証アプローチを提供します。詳細については、「マルチファクター認証要素ポリシー」を参照してください。

マルチファクター認証の検証方法

ServiceNowのMFAは、認証アプリ、Fast IDentity Online 2(FIDO2)、時間ベースのワンタイムパスワード(TOTP)などの検証方法をサポートしています。

ユーザーは、自分のユーザー名とパスワードに加えて、以下のオプションを使用して、マルチファクター認証要件を履行できます。ユーザーは、認証アプリケーション、生体認証スキャナー、ハードウェアキー、SMS などの MFA 要素を個別にセットアップできます。

認証アプリケーション

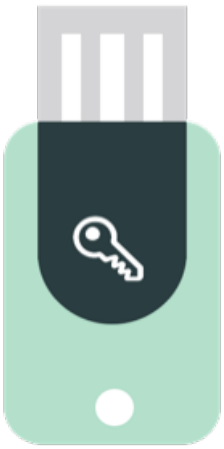
認証アプリケーションは、一時的なパスコードを生成するサードパーティソフトウェアです。ユーザーは、これらのパスコードをパスワードとともに使用して、マルチファクター認証 (MFA) を必要とするインスタンスにログインできます。これらのアプリケーションの詳細については、「認証アプリケーション」を参照してください。

ハードウェアキーと生体認証スキャナー (Web 認証)

Integration - Web Authentication (com.snc.integration.webauthn) プラグインを使用すると、インスタンスでハードウェアキーまたは生体認証を行えます。

生体認証装置は、指紋または顔認識を使用してユーザーを識別します。ユーザーは、マルチファクターログインプロセスの一部としてデバイスでこれらの認証システムを使用できます。生体認証装置の登録の詳細については、「生体認証装置の登録」を参照してください。

ハードウェアキーは、認証に使用できる物理ハードウェアです。ハードウェアキーをデバイスのポートに挿入して認証します。ハードウェアキーの登録の詳細については、「ハードウェアセキュリティキーの登録」を参照してください。



パスキー

パスキー認証は、パスワード不要の安全なログイン方法であり、ユーザーは生体認証センサー、PIN、またはパターンでデバイスのロックを解除することでパスキーを使用します。

SMS

アドミニストレーターは、インスタンスにログインしようとするユーザーに SMS ベースの OTP の使用を要求するように ServiceNow インスタンスを設定できます。

ユーザーが ServiceNow にログインしようとすると、sys_user レコードに関連付けられた携帯電話番号に SMS OTP が送信されます。ユーザーは、モバイルデバイスに送信された 6 桁の検証コードを入力して本人確認を行うことができます。詳細については、「MFA 要素としての SMS」を参照してください。

メール

アドミニストレーターは、インスタンスにログインしようとするユーザーにメールベースの OTP の使用を要求するように ServiceNow インスタンスを設定できます。

ユーザーが ServiceNowにログインしようとすると、メール OTP がユーザーのメールアドレスに送信されます。ユーザーは、メールアドレスに送信された 6 桁の検証コードを入力して本人確認を行うことができます。詳細については、「MFA 要素としてのメール」を参照してください。

関連トピック

- Web 認証 - MFA
- マルチファクター認証の検証方法
- MFA 要素としてのメール

Web 認証 - MFA

Integration - Web Authentication (com.snc.integration.webauthn) を使用して、インスタンスでハードウェアキーまたは生体認証リーダーの認証を許可します。

ハードウェアキーは、認証に使用できる物理ハードウェアです。ハードウェアキーをデバイスのポートに挿入して認証します。ハードウェアキーの登録の詳細については、「ハードウェアセキュリティキーの登録」を参照してください。

生体認証装置は、指紋または顔認識を使用してユーザーを識別します。ユーザーは、マルチファクターログインプロセスの一部としてデバイスでこれらの認証システムを使用できます。生体認証装置の登録の詳細については、「生体認証装置の登録」を参照してください。

生体認証を使用したマルチファクター認証の構成

アドミニストレーターは、[ユーザーの公開認証情報] リストを使用して、ユーザーが作成した認証情報を表示および管理できます。

ユーザーが認証アプリケーション、生体認証、またはハードウェアキーを登録すると、インスタンスにより、ユーザー公開認証情報 [sys_user_public_credential] テーブルにレコードが作成されます。このテーブルを使用して、認証システムを登録したユーザー、認証システムのタイプ、および認証システムが登録および使用された時期を確認します。これらのレコードを非アクティブとしてマークし、ユーザーがこれらの認証情報を使用できないようにすることもできます。

Web 認証 (FIDO2) を使用するには、 **Integration - Web Authentication (com.snc.integration.webauthn)** プラグインを有効にする必要があります。

注: **Integration - Web Authentication (com.snc.integration.webauthn)** プラグインはデフォルトでインストールされます。

ユーザーの公開認証情報フォーム

フィールド	説明
認証情報ニックネーム	認証情報のニックネーム。このニックネームは、ユーザーが認証システムを登録するときに選択します。
ユーザー	認証情報に関連付けられたユーザー
アクティブ	この認証情報がアクティブかどうか。アドミニストレーターは、レコードを非アクティブに設定して、ユーザーがこの認証情報を使用して認証されるのを防ぐことができます。
認証システム	ユーザーによって登録された認証システムのタイプ
登録時間	ユーザーがこの認証情報を作成した日時

ユーザーの公開認証情報フォーム (続く)

フィールド	説明
前回の使用時間	ユーザーがこの認証情報を使用してログインした前回の日時

制限付き認証システムのタイプ

生体認証装置など認証方法を制限している場合、ユーザーはそのタイプの新しい認証情報を作成できません。ただし、この制限を行う前に作成された認証情報は引き続き機能します。[ユーザーの公開認証情報] テーブルのレコードを無効にして、作成後にこれらの認証情報が使用されることがないようにすることができます。

認証システムの構成オプション

[認証システムの構成] ページを使用して、インスタンスの認証システムのオプションを管理します。

移動先 多要素認証 > **Web** 認証 > 認証システムの構成 をクリックして、デフォルトの構成オプションを表示および編集します。

[認証システムの構成] フォーム

フィールド	説明
許可された認証システムのタイプ	登録が許可されている認証システムのタイプ。次から選択します。 • プラットフォーム：認証システムが接続されているか、デバイスに組み込まれています。モバイルデバイスで利用可能な指紋リーダーまたは顔認識 (Apple FaceID や TouchID など) はこのカテゴリに分類されます。 • ローミング：認証システムをコンピューターや他のクライアントデバイスから取り除いて、別の場所で使用できます。ハードウェアキーはこのカテゴリに分類されます。
証明書タイプ	値を [直接] または [間接] に設定するには、登録時に認証システムの来歴を証明するために認証システムのメタデータをインポートする必要があります。 • なし • 直接 • 間接
プラットフォームの自己証明書	プラットフォーム認証システムに対して自己証明書が有効になっているかどうか。
クロスプラットフォームの自己証明書	ローミング認証システムに対して自己証明書が有効になっているかどうか。
ユーザー検証	[優先] または [必須] から選択します。必要に応じて、Web 認証フローは PIN または生体認証を使用した検証をユーザーに要求します。

[認証システムの構成] フォーム (続く)

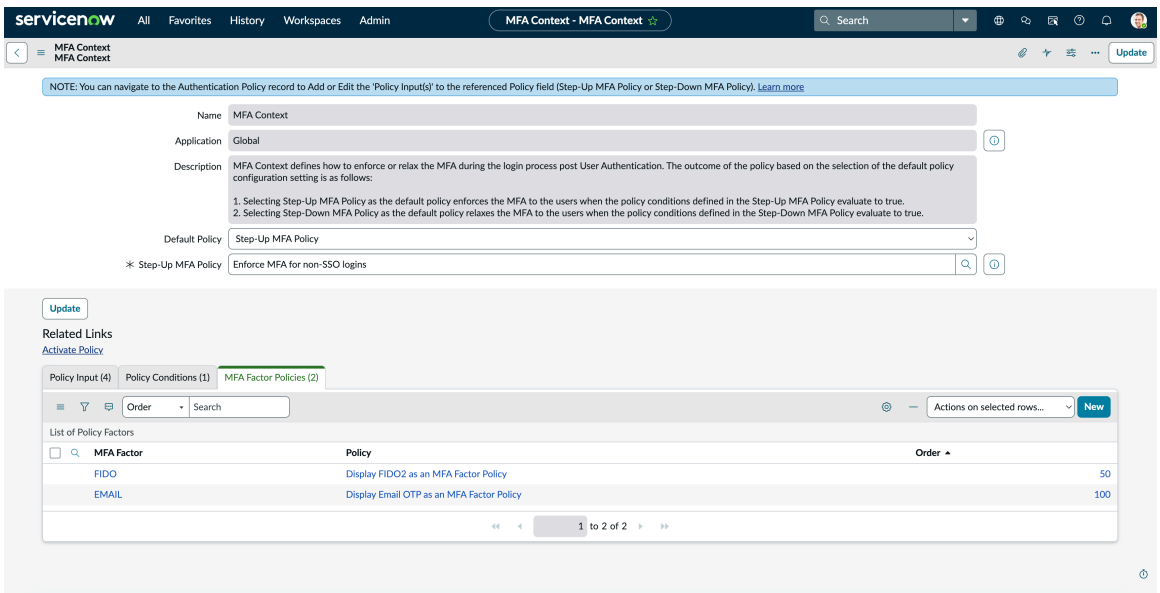
フィールド	説明
ユーザープレゼンスを検証	Web 認証フローでユーザープレゼンスの検証が必要かどうか。
レジデントキー	[優先] または [必須] から選択します。必要に応じて、認証システムは認証システムのストレージ内に公開鍵認証情報を保持します。
タイムアウト (ミリ秒)	Web 認証の登録と認証を完了するための最大時間制限。時間はミリ秒単位です。

マルチファクター認証要素ポリシー

MFA 要素ポリシーを使用して、インスタンスで許可する認証要素のタイプを指定します。

MFA 要素ポリシーは、組織のセキュリティ体制の重要なコンポーネントであり、パスワード以外の追加の検証手順を適用できます。これらのポリシーは、ユーザーが組織のリソースにアクセスするために採用する必要がある認証方法を定義し、柔軟でカスタマイズ可能な認証アプローチを提供します。

MFA 要素ポリシーの実装は、組織のシステムとデータのセキュリティを強化するために不可欠です。これらのポリシーは、サイバー脅威に対する追加の保護レイヤーを提供し、攻撃者が不正アクセスを取得することをより困難にします。



MFA 要素ポリシーを使用するには、ポリシー入力とポリシー条件を MFA コンテキストとともに構成する必要があります。詳細については、「マルチファクター認証コンテキスト」を参照してください。

許可または必要な認証要素のタイプを指定できる、 ServiceNow で使用可能な MFA 要素ポリシーは次のとおりです。

- FIDO2
- SMS
- メール

MFA 要素ポリシーを最大限に活用するには、ポリシーを効果的に構成および管理する方法を理解する必要があります。これには、認証方法の定義、ポリシーの入力と条件の指定、ポリシー適用 (MFA コンテキスト) の構成が含まれます。MFA 要素ポリシーを理解して実装することで、組織のシステムとデータのセキュリティと整合性を大幅に向上させることができます。

関連トピック

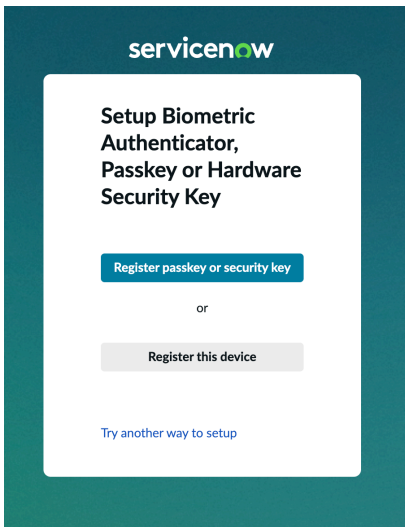
- マルチファクター認証コンテキスト
- MFA 要素としての FIDO2
- MFA 要素としての SMS
- MFA 要素としてのメール

MFA 要素としての FIDO2

FIDO2 を MFA 要素ポリシーとして構成して、MFA を強制できます。

FIDO2 は、ユーザーが物理的なセキュリティキーまたは生体認証を使用して認証できるようにする、パスワードなしの認証標準です。従来の MFA 手法に代わるより安全な代替手段を提供し、フィッシングやその他のサイバー攻撃のリスクを軽減します。

FIDO2 要素ポリシーの機能拡張により、マルチファクター認証 (MFA) ポリシーに安全な認証方法が提供されます。FIDO2 を MFA 要素ポリシーオプションとして構成し、メールや SMS などの従来の方法と比較してより高いレベルのセキュリティを提供できます。



FIDO2 要素ポリシーを構成し、ユーザーが要素ポリシー条件を満たすと、 ServiceNow へのログイン時に、登録済みのハードウェアキーまたは生体認証をプロファイルにまだ追加していないユーザーに対して FIDO2 セットアップが表示されます。

登録が完了すると、ログインするための第 2 要素検証画面が表示されます。

注: FIDO2 は、ユーザーが自己登録することもできます。自己登録方法の詳細については、「ユーザープロファイルでのマルチファクター認証の設定」を参照してください。

主なメリット

FIDO2 を MFA 要素として使用する主なメリットは次のとおりです。

排他的なFIDO2認証
: 高権限のアカウントは、FIDO2 の強力な認証機能 (生体認証、パスキー、またはハードウェアセキュリティキー) を使用してのみ認証できるようにします。

安全性の低い方法の除外

FIDO2 が唯一の一致するポリシーである場合、他の認証方法を抑制します。

強制登録

まだ登録していない場合は、ユーザーに FIDO2 キーの登録を要求します。

きめ細かな制御

ポリシーベースのターゲティングを使用して、特定のロールまたはグループに厳密な適用を適用します。

より高いセキュリティ要素として、FIDO2 には排他的な適用機能があります。ユーザーに一致する唯一のポリシーである場合:

- 登録によって登録された他の要素を上書きします。
- ユーザーが登録されていない場合、FIDO2 登録を強制します。
- 排他的な認証オプションになります。

構成とユーザーの動作例

次の表は、ロールと登録済み要素に基づいてさまざまなユーザーシナリオがどのように処理されるかを示しています。

ユーザーの例	ロールあり	登録済み要素	一致するポリシー	動作
andrew.och	**ITIL**	なし	FIDO2	ユーザーは、FIDO2 のみを使用した MFA セットアップにリダイレクトされます。登録後は、FIDO2 が唯一の認証オプションです。
abel.tuter	**ITIL**	認証システム	FIDO2	ユーザーが自己登録要素として認証システムを持っている場合でも、ユーザーは FIDO2 のみを使用した MFA セットアップにリダイレクトされます。 **注:** ユーザーが MFA 要素に登録していない場合、ユーザーは FIDO2 を使用した MFA セットアップにリダイレクトされます。
aileen.motterm	資産	認証システム	メール	ログイン時に [メール] と [認証システム] オプションが表示されます。ユーザーは、係数を選択するか、オプションで FIDO2 を登録することができます。
エイブラハム・リンカーン	資産、**ITIL**	認証システム	メールと FIDO2	ログイン時に [メールと認証システム] オプションが表示されます。ユーザーは検証中に FIDO2 を登録できます。登録後、ユーザーは 3 つの要素すべてを表示できます。

FIDO2 を MFA 要素ポリシーとして構成することで、認証プロセスのセキュリティを大幅に強化できます。

FIDO2 を MFA 要素として設定する

FIDO2 を認証の MFA 要素ポリシーとして表示するようにポリシー入力と条件を設定します。

始める前に
必要なロール：admin

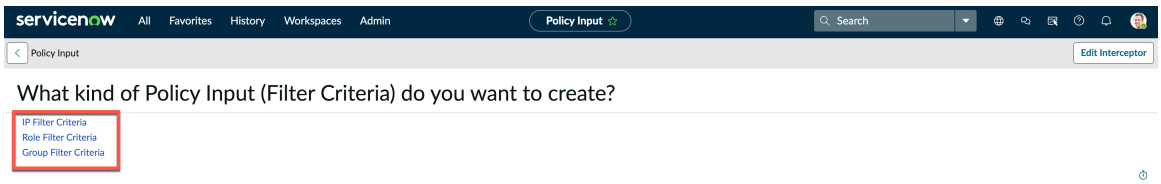
手順

1. 移動先 すべて > 多要素認証 > **MFA** コンテキスト.
2. [**MFA** 要素ポリシー] タブを選択します。
3. [**FIDO2** を **MFA** 要素ポリシーとして表示 **(Display FIDO2 as an MFA Factor Policy)**] を選択します。
4. [新規] を選択して ポリシー入力を追加します。
5. 作成するフィルター基準を選択します。

   フィルター基準のタイプは次のとおりです。

   - IP フィルター基準
   - ロールフィルター基準
   - グループフィルター基準

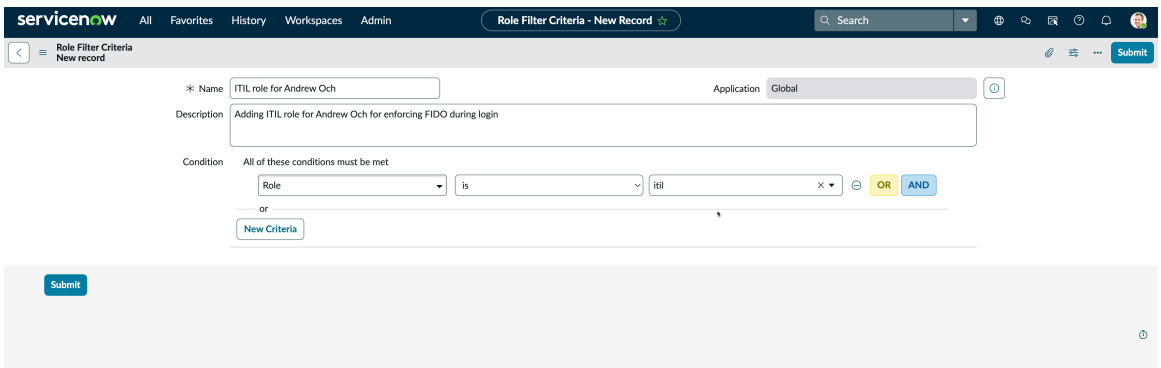
   例：ロールフィルタークライテリア



6. [ロールフィルター基準] を選択し、ロールフィルター基準のフィールドに入力してレコードを送信します。

   新しいポリシーが作成されます。詳細については、「ロールフィルター基準」を参照してください。

   ユーザー (**andrew.och**) の **ITIL** ロールをポリシーの入力と送信として使用する例を見てみましょう。

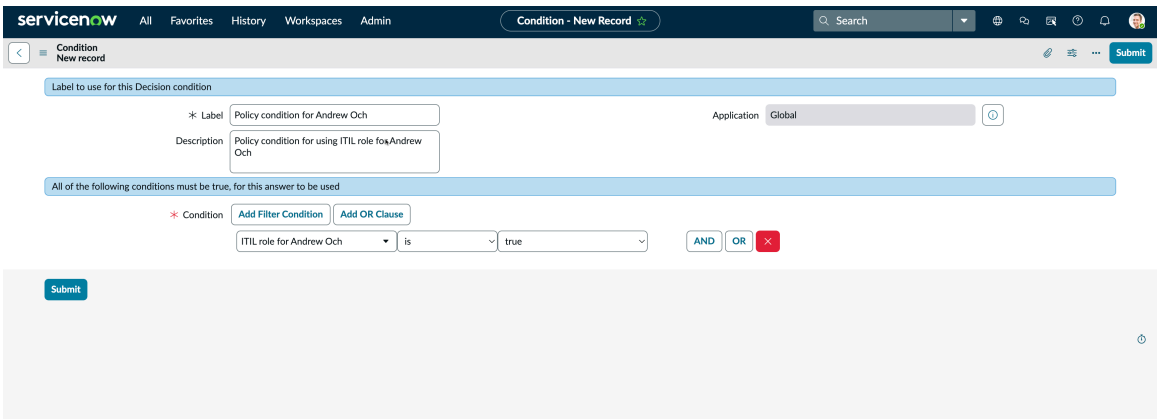


7. [ポリシー - FIDO2 を MFA 要素ポリシーとして表示 (Policy - Display FIDO2 as an MFA Factor Policy)] ページで、 [ポリシー条件] を選択します。

8. [新規] を選択してポリシー条件を追加します。
9. フォームで、フィールドに入力します。

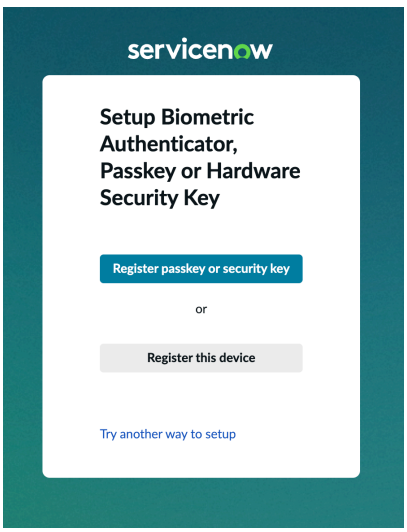
[条件] フォーム

フィールド	説明
ラベル	条件を識別する名前
説明	条件の説明
条件	認証要求を評価するために使用される複数のポリシー入力 (フィルター基準) の論理的な組み合わせ。 条件に対して作成されたロールベースのフィルター基準ポリシーを選択します。



10. **[Submit** (送信)] を選択します。

ポリシーの入力と条件に基づいて、ユーザー (**andrew.och**) がインスタンスにログインしようとすると、登録と登録を行うための FIDO 画面としてユーザーが表示されます。



さまざまな構成例とユーザーの動作の詳細については、「 構成とユーザーの動作例」を参照してください。

11. オプション: 追加のポリシー条件を作成するには、手順 8 を繰り返します。

> **注:** 複数のポリシー条件を作成する場合、アクセスポリシーの最終的な出力は、すべてのポリシー条件の論理 OR 出力によって決まります。条件に基づいてポリシーが評価されます。

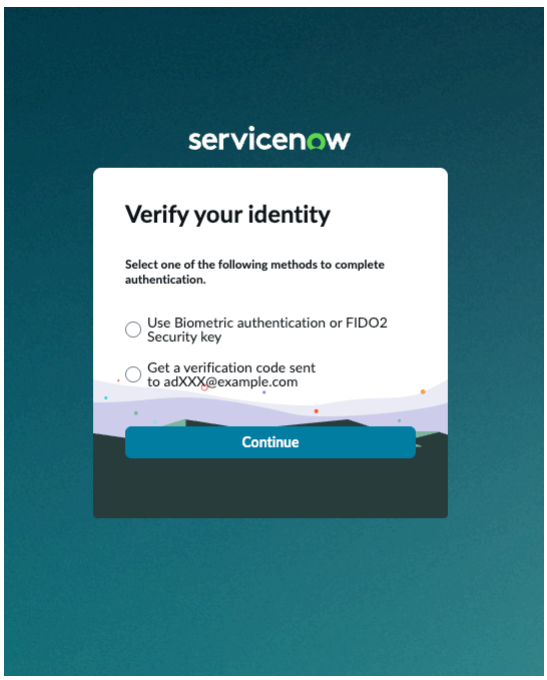
MFA 要素としての SMS

認証要素として SMS を使用するマルチファクター認証 (MFA)

アドミニストレーターは、インスタンスにログインしようとするユーザーに SMS ベースの OTP の使用を要求するように ServiceNow インスタンスを設定できます。

ユーザーが ServiceNow にログインしようとすると、sys_user レコードに関連付けられた携帯電話番号に SMS OTP が送信されます。ユーザーは、モバイルデバイスに送信された 6 桁の検証コードを入力して本人確認を行うことができます。

すぐに利用可能な Twilio を使用して、SMS で MFA を構成することもできます。詳細については、「マルチファクタ認証プロバイダー」を参照してください。



さらに、SMS を使用した MFA は、フィルター基準を使用してポリシー入力と条件に基づいて制御できます。フィルター基準のタイプは次のとおりです。

- IP フィルター基準
- ロールフィルター基準
- グループフィルター基準

MFA with SMS プラグインのアクティブ化

SMS を使用した MFA の場合、Multi-factor Authentication with SMS (com.snc.authentication.sms_mfa) プラグインをインストールします。

始める前に
必要なロール：admin

このタスクについて
次のアイテムは、Multi-factor Authentication with SMS とともにインストールされます。

- 適応認証 (com.snc.adaptive_authentication)
- Notify Twilio Direct ドライバー (com.snc.notify.twilio_direct)

依存プラグイン：Integration - マルチファクター認証 (MFA) (com.snc.integration.multifactor.authentication)

> **注:** SMS を生成するためのカスタムプロバイダーを設定する場合は、プラグインのインストール時にデモデータをロードできます。

手順

1. 移動先 すべて > システムアプリケーション > 利用可能なすべてのアプリケーション > すべて.
2. フィルター基準と検索バーを使用して、Multi-factor Authentication with SMS プラグイン (com.snc.authentication.sms_mfa) を検索します。

   名前または ID でプラグインを検索できます。プラグインが見つからない場合は、ServiceNow 担当者から要求する必要があります。
3. [インストール] を選択して、インストールプロセスを開始します。

   > **注:** ドメインセパレーションと代理アドミンがインスタンスで有効になっている場合、管理ユーザーはグローバルドメインに含まれている必要があります。それ以外の場合、次のエラーが表示されます： 「別の操作が実行されているため、アプリケーションのインストールは利用できません： <プラグイン名> のプラグインの有効化 (Application installation is unavailable because another operation is running: Plugin Activation for <plugin name>)」

   インストールが完了するとメッセージが表示されます。プラグインとともにインストールされるコンポーネントの詳細については、「 アプリケーションとともにインストールされているコンポーネントの検索 」を参照してください。

SMS を MFA 要素として設定

認証の MFA 要素ポリシーとして SMS OTP を表示するようにポリシー入力と条件を設定します。

始める前に
必要なプラグイン：Multi-factor Authentication with SMS (com.snc.authentication.sms_mfa)。

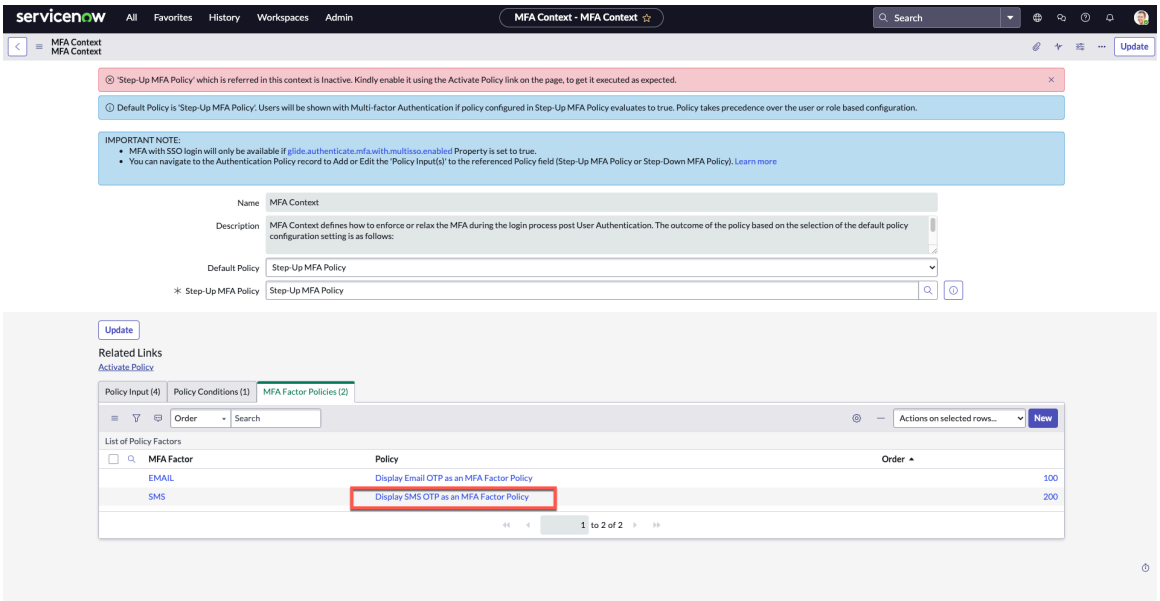
必要なロール：admin

> **注:** SMS 要素ポリシーを適用するには、MFA コンテキストポリシーが true と評価される必要があります。

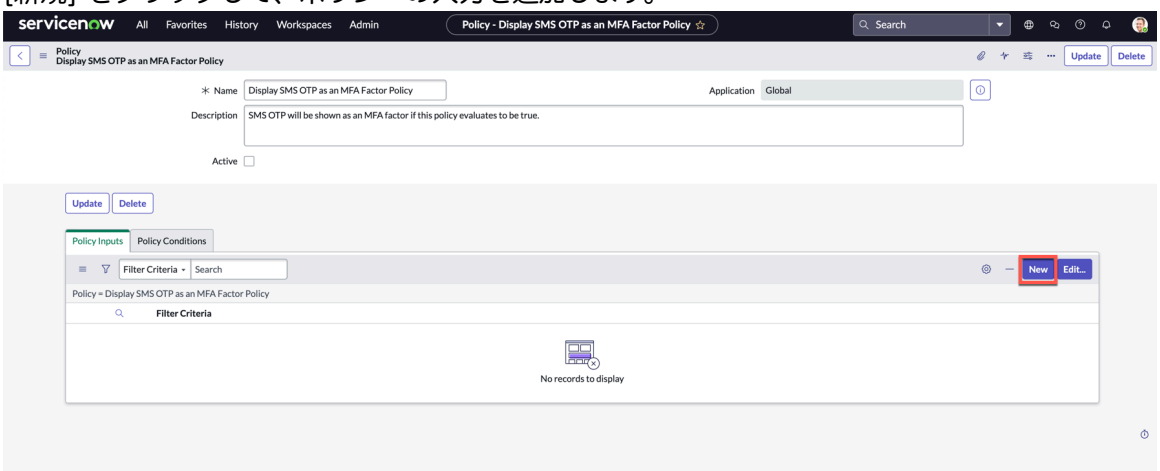
手順

1. 移動先 すべて > 多要素認証 > **MFA** コンテキスト.
2. **[MFA** 要素ポリシー] タブをクリックします。
3. **[SMS OTP** を **MFA** 要素ポリシーとして表示 **(Display SMS OTP as an MFA Factor Policy)]** を選択します。

自動翻訳



4. [新規] をクリックして、ポリシーの入力を追加します。

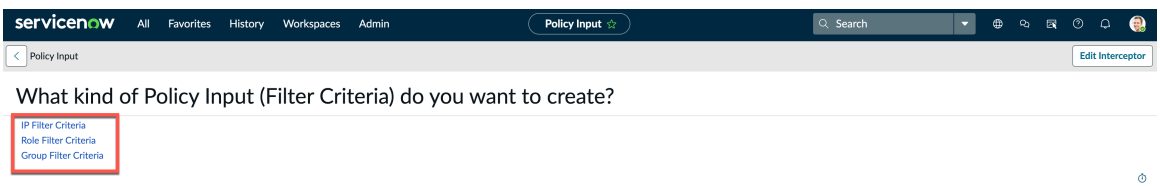


5. 作成するフィルター基準を選択します。

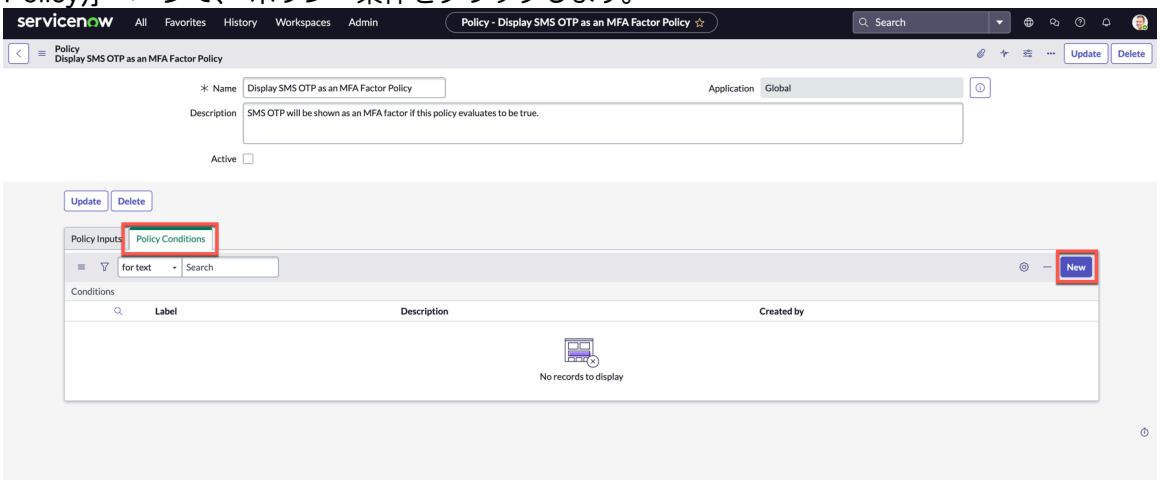
フィルター基準のタイプは次のとおりです。

- IP フィルター基準
- ロールフィルター基準
- グループフィルター基準

例：ロールフィルタークライテリア



6. [ロールフィルタークライテリア] をクリックし、ロールフィルタークライテリアのフィールドに入力してレコードを送信します。
   新しいポリシーが作成されます。詳細については、「ロールフィルター基準」を参照してください。
7. [ポリシー - SMS OTP を MFA 要素ポリシーとして表示 (Display SMS OTP as an MFA Factor Policy)] ページで、 ポリシー条件をクリックします。



8. [新規] をクリックして、ポリシー条件を追加します。
9. フォームのフィールドに入力します。

[条件] フォーム

フィールド	説明
ラベル	条件を識別する名前
説明	条件の説明
条件	認証要求を評価するために使用される複数のポリシー入力 (フィルター基準) の論理的な組み合わせ。 条件に対して作成されたロールベースのフィルター基準ポリシーを選択します。

10. [送信] をクリックします。
11. オプション: 追加のポリシー条件を作成するには、手順 8 を繰り返します。

> **注:** 複数のポリシー条件を作成する場合、アクセスポリシーの最終的な出力は、すべてのポリシー条件の論理 OR 出力によって決まります。これは、ポリシー条件のいずれかが満たされると、ポリシーが true と評価されることを意味します。

ロールフィルター (ユーザー) ポリシーに基づいて、ロールに指定された条件が一致すると、MFA 要素としての SMS がユーザーの認証のオプションとして表示されます。

マルチファクタ認証プロバイダー

MFA プロバイダーを使用して SMS およびメールベースの認証を構成し、すべてのユーザーが安全にログインできるようにします。

ServiceNow AI Platform では、メールや SMS などのマルチファクター認証を使用するメカニズムで MFA プロバイダーを構成できます。

ServiceNow AI Platform 内の MFA で利用可能な次のプロバイダー構成を使用できます。

- メールプロバイダー構成
- Twilio プロバイダー構成
- Infobip プロバイダー構成

注: Twilio および Infobip プロバイダー構成は、Multi-factor Authentication with SMS (com.snc.authentication.sms_mfa) および Notify - Twilio Direct Driver (com.snc.notify.twilio_direct) プラグインをインストールするときに [デモデータのロード] を有効にすると自動的に入力されます。

独自のプロバイダー構成を作成して、SMS とメールを使用したマルチファクター認証を有効にすることもできます。

注: Infobip プロバイダー構成はデモデータの一部として提供されます。独自のプロバイダーを構成するための要件に基づいてフィールドを編集できます。

MFA プロバイダーの構成

プロバイダーと SMS およびメールを構成して、すべてのユーザーが安全にログインできるようにします。

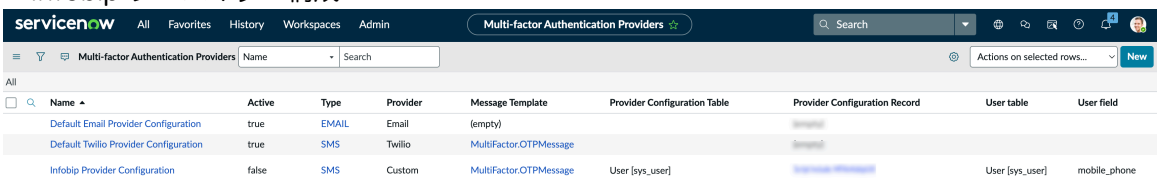
始める前に
必要なロール：admin

手順

1. 移動先 すべて > 多要素認証 > プロバイダー.

   ServiceNow AI Platform 内の MFA で次のプロバイダー構成を使用できます。

   - メールプロバイダー構成
   - Twilio プロバイダー構成
   - Infobip プロバイダー構成



2. 新しいプロバイダーを作成するには、[新規] をクリックします。
3. フォームの各フィールドに入力します。

[条件] フォーム

フィールド	説明
名前	レコードの名前

フィールド	説明
タイプ	レコードの説明です。
プロバイダー	**[Twilio]** または [カスタム] を選択します。 注: Twillio を設定するには、「Twilio の使用を通知で設定する」を参照してください。 [カスタム] を選択する場合は、次のフィールドを指定する必要があります。 ○ プロバイダー構成テーブル ○ プロバイダー構成レコード ○ スクリプト ○ ユーザーテーブル ○ ユーザーフィールド
メッセージテンプレート	レコードのメッセージテンプレート
有効	プロバイダー構成をアクティブにするオプション

4. [送信] をクリックします。

   メッセージテンプレートとプロバイダー構成に基づいて、ログインプロセス中に認証のための要素として SMS またはメールがユーザーに送信されます。

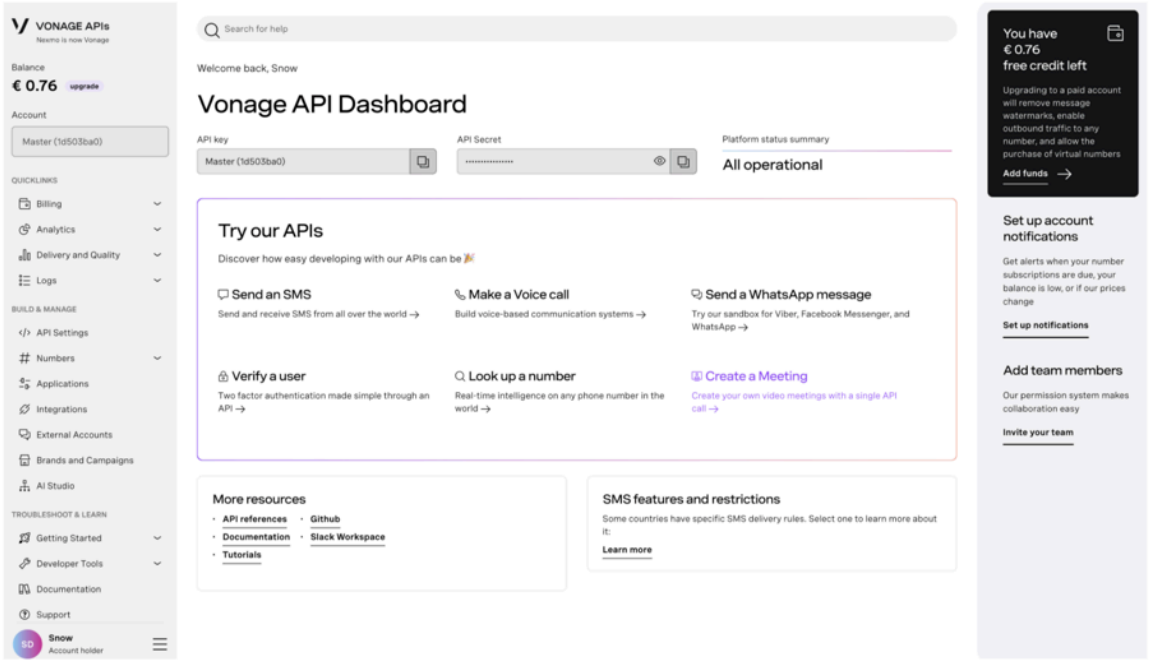
Vonage プロバイダーのカスタム構成 (チュートリアル)

SMS と Vonage プロバイダーを構成して、すべてのユーザーが安全にログインできるようにします。

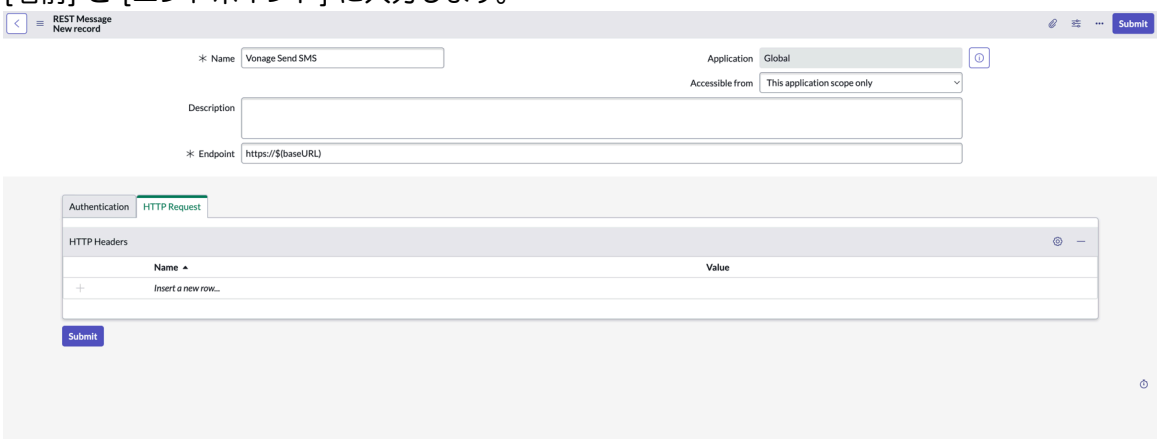
始める前に
必要なロール：admin

手順

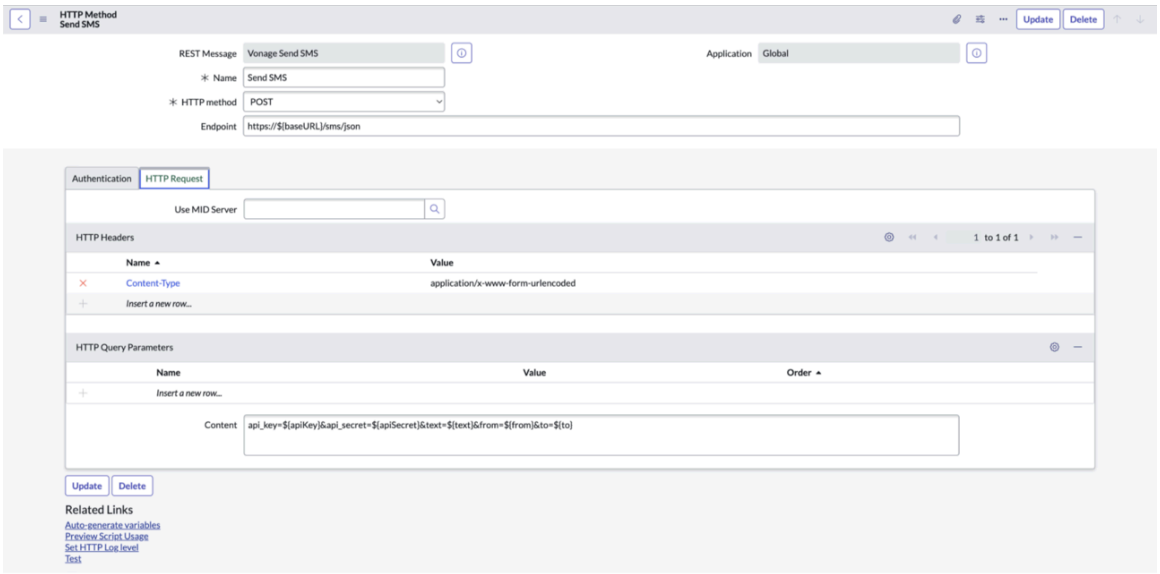
1. 移動先 すべて > システム **Web** サービス > アウトバウンド > **REST** メッセージ を行い、Vonage API ダッシュボードからの情報に基づいて REST メッセージ構成を実行します。



2. [新規] をクリックし、新しい **REST** メッセージを作成します。
3. [名前] と [エンドポイント] に入力します。



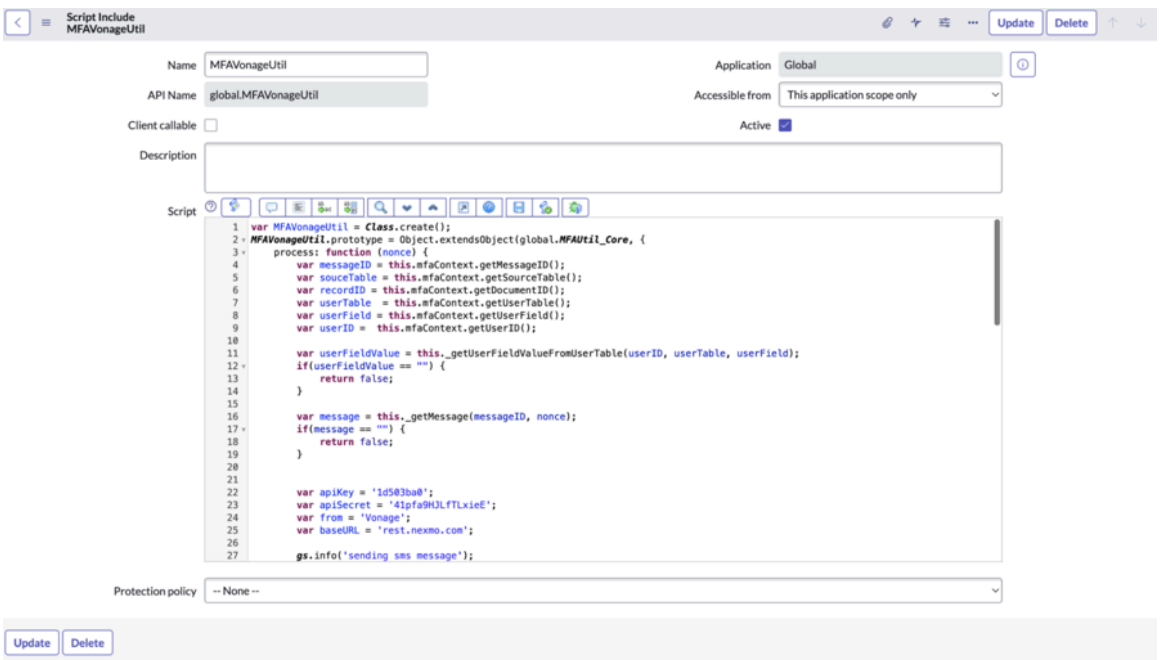
4. [送信] をクリックします。
5. **[HTTP** メソッド] 関連リストで、作成したレコードを開き、[新規] をクリックして、**POST** として HTTPメソッドを選択します。
6. 次のフィールドに入力します。
   - エンドポイント： `https://${baseURL}/sms/json`
   - コンテンツ：`api_key=${apiKey}&api_secret=${apiSecret}&text=${text}&from=${from}&to=${to}`
   - コンテンツタイプ：`application/x-www-form-urlencoded`



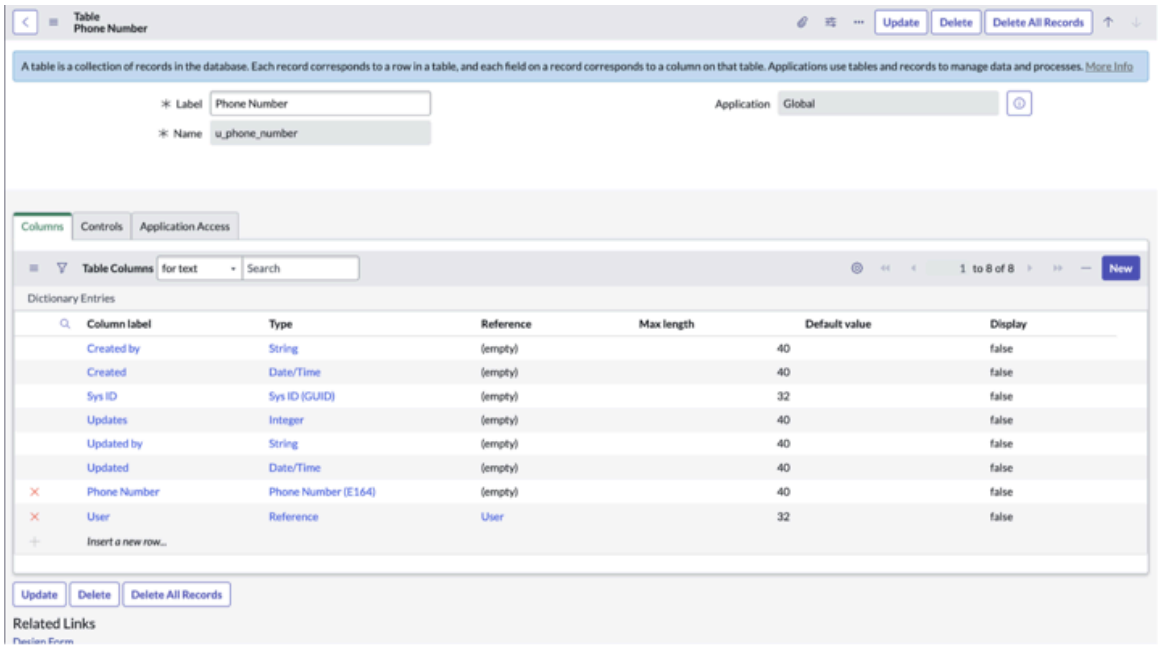
7. レコードを更新します。
8. [関連リンク] セクションで [変数の自動生成] をクリックします。
9. フォルダーで共有されている SI をインポートし、Vonage API ダッシュボードから apiKey とシークレットをコピーします。

   注:

   apiKey とシークレットは、Vonage ダッシュボードの Vonage アカウント設定にあります。



10. 次の例のように、2 つの列を持つカスタムテーブルの電話番号を作成します。
    - 列ラベル：ユーザー、タイプ：参照、参照：ユーザー (sys_user)。
    - 列ラベル：電話番号、タイプ：電話番号 (E164)。



11. マルチファクタープロバイダーテーブルにカスタムプロバイダーを作成します。

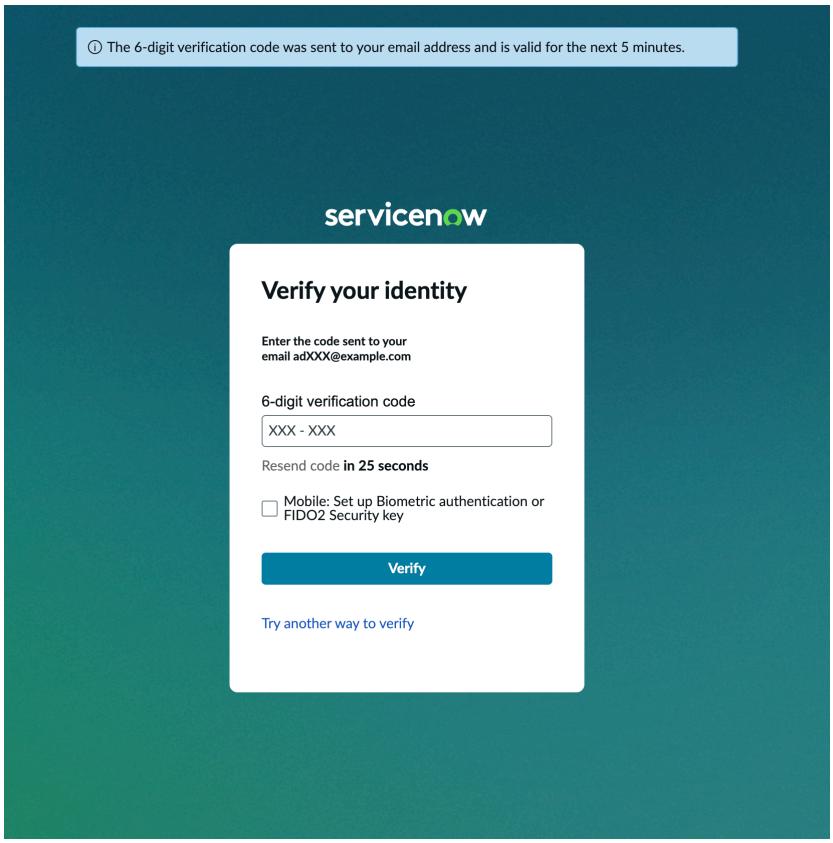
プロバイダー構成の詳細については、「MFA プロバイダーの構成」を参照してください。

MFA 要素としてのメール

認証要素としてメールを使用するマルチファクター認証 (MFA)。

アドミニストレーターは、インスタンスにログインしようとするユーザーにメールベースの OTP の使用を要求するように ServiceNow インスタンスを設定できます。

> **注:** メールを使用した MFA は、Integration - マルチファクター認証 (MFA) (com.snc.integration.multifactor.authentication) プラグインでデフォルトでアクティブ化されています。ポリシーの入力と条件を設定する必要があります。



ユーザーが ServiceNow にログインしようとすると、メール OTP が、関連付けられているメールアドレスに送信されます。ユーザーは、メールアドレスに送信された 6 桁の検証コードを入力して本人確認を行うことができます。

メールを **MFA** 要素として設定

認証の MFA 要素ポリシーとして メール OTP を表示するようにポリシー入力と条件を設定します。

始める前に

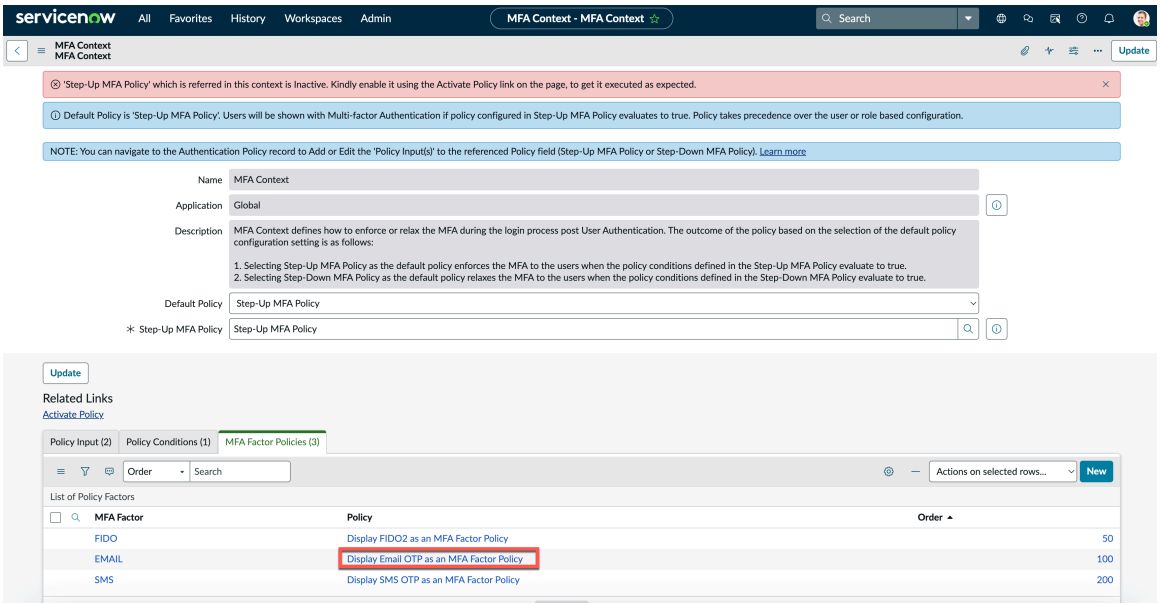
必要なロール：admin

手順

1. 移動先 すべて > 多要素認証 > **MFA** コンテキスト.
2. **[MFA** 要素ポリシー] タブをクリックします。

   注: デフォルトでは、ポリシー付きの **MFA** 要素としてメールを利用できますポリシーを編集し、ポリシーの入力と条件を指定できます。

3. [メール **OTP** を **MFA** 要素ポリシーとして表示 **(Display Email OTP as an MFA Factor Policy)]** を選択します。

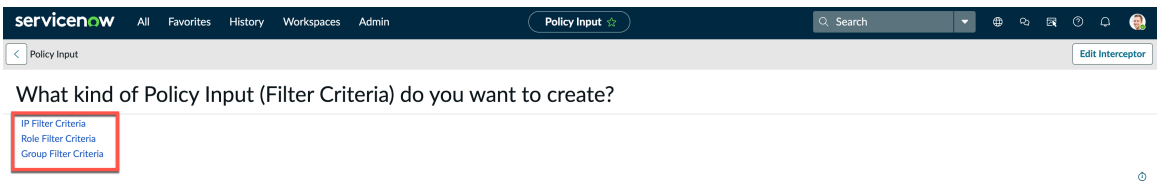


4. [新規] をクリックして、ポリシーの入力を追加します。
5. 作成するフィルター基準を選択します。

   フィルター基準のタイプは次のとおりです。

   - IP フィルター基準
   - ロールフィルター基準
   - グループフィルター基準

   例：ロールフィルター基準



6. [ロールフィルター基準] をクリックし、ロールフィルター基準のフィールドに入力してレコードを送信します。
   新しいポリシーが作成されます。詳細については、「ロールフィルター基準」を参照してください。
7. [ポリシー - メール OTP を MFA 要素ポリシーとして表示 (Display Email OTP as an MFA Factor Policy)] ページで、 [ポリシー条件] をクリックします。
8. [新規] をクリックして、ポリシー条件を追加します。
9. フォームのフィールドに入力します。

[条件] フォーム

フィールド	説明
ラベル	条件を識別する名前
説明	条件の説明
条件	認証要求を評価するために使用される複数のポリシー入力 (フィルター基準) の論理的な組み合わせ。 条件に対して作成されたロールベースのフィルター基準ポリシーを選択します。

10. [送信] をクリックします。
11. オプション: 追加のポリシー条件を作成するには、手順 8 を繰り返します。

> **注:** 複数のポリシー条件を作成する場合、アクセスポリシーの最終的な出力は、すべてのポリシー条件の論理 OR 出力によって決まります。これは、ポリシー条件のいずれかが満たされると、ポリシーが true と評価されることを意味します。

ロールフィルター (ユーザー) ポリシーに基づいて、ロールに指定された条件が一致すると、ユーザーの認証のオプションとしてメール MFA 要素が表示されます。

マルチファクター認証システムプロパティ

システムプロパティを使用すると、MFA を有効化してカスタマイズし、セキュリティ要件を満たすことができます。

マルチファクター認証プロパティ

プロパティ	説明
glide.authenticate.multifactor	マルチファクター認証を有効にします。
glide.authenticate.multifactor.setup.bypass.count	ユーザーが MFA のセットアップをスキップできる回数。デフォルトは 0 です。
glide.multifactor.onetime.code.validity	リセットコードが有効である時間 (分)。「マルチファクター認証によるログイン」を参照してください。デフォルトは 5 です。 **注:** このプロパティはメールの OTP 検証回数を表します。
glide.authenticate.multifactor.clock_skew	リセットコードを有効にする追加の秒数。最大値は 60 です。デフォルトは 10 です。 インスタンスは、現在時刻に生成された単一のアプリ生成コードに対して、ユーザーが入力したコードを検証します。このプロパティを使用して期間をスキューし、期間中に生成された 1 つ以上のコードが有効と見なされるようにすることができます。 プロパティの値は、次の計算で使用されます：「現在時刻 - X/2」および「現在時刻 + X/2」、

マルチファクター認証プロパティ (続く)

プロパティ	説明
	ここで「X」はこのプロパティの値です。たとえば、値 10 を使用すると、インスタンスは、「現在時刻 - 5 秒」から「現在時刻 + 5 秒」の時間範囲内にアプリが生成したコードを有効と見なします。 このプロパティを使用すると、ユーザーがデフォルトで割り当てられた時間内に正しいコードを入力できない場合のログインの問題を防ぐことができます。
glide.authenticate.multifactor.remember.browser.challenge	ユーザーが新しいデバイスまたはブラウザーからログインしたときに MFA のプロンプトを表示するようにインスタンスを設定します。デフォルトは [はい] です。
glide.authenticate.multifactor.browser.fingerprint.validity	MFA がブラウザーを記憶した後、ユーザーはこの期間、同じブラウザーで MFA から干渉を受けません。デフォルトは 8 時間です。
glide.authenticate.multifactor.remember.browser.max.count	このユーザーについて MFA が記憶しているブラウザーの数
glide.authenticate.multifactor.remember.browser.default	[マルチファクター検証] ページのブラウザーを保存するためのチェックボックスのデフォルト値
glide.webauthn.enabled)	ハードウェアキーや生体認証などのパスワードレス認証 (FIDO2 ベースの MFA) 方式を有効にするオプション。
glide.authenticate.multifactor.email.otp.enable	MFA の要素としてメールベースの OTP を有効にするオプション。
glide.auth.mfa.ui.v2.enabled	認証アプリを設定せずに、ユーザーが自分で MFA 要素を有効にするオプション。

マルチファクター認証基準

MFA 基準を使用して、2 段階認証を使用する必要があるユーザーとロールを決定します。

マルチファクター基準

MFA 基準を使用して、2 段階認証を使用する必要があるユーザーとロールを決定します。これらの基準は、ビジネスニーズに合わせて単独でも組み合わせても使用できます。これらの基準は、ビジネスニーズに合わせて単独でも組み合わせても使用できます。

注: 適応認証ポリシーベースの MFA を使用することをお勧めします。

ユーザーベースのマルチファクター基準

ユーザーベースのマルチファクター基準を使用して、MFA を使用したログインが必要なユーザーを個別に選択します。アドミニストレーターは、ユーザーレコードの [マルチファクター認証 **(MFA)** を有効にする] フィールドを更新して、ユーザーの MFA 要件を有効または無効にします。このプロセスの詳細については、「ユーザーベースのマルチファクター基準の設定」を参照してください。

ロールベースのマルチファクター基準

ロールベースのマルチファクター基準を使用して、特定のロールに割り当てられたすべてのユーザーに MFA ログインを要求します。マルチファクター基準 [multi_factor_criteria] テーブルの [ロールベースのマルチファクター認証 **(Role-based multi-factor authentication)]** レコードには、MFA ログインが必要なロールのリストが含まれています。このリストの管理の詳細については、「ロールベースのマルチファクター基準の設定」を参照してください。

適応認証ポリシーベースのマルチファクター基準

適応認証を使用して、インスタンスで MFA が必要な場合を判断します。適応認証では、認証ポリシーを使用して、ユーザーの IP アドレスやユーザーグループなどの基準を評価します。適応認証機能の詳細については、「適応認証」を参照してください。

ユーザーベースのマルチファクター基準の設定

ユーザーベースのマルチファクター基準を使用して、ユーザーの MFA を有効にします。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > ユーザー管理 > ユーザー.
2. [マルチファクター認証を有効にする] 列を表示するようにリストを構成します。
3. 選択したユーザーの [マルチファクター認証を有効にする] 列の値を **true** に変更します。

ユーザーがユーザー名とパスワードを使用してログインすると、マルチファクター認証を設定するように求められます。

ロールベースのマルチファクター基準の設定

ロールベースのマルチファクター基準を使用して、特定のロールに割り当てられたすべてのユーザーにマルチファクター認証を適用します。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > 多要素認証 > 多要素基準.
2. [マルチファクター基準] リストで、[ロールベースのマルチファクター認証 **(Role-based multi-factor authentication)]** レコードを開きます。
3. [マルチファクターロール] リストを使用して、ロールを追加または削除します。

オプション	説明
ロールを追加	[新規行を挿入...] をダブルクリックし、ロール名を入力または選択します。保存アイコン () をクリックしてエントリを保存します。
ロールを削除	削除アイコン () をクリックして、リストからロールを削除します。

4. [更新] をクリックします。

結果
インスタンスは、[マルチファクターロール] リストに記載されているロールのメンバーであるユーザー全員に対してマルチファクター認証を適用します。

重要: ロールベースのマルチファクター認証を適用するには、レコードをアクティブにする必要があります。

適応認証ポリシーベースのマルチファクター基準の設定

適応ポリシーを使用して、2 ステップマルチファクター (MFA) 検証の使用が必要なユーザーを決定します。

始める前に
必要なロール：admin

注:

- デフォルトのポリシーがステップアップ **MFA** ポリシーの場合、ステップアップ **MFA** ポリシーで構成されたポリシーが true と評価されると、ユーザーにはマルチファクター認証が表示されます。ポリシーは、ユーザーまたはロールベースの構成よりも優先されます。
- SSO ログインを使用する MFA は、glide.authenticate.mfa.with.multisso.enabled プロパティが true に設定されている場合にのみ使用できます。
- [認証ポリシー] レコードに移動して、参照されているポリシーフィールド ([ステップアップ **MFA** ポリシー] または [ステップダウン **MFA** ポリシー]) に「ポリシー入力」を追加または編集します。

手順

1. 移動先 すべて > 適応認証 > 認証ポリシーのコンテキスト > **MFA** コンテキスト.
   [MFA コンテキスト] ポリシーコンテキストレコードが開きます。
2. [デフォルトポリシー] フィールドでデフォルトポリシーを選択します。
   この選択により、MFA が必要かどうか判断するために、インスタンスがポリシー条件を使用する方法が決まります。

   デフォルトポリシー

デフォルトポリシー	定義
ステップアップ MFA ポリシー	選択すると、[ステップアップ **MFA** ポリシー] で定義されたポリシー条件が true と評価された場合に MFA が適用されます。
ステップダウン MFA ポリシー	選択すると、[ステップダウン **MFA** ポリシー] で定義されたポリシー条件が true と評価された場合に MFA がバイパスされます。

3. [ステップアップ **MFA** ポリシー] フィールドで、このコンテキストで使用するポリシーを選択します。
4. [更新] をクリックします。
   レコードを保存した後、[ポリシー入力] と [ポリシー条件] のリストが更新され、[ステップアップ **MFA** ポリシー] フィールドで選択されたポリシーに関連付けられているポリシー入力と条件が表示されます。

シングルサインオンによるマルチファクター認証

ServiceNow インスタンスの SSO プロバイダーで MFA を使用できます。

組織内では多くのユーザーが企業ネットワークにログインして ServiceNow インスタンスにアクセスしているため、安全な認証の必要性はますます強くなっています。

MFA と SSO の組み合わせにより、インスタンスのセキュリティが強化されます。この機能は、ユーザーに対して条件付きで MFA を有効にする柔軟性を提供します。

注:

- ServiceNow MFA は、ID プロバイダーでの認証成功後にユーザーがリダイレクトされた後で適用されます。
- San Diego リリースより前の SSO では、MFA は適用されませんでした。

たとえば、外部ユーザーに追加のセキュリティプロトコルを付与する場合は、それらのユーザーにのみ MFA を適用できます。このようにして追加の認証機能を追加して、ユーザーアクセスを制御できます。

SAML、OpenID Connect、Digest などの SSO ベースのログインでは、認証に MFA を適用できます。

SSO を使用した MFA は、要件に基づいてオンデマンドで構成できます。認証スキームと ID プロバイダーを使用すると、特定のログインメカニズムで特定のユーザーに MFA を適用できます。

次の条件に対して MFA を適用できます。

- 認証スキーム
- ID プロバイダー

SSO を使用する MFA は、適応認証プラグイン (com.snc.adaptive_authentication) の一部として提供されます。適応認証の設定方法の詳細については、「適応認証」を参照してください。

SSO を使用した MFA の構成

組織内外のユーザーに対して、SSO で MFA を適用します。

始める前に

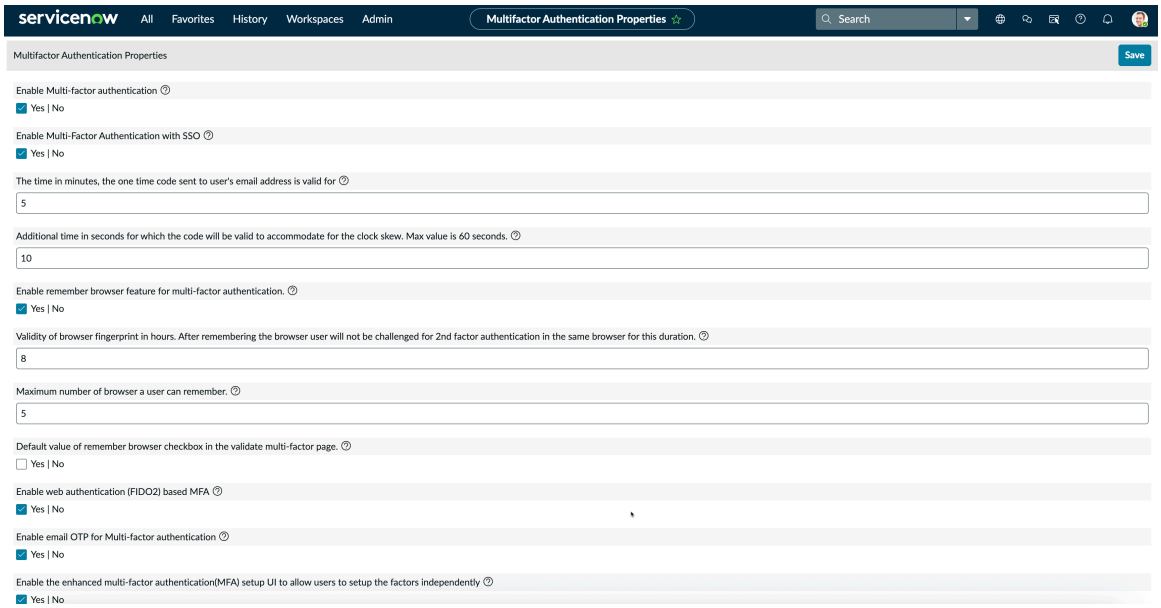
SSO 機能を使用する MFA は、適応認証プラグイン (com.snc.adaptive_authentication) の一部として提供されます。SSO 機能とともに MFA を使用するには、[適応認証] プロパティを有効にする必要があります。適応認証の設定方法の詳細については、「適応認証」を参照してください。

注: `glide.authenticate.mfa.with.multisso.enabled` プロパティが **true** に設定されている場合は、SSO ログインを使用する MFA を使用できます。

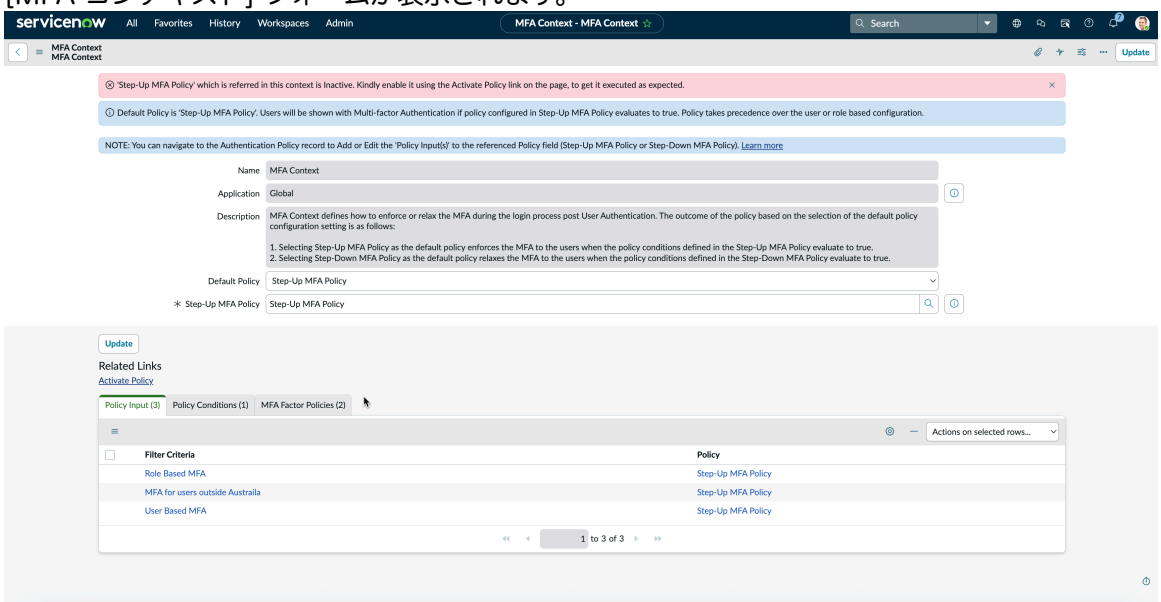
必要なロール：admin

手順

1. 移動先 すべて > 多要素認証 > プロパティ.
2. [マルチファクター認証を有効にする] および **[SSO** を使用したマルチファクター認証を有効にする] チェックボックスをオンにします。



3. **[Save** (保存)] をクリックします。
4. 移動先 多要素認証 > **MFA** コンテキスト.
   [MFA コンテキスト] フォームが表示されます。

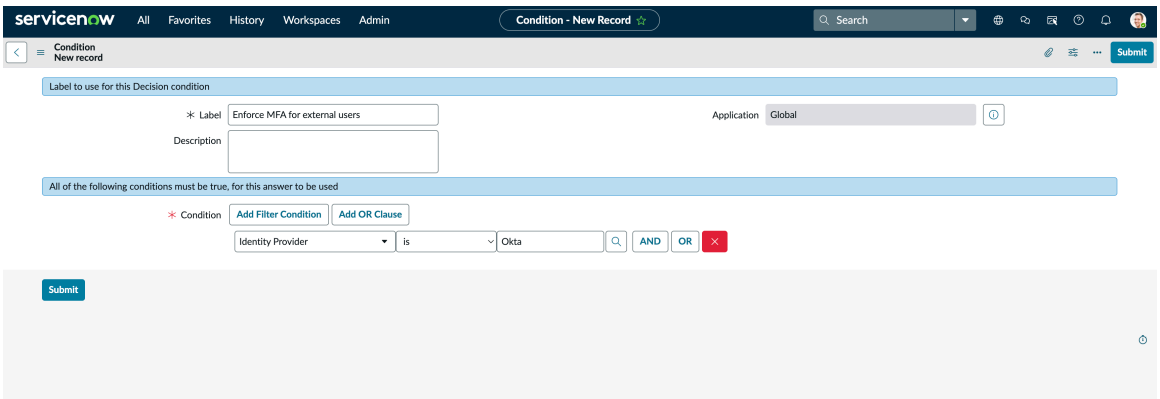


> **注:** デフォルトでは、ポリシーは [ステップアップ **MFA** ポリシー] です。[ステップアップ **MFA** ポリシー] で設定された条件が **true** と評価された場合、ユーザーにはマルチファクター認証が表示されます。ポリシーは、ユーザーまたはロールベースの構成よりも優先されます。

5. オプション: ポリシーを編集するには、[認証ポリシー] レコードに戻り、条件を変更してから戻ります。
   [認証ポリシー] レコードに移動する際に、参照されているポリシーフィールド ([ステップアップ **MFA** ポリシー] または [ステップダウン **MFA** ポリシー]) に対して「ポリシー入力」を追加または編集できます。
6. 新しいポリシー条件を設定するには、[ポリシー条件] をクリックします。
7. [新規] をクリックします。
8. フォームのフィールドに入力します。

[条件] フォーム

フィールド	説明
ラベル	ラベルに対して作成する条件の一意の名前
説明	ポリシー条件の説明
条件	ポリシーに適用する条件のタイプ。フィルター条件と「OR」節を追加できます。 **注:** さまざまなフィルター条件と節を追加すると、特定のユーザーに MFA を要求することができます。



Example
条件を設定するには、次の例を検討してください。外部ユーザーの条件として ID プロバイダーを使用する認証スキームを設定したいとします。条件は次のように設定できます。

a. [認証スキーム] を選択し、基準を [シングルサインオン] または [ユーザー名とパスワード] として設定します。

この選択に基づいて、SSO ベースのログインまたはユーザー名とパスワードを指定するためのログインフォームがユーザーに表示されます。

b. **[ID** プロバイダー] を選択し、MFA を有効にする IDP レコードとしてプロバイダーを指定します。たとえば、**Okta** などです。

この選択に基づいて、ユーザーが SSO を使用してログインした場合、MFA からのチャレンジはありません。逆に、ユーザーが Okta を使用してログインすると MFA からのチャレンジがあります。

9. [送信] をクリックします。

関連トピック

適応認証

シングルサインオンによるマルチファクター認証

ユーザーのマルチファクター認証 **(MFA)** をリセットする

アドミニストレーターは、アプリを削除したユーザー、デバイスへのアクセス権を失ったユーザー、またはデバイスに代替 MFA が関連付けられていないユーザーの MFA をリセットできます。

始める前に
必要なロール：admin

次の手順では、ServiceNow® アドミニストレーターが MFA 検証をリセットしてユーザーのブロックを解除し、ユーザーが MFA を再登録できるようにする方法について説明します。

手順

1. 移動先 すべて > 多要素認証 > ユーザー多要素設定.
2. ブロックを解除するユーザーを検索します。
3. [検証] を **false** に設定します。
4. 移動先 すべて > 多要素認証 > **Web** 認証 > ユーザーの公開認証情報 **(sys_user_public_credential**.
5. ブロックを解除するユーザーを検索します。
6. このユーザーのすべてのレコードを削除します。
7. 移動先 すべて > 多要素認証 > ユーザーの最近のユーザー要素.
8. 同じユーザーを検索します。
9. このユーザーのすべてのレコードを削除します。

結果
ブロック解除されたユーザーが認証情報を入力してログインすると、[マルチファクター認証 **(MFA)** を有効にする] ページが表示されます。ユーザーは、ページの手順に従って MFA を再登録できます。

参照トピック - マルチファクター認証

MFA の構成に関連する参照トピック。

ユーザーマルチファクター認証

ユーザーの MFA の詳細にアクセスするには、 すべて > 多要素認証 > ユーザー多要素設定.

ユーザーマルチファクター認証

フィールド	説明
ユーザー	ユーザーのユーザー名
残りのバイパス	ユーザーに残っているバイパスの合計数。
マルチファクターシークレット	マルチファクターシークレットの詳細。
検証済み	マルチファクターが検証済みかどうか。値を「**false**」に変更すると、既存の認証アプリのセットアップが無効になります。

注: ユーザーが最近使用した MFA 要素は、[ユーザーが最近使用した要素 (User Recent Used Factors)] モジュールで利用できます。移動先 すべて > 多要素認証 > ユーザーの最近のユーザー要素. 詳細については、「ユーザーが最近使用した要素」を参照してください。

多要素ブラウザ指紋認証

ユーザーのブラウザーの指紋認証に関する詳細にアクセスするには、 すべて > 多要素認証 > 多要素ブラウザ指紋認証.

多要素ブラウザ指紋認証

フィールド	説明
ユーザー	ユーザーのユーザー名
ブラウザー	ユーザーが使用したブラウザー。
ブラウザーの指紋認証	ブラウザーの指紋認証の詳細。
ブラウザーの指紋認証 cookie	ブラウザーの指紋認証 cookie の詳細。
ブラウザーの指紋認証 cookie の有効期限	ブラウザーの指紋認証 cookie の有効期限の詳細。

ユーザーが最近使用した要素

ユーザーが最近使用した要素の詳細にアクセスするには、次に移動します: すべて > 多要素認証 > ユーザーの最近のユーザー要素.

ユーザー多要素セットアップ

フィールド	説明
ユーザー	ユーザーのユーザー名
マルチファクタータイプ	ユーザーがログインに使用したマルチファクタータイプ。
最新のファクターである	ユーザーの最近のマルチファクターの詳細。

MFA メトリクス

さまざまな MFA メトリクスを表示して、MFA の採用と使用状況を把握します。

MFA 関連のメトリクスはセキュリティセンターで利用できます。Security Center は、ServiceNow Store からダウンロードできる無料のアプリケーションです。Security Center の詳細については、「 セキュリティセンターのランディングページ」を参照してください。

Security Center で、ページの上部にあるバーを使用して Security Monitoring コンソールのセクション間を移動し、[**Security Metrics]** タブを選択します。

MFA メトリクスは次のとおりです。

- MFA に登録されているユーザー：MFA の使用に登録されているユーザーの合計数
- MFA バイパスを使用しているユーザー (Users using MFA bypass)：マルチファクター認証を回避しているユーザーの合計数
- 高権限非 MFA ユーザー (High privileged non-mfa user)：MFA を使用していない高権限ユーザーの合計数
- アクティブな MFA ユーザー：インスタンスでアクティブな MFA ユーザーの合計数
- ロックアウトされた MFA ユーザー：インスタンスでロックアウトされた MFA ユーザーの合計数
- MFA で保護されていないローカルログイン (Local logins not protected by MFA)：MFA なしでログインしたユーザー

セキュリティメトリクスの詳細については、「 セキュリティ測定基準」を参照してください。

マルチファクター認証の使用

マルチファクター認証ツールを使用してインスタンスに安全にアクセスする方法について説明します。

MFA でログイン

ServiceNow では、時間ベースのワンタイムパスワード (TOTP) をサポートする認証アプリケーションが必要です。ServiceNow は次の認証システムで MFA をテストします。

- Google 認証システム
- Microsoft 認証システム
- LastPass 認証システム
- Authy
- FreeOTP
- Duo
- Okta Verify

注:

- リストにない他の認証システムも適合する可能性がありますが、ServiceNowではテストされていません。
- ブラウザ固有の動作の変更に関連する情報については、このナレッジベース記事 を参照してください。

認証アプリの登録

認証アプリ 認証アプリを使用して、認証の第 2 要素を使用できます。	アドミニストレーターがインスタンスでマルチファクター認証 (MFA) を有効にしている場合は、ユーザー名とパスワードを入力した後に 2 番目の認証が要求されます。MFA ログインプロセスの詳細については、「 マルチファクター認証によるログイン」を参照してください。
認証アプリによる検証 認証アプリに表示されたコードを入力してログインします。	2 番目の認証フォームを設定していない場合は、ログイン後に設定ページが表示され、認証アプリの設定手順が示されます。このセットアップの詳細については、「マルチファクター認証を初めて設定する」を参照してください。

自動翻訳

認証デバイスの登録

認証アプリを構成した後、他の認証方法を登録できます。

生体認証装置

2 番目の MFA 認証として、指紋認証や顔認識などの生体認証を使用できます。アドミニストレーターがこのオプションを許可している場合は、「生体認証装置の登録」の手順を使用して生体認証装置を設定できます。

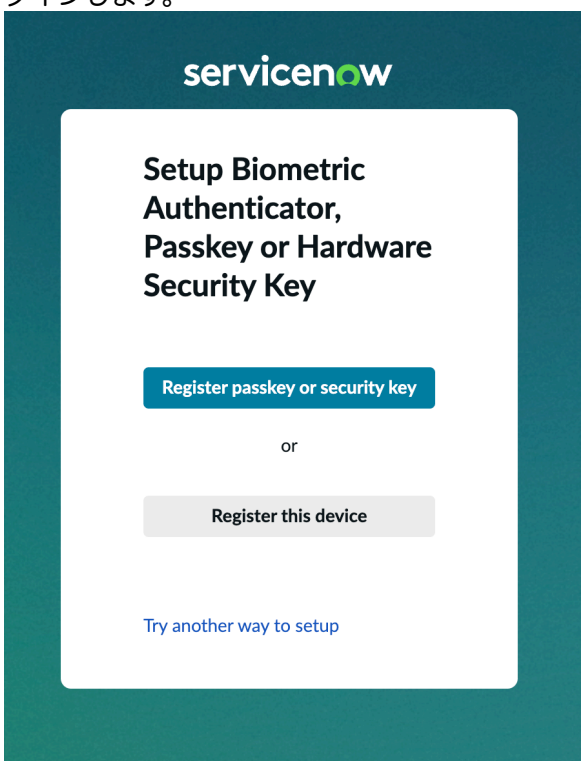
ハードウェアキー認証システム

ハードウェアキーは、認証に使用できる物理的なセキュリティデバイスです。「ハードウェアセキュリティキーの登録」の手順を使用して、インスタンスで使用するハードウェアデバイスを登録できます。

生体認証またはハードウェアキーを使用した検証

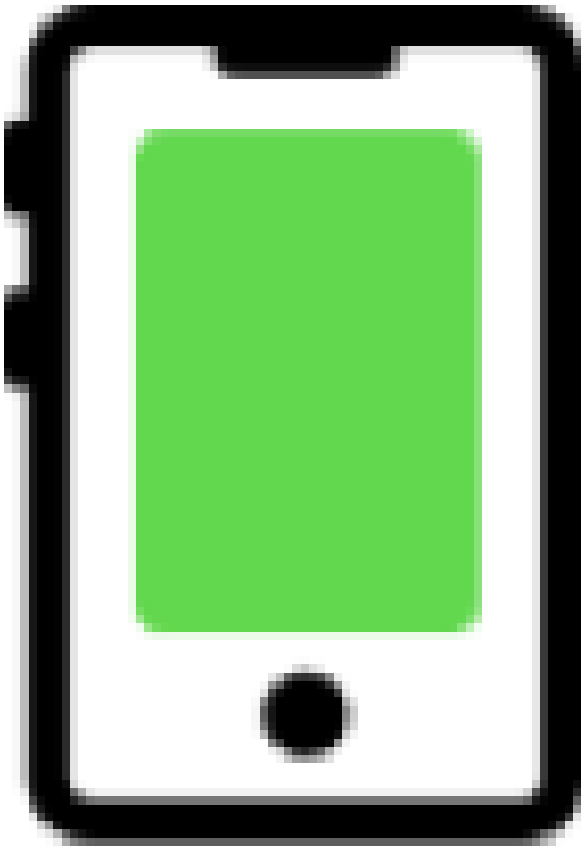
生体認証またはハードウェアキーを使用してログインします。

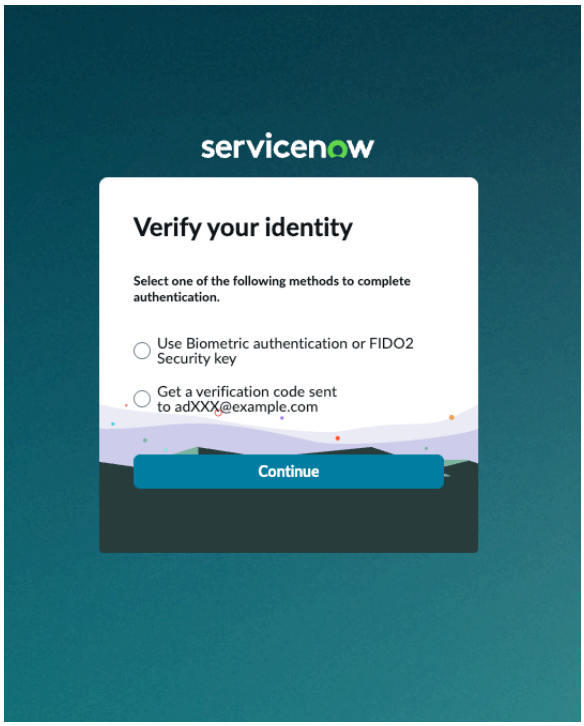
生体認証またはセキュリティキーを使用してログインします。



自動翻訳

OTP 用の電話番号の登録

SMS SMS ベースの OTP を使用して、ログインを試みるユーザーに要求します。	ユーザーが ServiceNow にログインしようとすると、sys_user レコードに関連付けられた携帯電話番号に SMS OTP が送信されます。ユーザーは、モバイルデバイスに送信された 6 桁の検証コードを入力して本人確認を行うことができます。 
SMS を使用した検証 SMS による検証を使用し、生成された OTP に基づいてログインします。	携帯電話番号に送信された 6 桁のコードを入力してログインします。送信されたコードは 5 分間有効です。コードの再送信を使用してコードを再送信できます。



OTP のメールアドレスの登録

メールアドレス

メールベースの OTP を使用して、ログインを試みるユーザーに要求します。

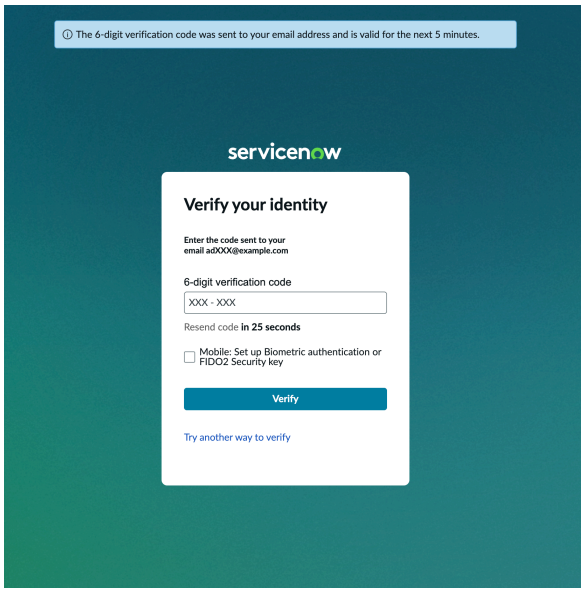
ユーザーが ServiceNowにログインしようとすると、メール OTP がユーザーに関連付けられているメールアドレスに送信されます。ユーザーは、モバイルデバイスに送信された 6 桁の検証コードを入力して本人確認を行うことができます。

メールを使用した検証

生成された OTP に基づいてログインするには、メールでの検証を使用します。

メールアドレスに送信された 6 桁のコードを入力してログインします。送信されたコードは 5 分間有効です。コードの再送信を使用してコードを再送信できます。

自動翻訳



マルチファクター認証を初めて設定する

アドミニストレーターがプロファイルで MFA を有効にしていても、アプリケーションをまだ設定していない場合は、ログイン時に設定できます。

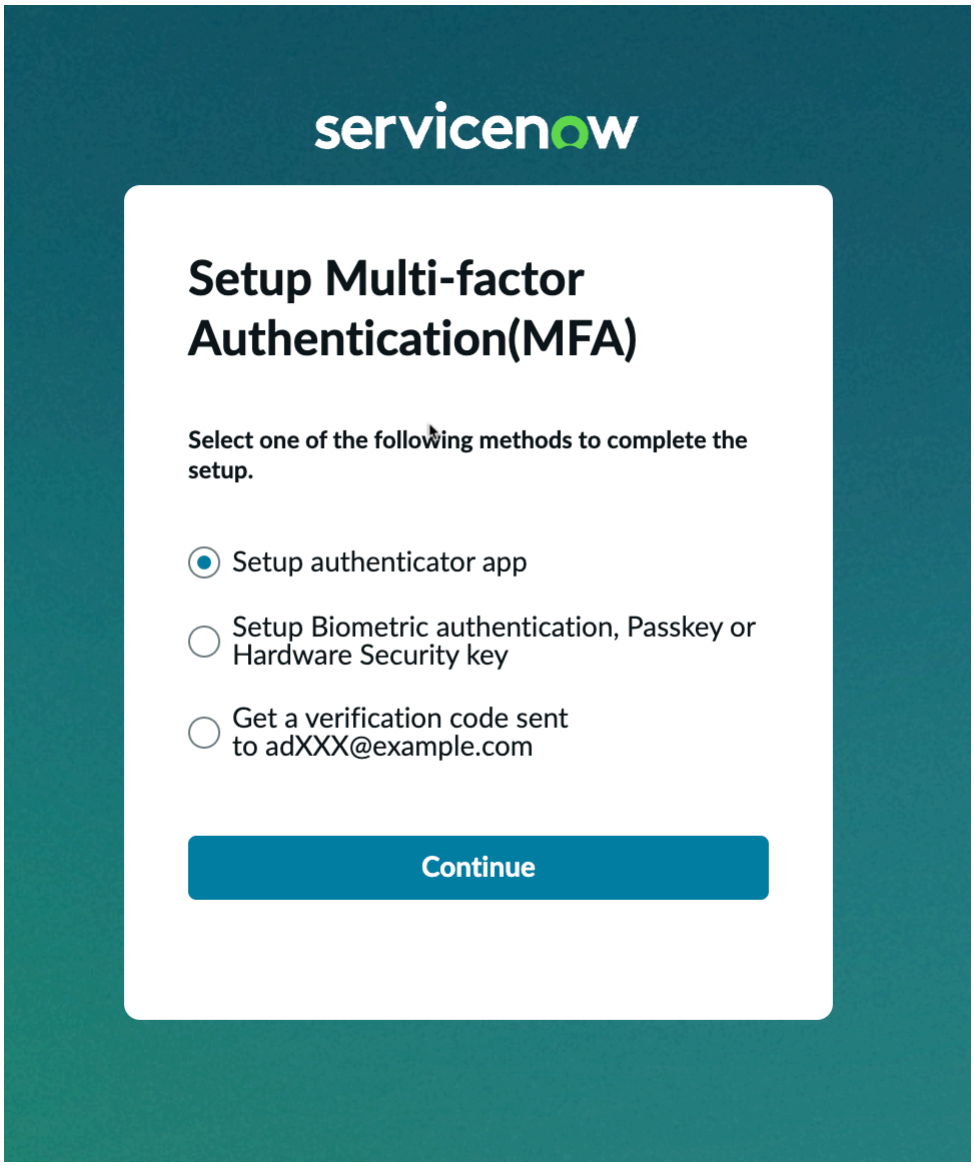
始める前に

必要なロール：なし

手順

1. ユーザー名とパスワードを使用してインスタンスにログインします。

   マルチファクター認証の設定画面がログインをインターセプトします。



注: ここで設定した認証をスキップする場合は、[バイパスセットアップ] をクリックします。アドミニストレーターが許可する限られた回数だけ、マルチファクター認証をバイパスできます。最終的には、マルチファクター認証を構成する必要があります。

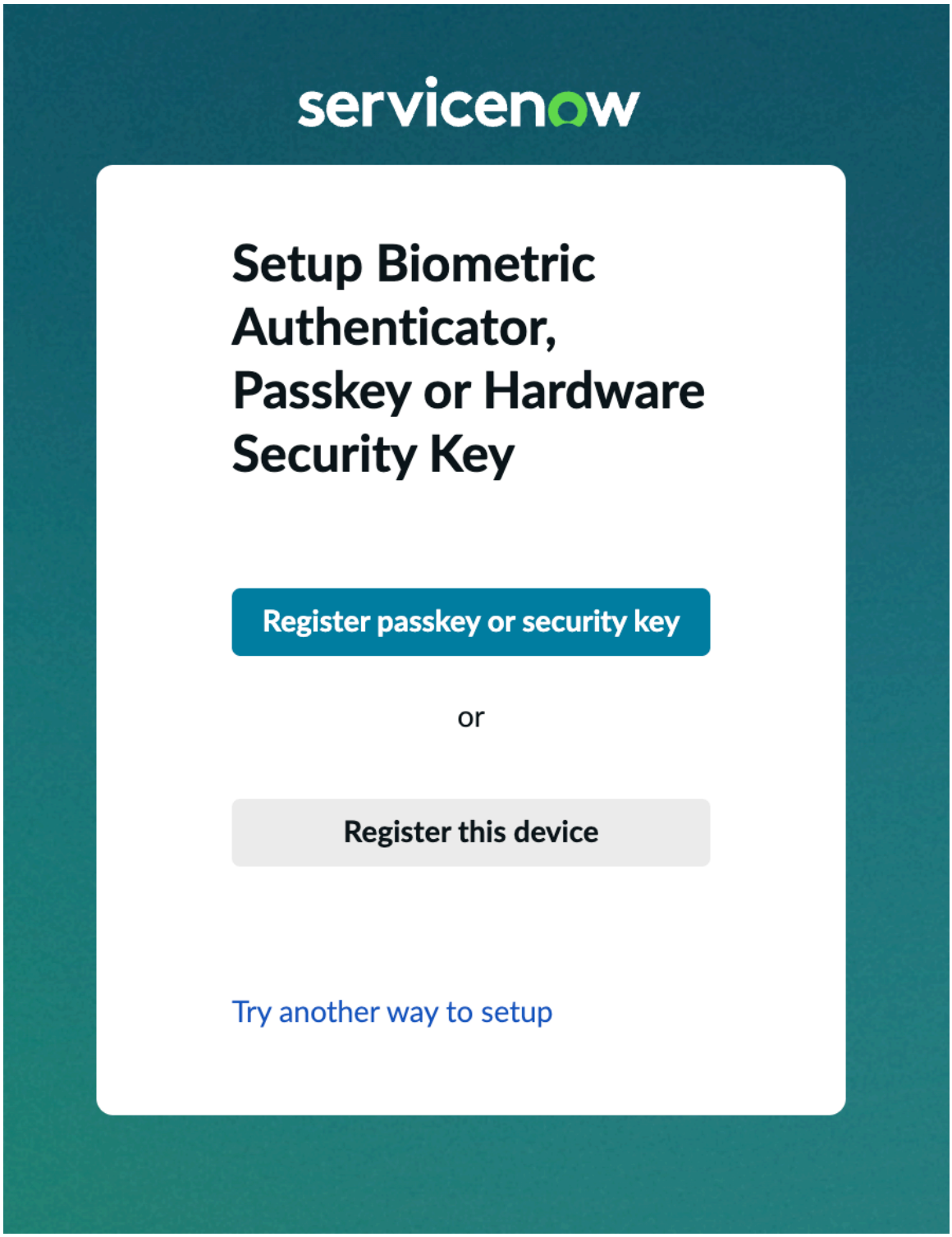
2. 次のいずれかの方法を選択して、MFA セットアップを完了します。

a. 認証アプリをセットアップ

画面の指示に従って、デバイスをペアリングしてログインします。

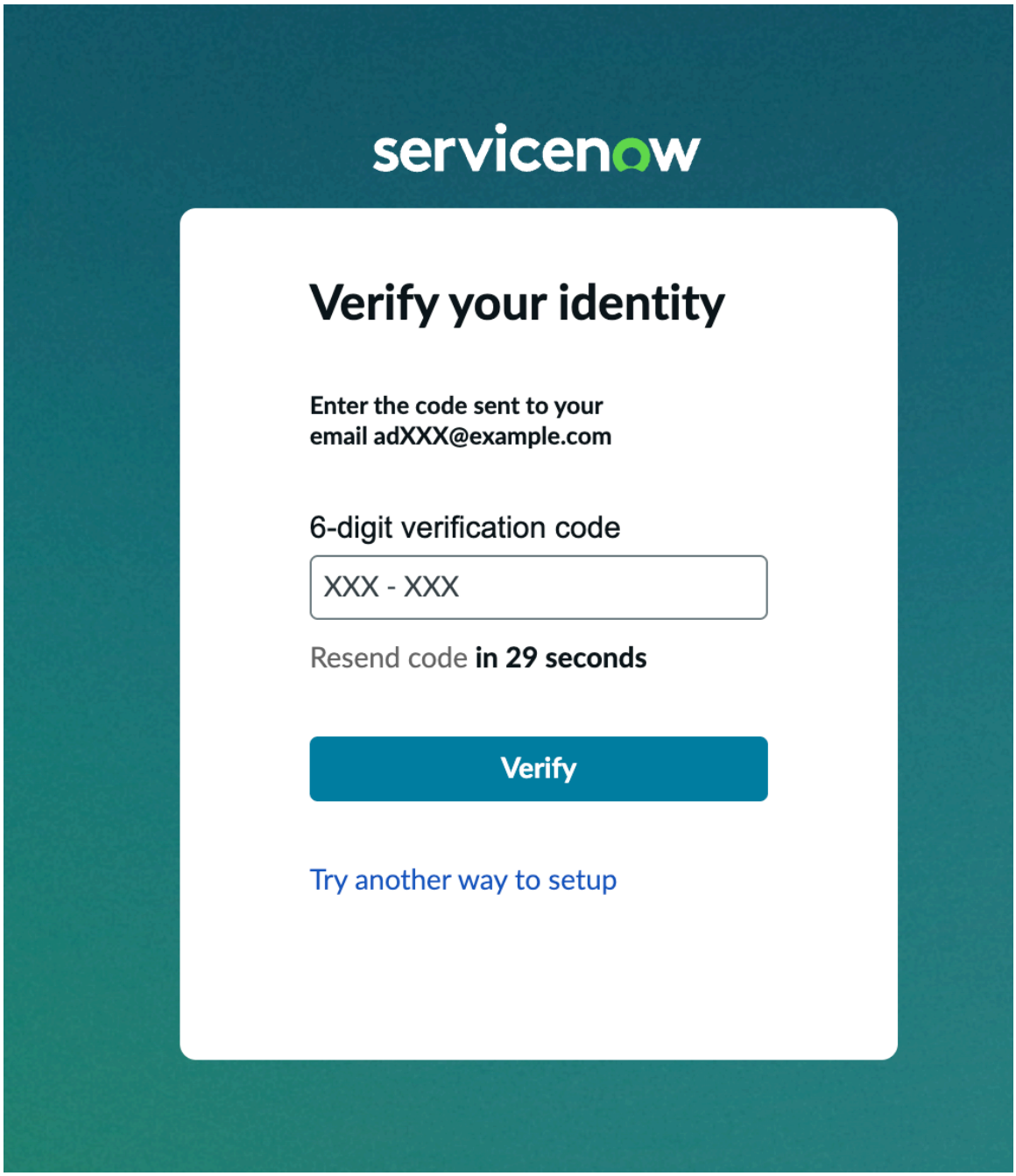
b. 生体認証、パスキー、またはハードウェアセキュリティキーのセットアップ

いずれかのオプションを選択してセットアップを完了します。



c. メールに送信された検証コードの取得

メールに送信された検証コードを入力します。



いずれかのセットアップが正常に完了すると、インスタンスにログインします。

ユーザープロファイルでのマルチファクター認証の設定

ユーザープロファイル設定で、 アカウントのマルチファクター認証を有効にします。

始める前に
必要なロール：なし

インスタンスでマルチファクター認証を有効にする必要があります。

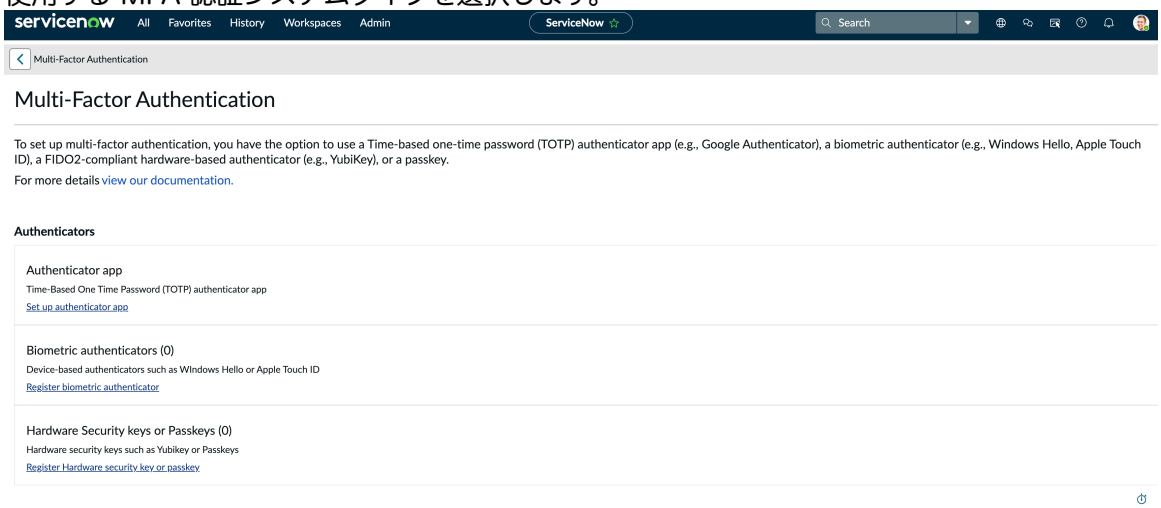
注: アドミニストレーターがマルチファクター認証の使用を要求する場合があります。この場合、ログインすると自動的にプロンプトが表示されます。「マルチファクター認証を初めて設定する」を参照してください。アドミニストレーターがマルチファクター認証のオプトインを許可している場合は、以下のプロセスを使用します。

手順

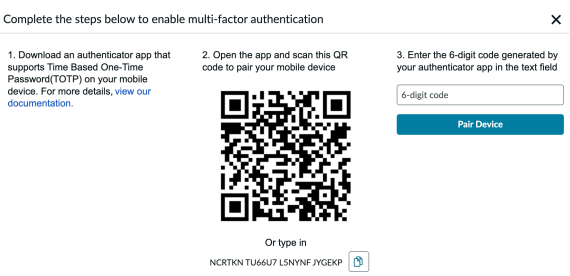
1. 移動先 すべて > セルフサービス > プロファイル.

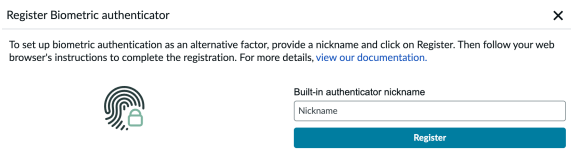
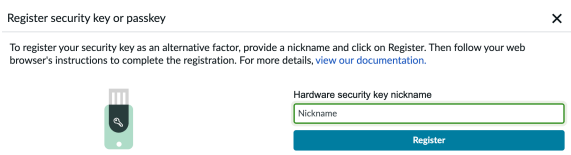
   注: インスタンスヘッダーのユーザー名をクリックして、プロファイルにアクセスすることもできます。

2. ユーザープロファイルで、[関連リンク] セクションの [マルチファクター認証の構成] をクリックします。
3. 使用する MFA 認証システムタイプを選択します。



MFA 認証システム

MFA 認証システム	説明
認証アプリ	時間に基づくワンタイムパスワード (TOTP) 認証アプリ [認証アプリを設定] を選択し、画面の指示に従って、認証アプリを認証の第 2 要素として登録します。 
生体認証装置	Windows Hello や Apple Touch ID などのデバイスベースの認証機能

MFA 認証システム	説明
	[生体認証を登録 **(Register biometric authentication)**] を選択し、画面の指示に従って、認証アプリを認証の第 2 要素として登録します。 
ハードウェアセキュリティキー	YubiKey や Google Titan Key などのハードウェアセキュリティキー [ハードウェアセキュリティキーを登録] を選択し、画面の指示に従って、認証の第 2 要素として認証アプリに登録します。 

結果
ユーザーアカウントでマルチファクター認証が有効になっています。次回ログイン時に、マルチファクター認証を使用するように求められます。

注: アカウントでマルチファクター認証を有効にした後で、無効にすることはできません。マルチファクター認証を無効にする必要がある場合は、アドミニストレーターにお問い合わせください。

マルチファクター認証によるログイン

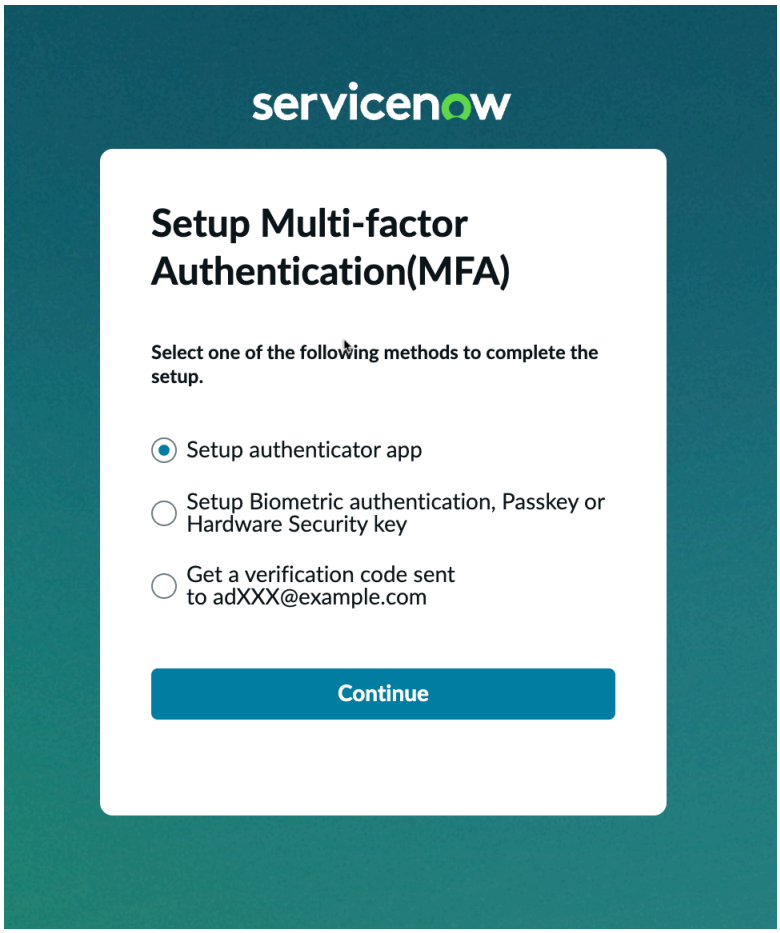
アドミニストレーターがインスタンスで MFA を有効にしてから MFA を使用してログインします。

始める前に
プロファイルの MFA を有効にする必要があります。ユーザープロファイルで自分で有効にすることも、アドミニストレーターが有効にすることもできます。
必要なロール：なし

注: 最後に使用された MFA 検証要素は、後続のログイン時にユーザーに自動的に表示されます。

手順

1. インスタンスの URL に移動して、ログイン画面を開きます。
2. ユーザー名とパスワードを入力します。
3. [ログイン] をクリックします。
   マルチファクター認証画面が表示されます。



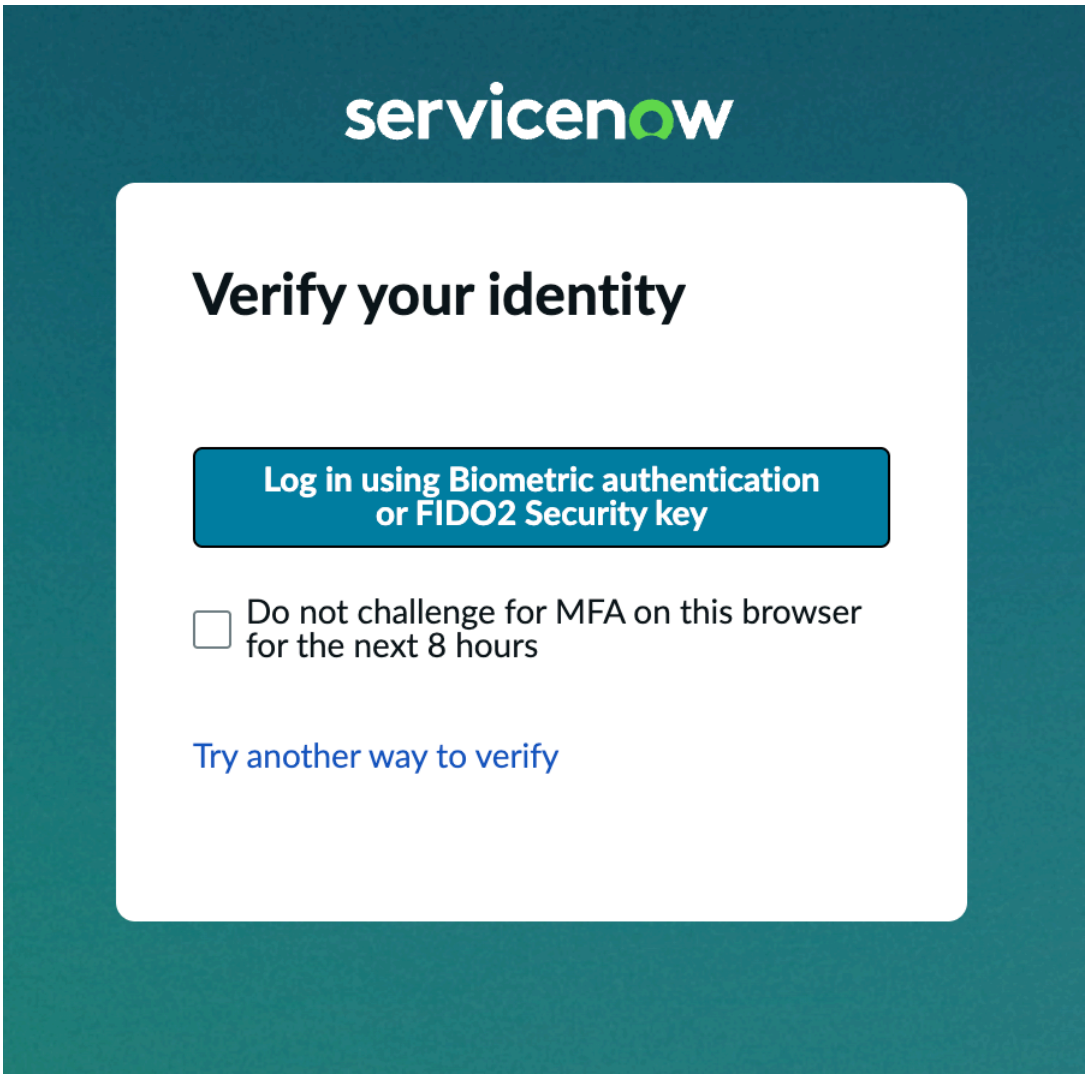
4. いずれかの方法を選択して、MFA セットアップを完了します。
   個々のセットアップの詳細については、「マルチファクター認証を初めて設定する」を参照してください。

   セットアップを延期する場合は、[セットアップを延期] を選択します。セットアップを延期できる最大回数が画面に表示されます。この値はアドミニストレーターによって設定されています。

結果
MFA を既にセットアップしている場合は、ユーザー名とパスワードを入力してインスタンスにログインすると、最近使用した要素が表示されます。

Example: 最近使用した **MFA** 要素

最近使用した要素が生体認証ログインの場合、ユーザー名とパスワードを使用してログインすると、生体認証 MFA セットアップ画面が直接表示されます。



認証アプリケーション

サードパーティの認証アプリケーションを使用して、一時的な MFA パスコードを生成します。

認証アプリケーションは、一時的なパスコードを生成するサードパーティソフトウェアです。これらのパスコードをパスワードとともに使用して、マルチファクター認証 (MFA) を必要とするインスタンスにログインできます。

アドミニストレーターがインスタンスで MFA を有効にしている場合は、ログイン時にユーザーとパスワードを入力するとパスコードの入力が求められます。

ServiceNow では、時間ベースのワンタイムパスワード (TOTP) をサポートする認証アプリケーションが必要です。

ServiceNow は次の認証システムで MFA をテストします。

- Google Authenticator
- Microsoft 認証システム
- LastPass 認証システム
- Authy
- FreeOTP
- Duo
- Okta Verify

注: リストにない他の認証システムも適合する可能性がありますが、ServiceNowではテストされていません。

認証アプリの変更

デバイスの認証アプリを変更するための新しいコードを生成します。

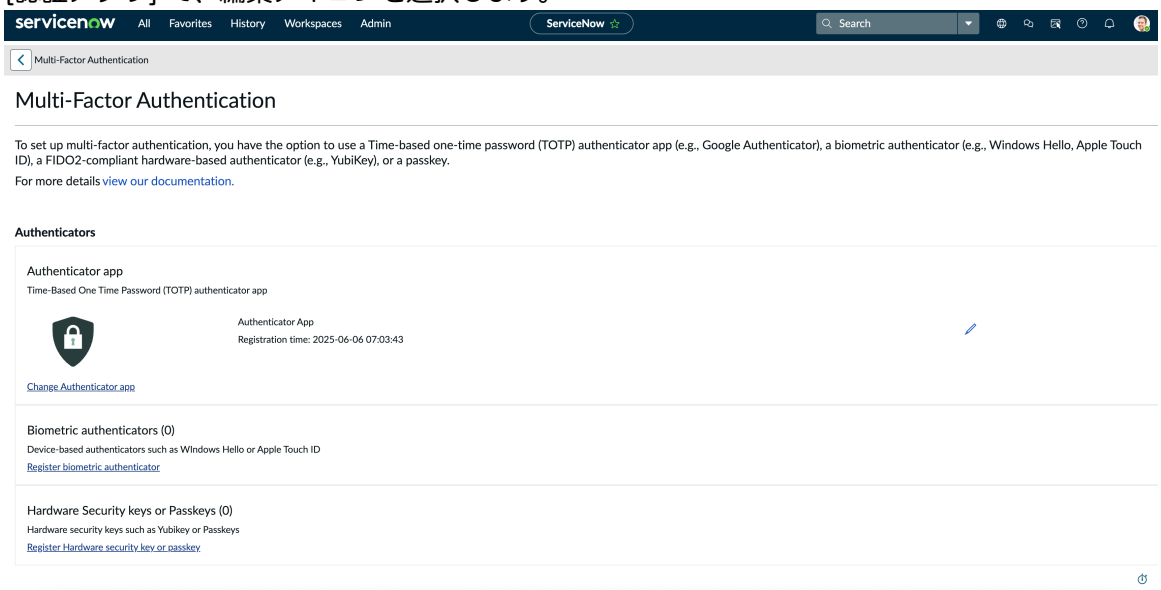
始める前に
プロファイルの MFA を有効にする必要があります。ユーザープロファイルで自分で有効にすることも、アドミニストレーターが有効にすることもできます。
必要なロール：なし

手順

1. 移動先 すべて > セルフサービス > プロファイル.

   注: インスタンスヘッダーのユーザー名をクリックして、プロファイルにアクセスすることもできます。

2. ユーザープロファイルで、[関連リンク] セクションの [マルチファクター認証] をクリックします。
3. [認証アプリ] で、編集アイコンを選択します。



4. [認証コードを変更] ウィンドウで、[変更] をクリックします。
5. 画面の指示に従って、認証アプリをデバイスに登録します。

Web 認証

ユーザーは、ハードウェアキーまたはデバイスの生体認証リーダー (FIDO2) を使用して、インスタンスに対する認証を行うことができます。

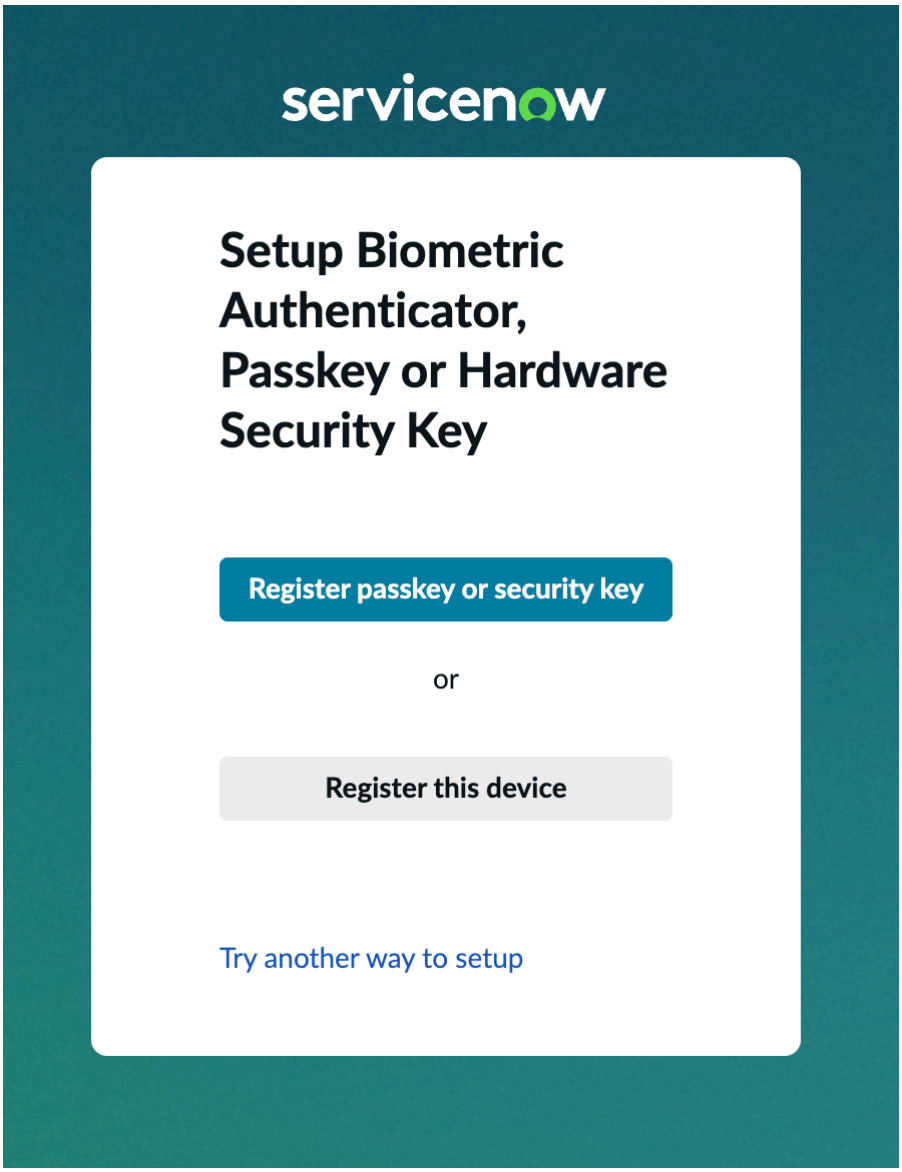
ハードウェアキー

ハードウェアキーは、認証に使用できる物理ハードウェアです。ハードウェアキーをデバイスのポートに挿入して認証します。ハードウェアキーの登録の詳細については、「ハードウェアセキュリティキーの登録」を参照してください。

生体認証

生体認証装置は、指紋または顔認識を使用してユーザーを識別します。ユーザーは、マルチファクターログインプロセスの一部としてデバイスでこれらの認証システムを使用できます。生体認証装置の登録の詳細については、「生体認証装置の登録」を参照してください。

インスタンスに対する認証を行う第 2 要素を選択します。



Web 認証プラグインを設定するには、「」を参照してください。

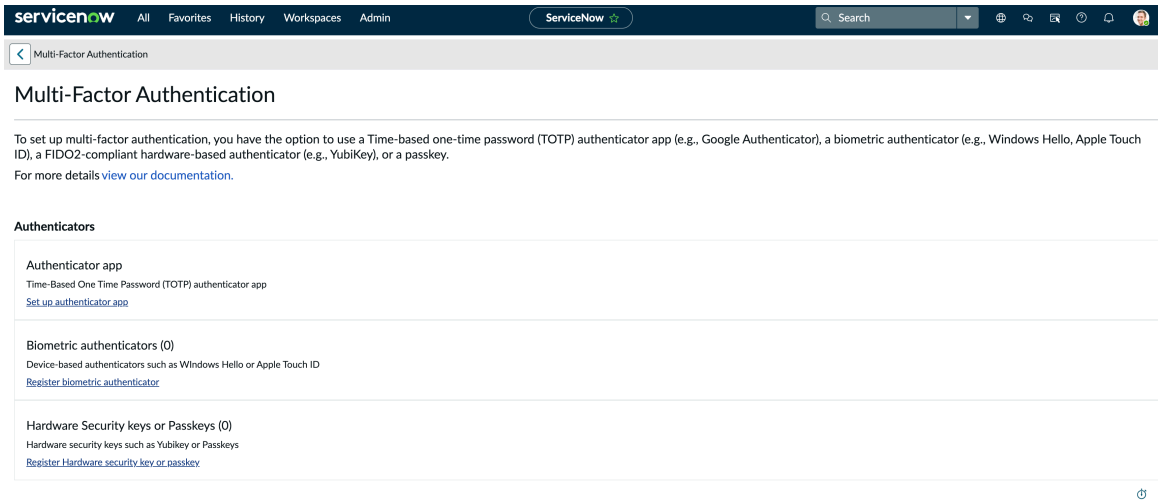
生体認証装置の登録

MFA ログインの一部として使用する生体認証装置を登録します。

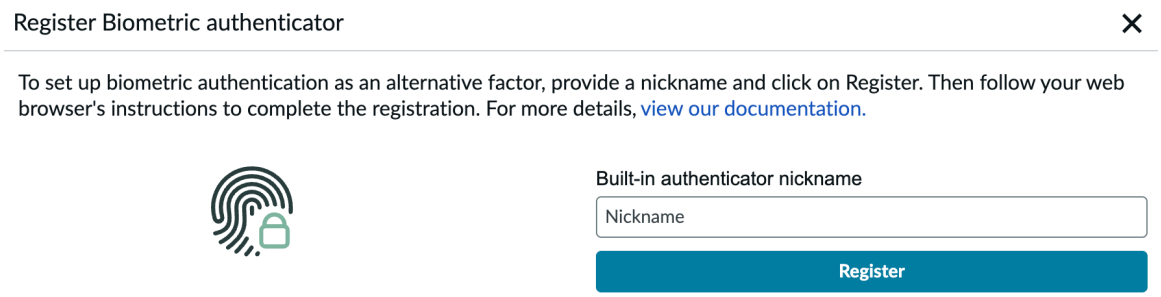
始める前に
必要なロール：なし

手順

1. 移動先 すべて > セルフサービス > プロファイル.
2. [関連リンク] で、[マルチファクター認証] をクリックします。
   [マルチファクター認証] ページが開きます。
3. [生体認証を登録 **(Register biometric authentication)]** をクリックします。

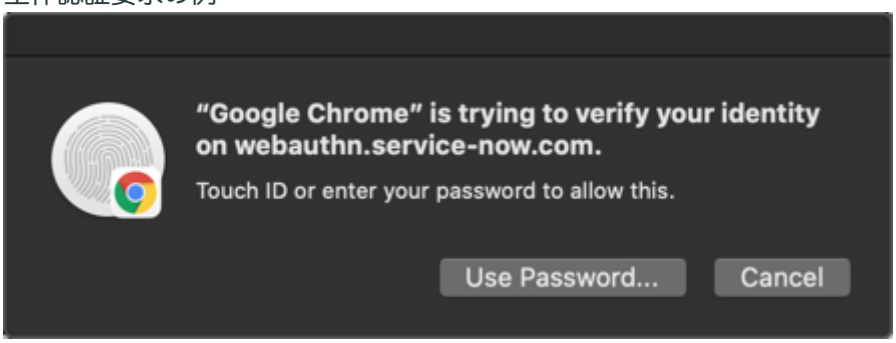


4. 認証システムのニックネームを入力し、[登録] をクリックします。



5. プロンプトが表示されたら、画面の指示に従って生体認証装置で認証します。
このメッセージは、個々の認証システムによって異なります。

生体認証要求の例



正常に認証されると、確認ウィンドウが表示されます。[X] をクリックして確認画面を閉じます。

結果
生体認証装置が登録されました。[マルチファクター認証] ページに生体認証システムのリストが表示されます。

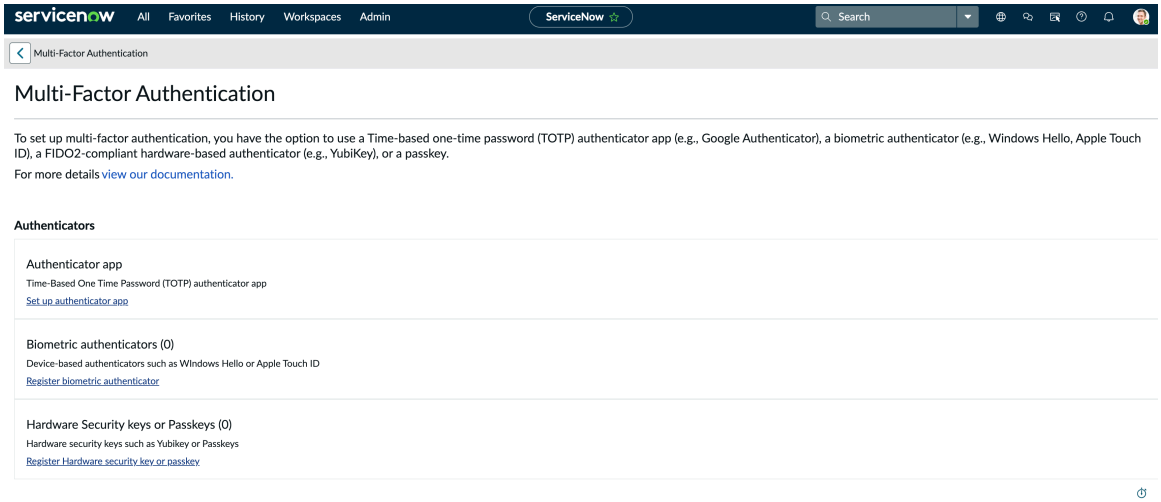
ハードウェアセキュリティキーの登録

MFA ログインの一部として使用するハードウェアキーを登録します。

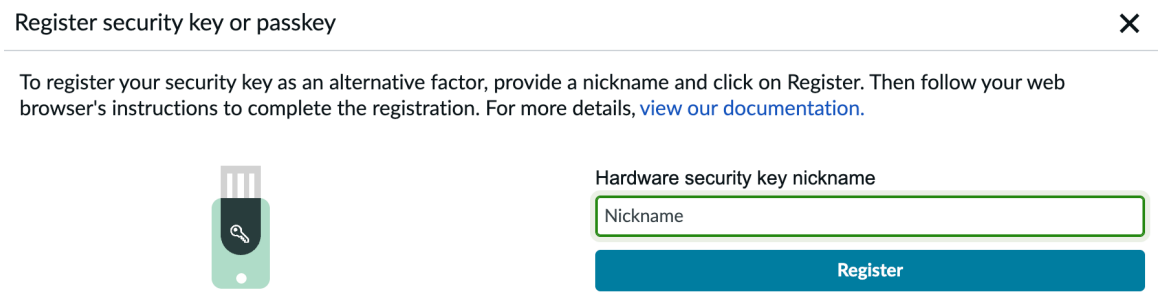
始める前に
必要なロール：なし

手順

1. 移動先 すべて > セルフサービス > プロファイル.
2. [関連リンク] で、[マルチファクター認証] をクリックします。
   [マルチファクター認証] ページが開きます。
3. [ハードウェアセキュリティキーを登録] をクリックします。



4. ハードウェアキーのニックネームを入力し、[登録] をクリックします。



5. プロンプトが表示されたら、ハードウェアセキュリティキーを挿入してアクティブ化します。
   正常に認証されると、確認ウィンドウが表示されます。[X] をクリックして確認画面を閉じます。

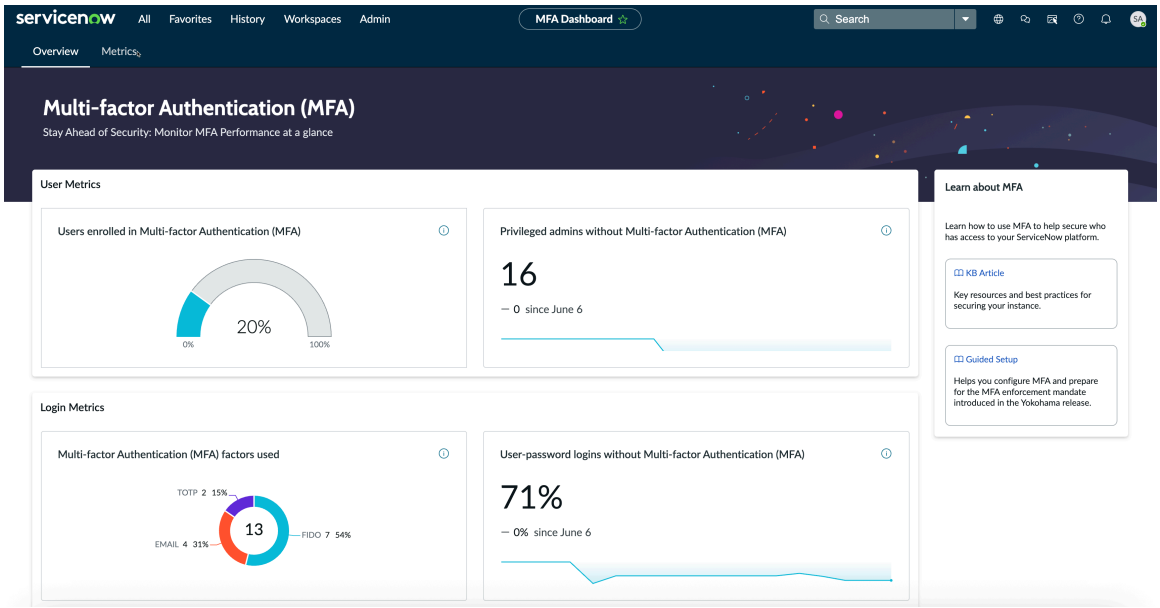
結果
ハードウェアキーが登録されました。[マルチファクター認証] ページにハードウェアキーのリストが表示されます。

MFA ダッシュボード

さまざまな MFA メトリクスを表示して、MFA の採用と使用状況を把握します。

MFA は、ログインプロセス中に追加の検証手順を要求することでセキュリティを強化します。MFA ダッシュボードは、組織の MFA 設定を監視および管理するのに役立ちます。このダッシュボードには、MFA ユーザー登録、MFA を選択していない特権アドミン、およびコンプライアンスの概要が表示されます。これを使用して、すべてのユーザーの MFA を有効にしてセキュリティを強化できます。

MFA ダッシュボードにアクセスするには、以下に移動します。 すべて > 多要素認証 > **MFA** ダッシュボード.



注: メトリクスを表示するには、MFA を有効にする必要があります。詳細については、「マルチファクター認証システムプロパティ」を参照してください。

MFA ダッシュボードに表示されるメトリクスのタイプは次のとおりです。

- ユーザーメトリクス
- ログインメトリクス

ユーザーメトリクス

次の表に、MFA ダッシュボードのユーザーメトリクスの詳細を示します。

ユーザーメトリクス

メトリクス	説明
多要素認証 (MFA) に登録されているユーザー	ユーザー名とパスワードベースのログインを実行でき、MFA に登録されているユーザーの割合。このメトリクスは、一定期間にわたるユーザーによる MFA の採用に関するインサイトを提供します。 **注:** 理想的には、スコアを徐々に増加させ、一定期間にわたって 100% にする必要があります (前日のレコードを収集するために 1 日 1 回リフレッシュされます)。
多要素認証 (MFA) のない特権アドミン	特権アドミニストレーターが MFA を使用しないことは、プラットフォームのセキュリティにとって重大なリスクです。MFA を使用してこれらのユーザーを取得することをお勧めします。

ユーザーメトリクス (続く)

メトリクス	説明
	注: 特権アドミニストレーターは、sys_icenter_role_configテーブルから少なくとも 1 つのロールを持つユーザーです。(前日のレコードを収集するために 1 日 1 回更新されます)。

ログインメトリクス

次の表は、MFA ダッシュボードのログインメトリクスの詳細を示しています。

ログインメトリクス

使用された多要素認証 (MFA) 要素	ユーザー名とパスワードベースのログイン中に使用される MFA 要素の分類。
多要素認証 (MFA) を使用しないユーザーパスワードのログイン	MFA を使用しないユーザー名とパスワードベースのログインの割合。このメトリクスは、一定期間の MFA の採用に関するインサイトを提供します。 注: 理想的には、スコアは徐々に減少し、一定期間にわたってゼロになります (前日のレコードを収集するために 1 日 1 回リフレッシュされます)。

ユーザーメトリクス

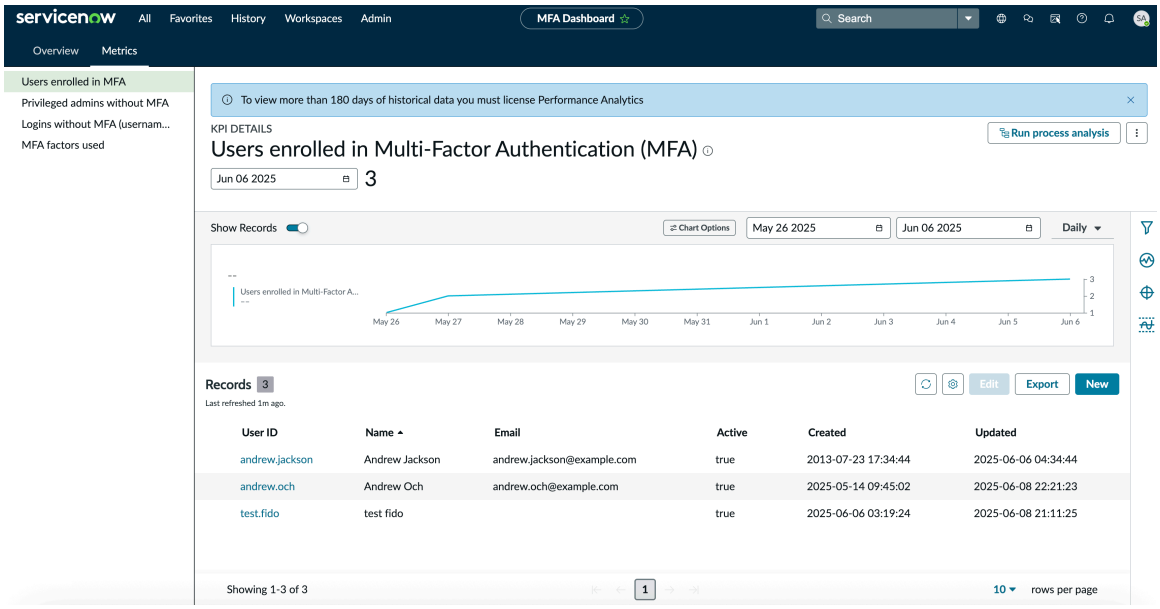
[ユーザーメトリクス] では、ユーザーの MFA 登録の傾向を ServiceNowに表示します。

MFA ダッシュボードのユーザーメトリクスは次のとおりです。

- 多要素認証 (MFA) に登録されているユーザー
- 多要素認証 (MFA) のない特権アドミン

多要素認証 **(MFA)** に登録されているユーザー

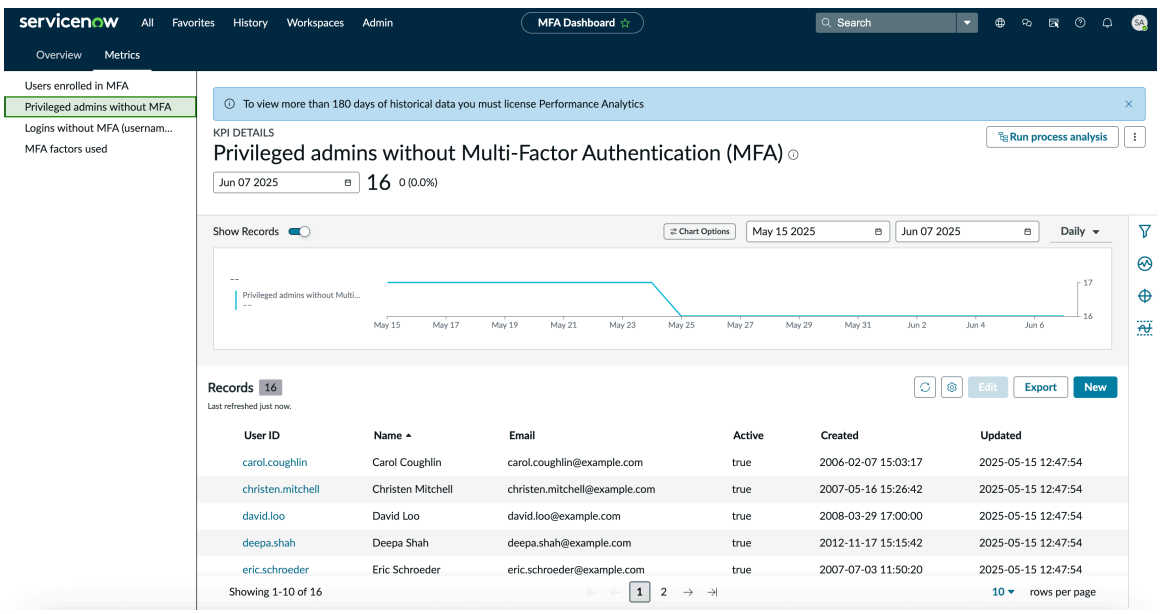
ユーザー名とパスワードベースのログインを実行でき、MFA に登録されているユーザーの割合。このメトリクスは、一定期間にわたるユーザーによる MFA の採用に関するインサイトを提供します。



注: 理想的には、スコアを徐々に増加させ、一定期間にわたって 100% にする必要があります (前日のレコードを収集するために 1 日 1 回リフレッシュされます)。

多要素認証 (MFA) のない特権アドミン

特権アドミニストレーターが MFA を使用しないことは、プラットフォームのセキュリティにとって重大なリスクです。MFA を使用してこれらのユーザーを取得することをお勧めします。



注: 特権アドミニストレーターは、 **sys_icenter_role_config** テーブルから少なくとも 1 つのロールを持つユーザーです。(前日のレコードを収集するために 1 日 1 回更新されます)。

ログインメトリクス

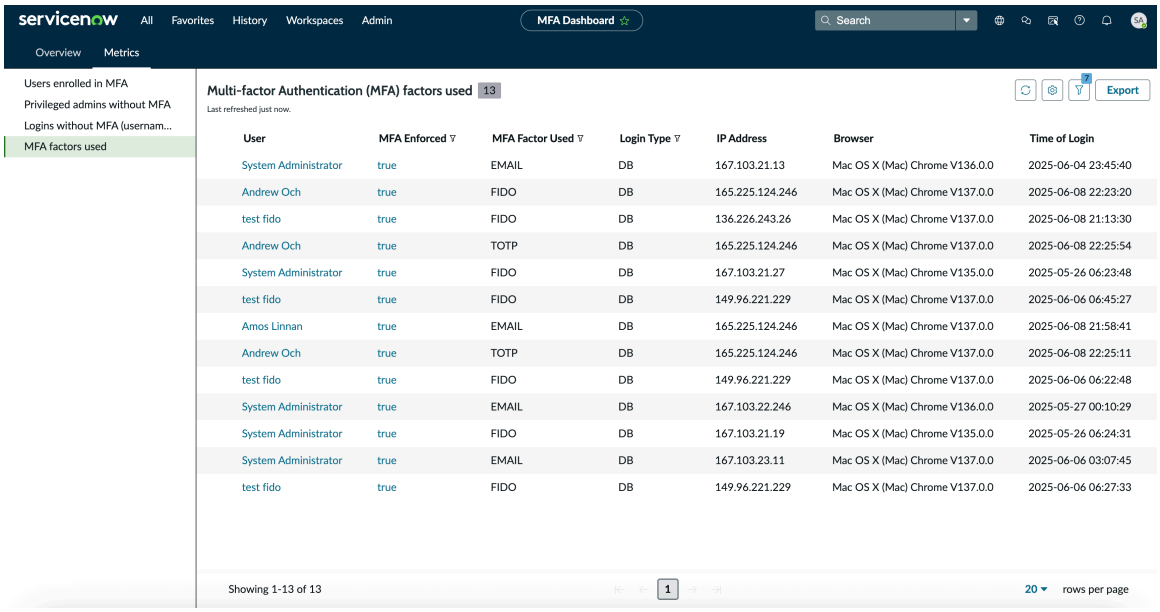
[ログインメトリクス] では、 ServiceNowにログインの傾向が表示されます。

MFA ダッシュボードのログインメトリクスは次のとおりです。

- 使用された多要素認証 (MFA) 要素
- 多要素認証 (MFA) を使用しないユーザーパスワードのログイン

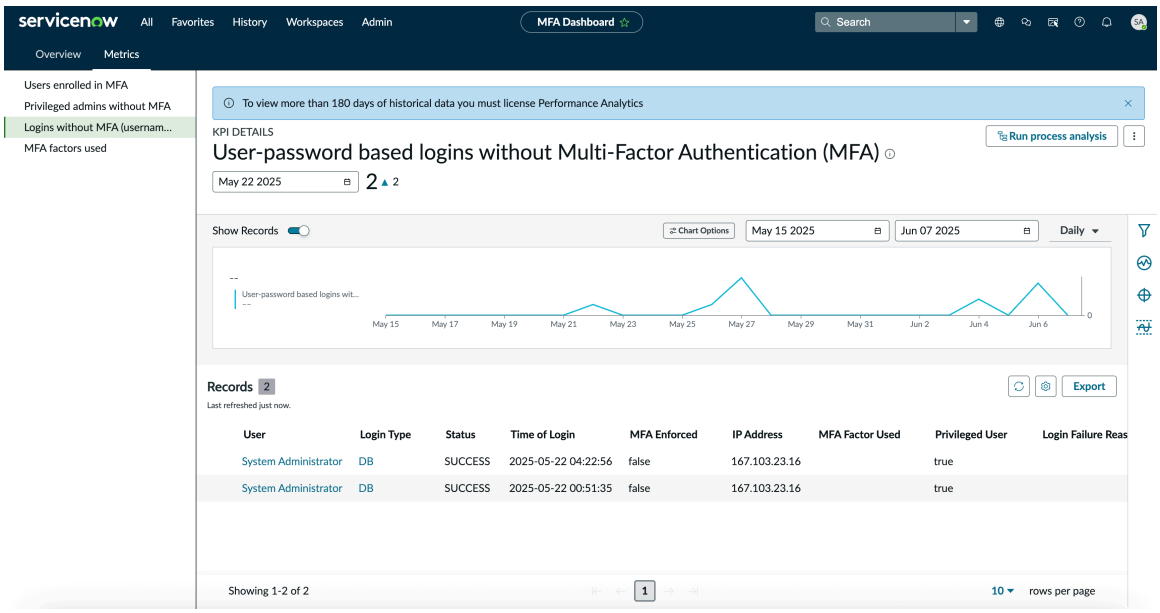
使用された多要素認証 (MFA) 要素

ユーザー名とパスワードベースのログイン中に使用される MFA 要素の分類。



多要素認証 (MFA) を使用しないユーザーパスワードのログイン

マルチファクター認証を使用しないユーザー名とパスワードベースのログインの割合。このメトリクスは、一定期間の MFA の採用に関するインサイトを提供します。



注: 理想的には、スコアは徐々に減少し、一定期間にわたってゼロになります (前日のレコードを収集するために 1 日 1 回リフレッシュされます)。

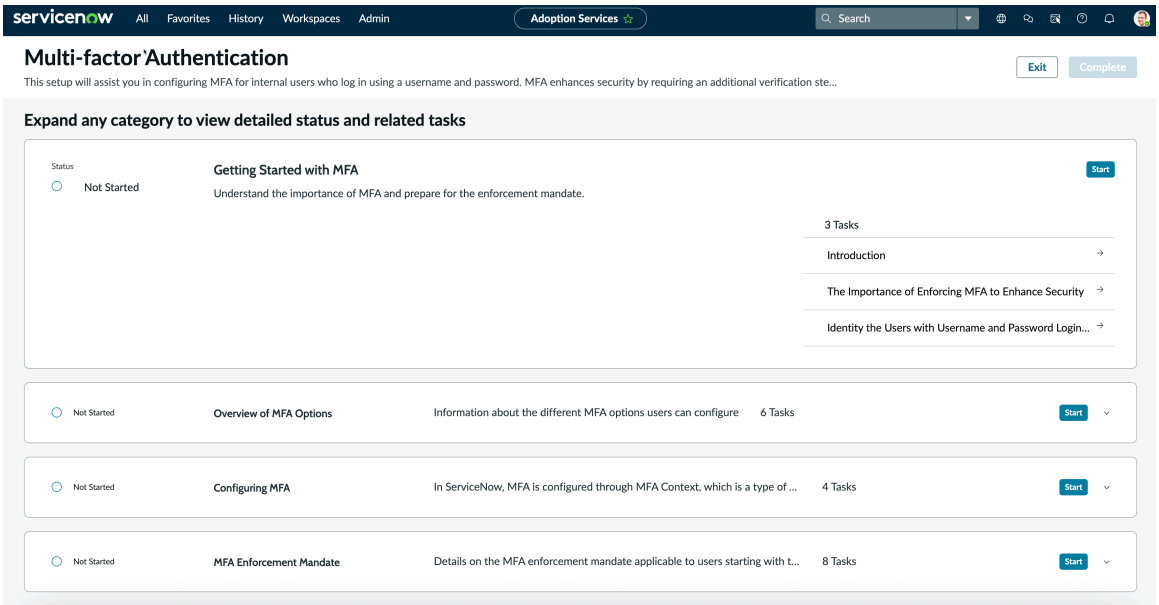
MFA ガイド付きセットアップ

MFA ガイド付きセットアップを使用して MFA モジュールの初期構成を順を追って実行し、MFA の適用の要件を理解します。

MFA ガイド付きセットアップは、ユーザー名とパスワードを使用してログインする内部ユーザーの MFA を構成するのに役立ちます。

MFA ガイド付きセットアップにアクセスするには、次の場所に移動します すべて > アダプションサービス > ガイド付きセットアップ. [すべてのアプリケーション] を選択し、[**Multi-factor Authentication**] を選択できます。

マルチファクター認証のガイド付きセットアップが新しいタブで開きます。任意のカテゴリを展開すると、表示されるページに詳細なステータスと関連タスクが表示されます。



MFA 構成を理解するには、次のタスクを完了します。

- MFA の開始
- MFA オプションの概要
- MFA の構成
- MFA 適用の義務
- ユーザーコミュニケーションとオンボーディング
- モニタリング

ガイド付きセットアップの使用方法の詳細については、「 ガイド付きセットアップ 」を参照してください。

複数プロバイダーのシングルサインオン (SSO)

外部 SSO を使用すると、組織内で複数の SSO ID プロバイダー (IdP) を使用して認証を管理し、同時にローカルデータベース (基本) 認証も保持できます。

複数プロバイダーのシングルサインオン (SSO) は、ユーザーが 1 回のログインと 1 セットの認証情報で複数のアプリケーションにアクセスできるようにするための認証方法です。

SSO を使用するには、次のことを理解する必要があります。

- *`Service Providers`*: ServiceNow インスタンスにアクセスしようとするユーザーが、検証の成功後に認証情報を検証するために ID プロバイダー (IdP) にリダイレクトされると、ユーザーはインスタンスへのアクセスを許可されます。ここでは、 ServiceNow はサービスプロバイダーとして機能し、ID プロバイダー (IdP) に依存してユーザー認証を処理し、インスタンスへのアクセスを許可します。
- ID プロバイダー:IdP は、システムにアクセスするためのユーザーの ID と認証情報を検証する外部システムです。

SSO を確立するには、 ServiceNowにアクセスするために、マルチプロバイダーシングルサインオン (SSO) をアクティブ化し、 *Integration - Multiple Provider Single Sign-On Installer (com.snc.integration.sso.multi.installer)* プラグインをインストールする必要があります。詳細については、「Multi-Provider SSO プラグインのアクティブ化」を参照してください。

プラグインが正常にインストールされたら、プラグインに付属する SSO プロパティ、アクセステーブル、およびスクリプトをカスタマイズできます。詳細については、「複数プロバイダー SSO のプロパティ、テーブル、およびスクリプト」を参照してください。

ServiceNow は、次の SSO メソッドをサポートしています。

- OpenID Connect
- SAML 2.0
- ダイジェスト認証 (トークンベースの認証)。

要件に基づいて SSO 方法を選択し、SSO を構成するための準備方法の詳細を確認してください。複数プロバイダー SSO を設定するには、プロパティの設定、ID プロバイダー (IdP) の作成、SSO を使用するユーザーの設定など、いくつかの手順を実行する必要があります。詳細については、「複数プロバイダー SSO の設定」を参照してください。

構成が正常に完了すると、インスタンス内のアクティブな IdP が ServiceNowに一覧表示されます。さまざまな SAML または OIDC ID プロバイダー (IdP) を一覧表示できます。

> **注:** ログインページには最大 10 個の IdP をリストできます。インスタンスに Domain Support - Domain Extensions Installer (*com.glide.domain.msp_extensions.installer*) プラグインがインストールされて有効になっている場合、IdP オプションは表示されません。

ServiceNow のチューリッヒリリースには、SSO に関する次の機能拡張が含まれています。

- ログインページで **SAML IdP** をリストする:プラットフォームとポータルの両方のログインエクスペリエンスに一覧表示されている SAML および OIDC IdP を使用してログインし、ユーザーが希望する IdP を簡単に選択できるようにします。以前は、OIDC IdP のみがリストされていました。
- 自動プロビジョニングのグループを選択:SAML と OIDC の自動プロビジョニング構成中に特定のグループを選択し、ユーザーが正しいグループに自動的に割り当てられるようにします。
- 同じ既知の **URL** を使用して複数の **OIDC** レコードを設定する:同じ既知の URL を使用して OIDC レコードを作成できるようにすることで OIDC セットアップを簡素化し、構成プロセスを簡素化します。
- 拡張 外部ログアウト完了ページ:ログイン失敗の理由をユーザーに表示する。ログアウトが成功した場合に、再度ログインして外部ログアウト完了ページで ServiceNow できるようにプロビジョニングします。
- 拡張エラーメッセージ:シングルログアウト (SLO) が失敗した場合の一般的なエラーメッセージを表示し、一貫性のある安全な通信を確保します。
- **SAML** 証明書と暗号化キーストアの通知の機能拡張:SAML 証明書と暗号化キーストアの更新の有効期限に関する通知をアドミニストレーターにタイムリーに受信し、SSO 構成を安全で最新の状態に保ちます。

Example: 組織にSSOが必要な理由

世界中に分散している企業では、従業員に対して 1 つの SSO プロバイダー、ベンダーに対して別のプロバイダー、アドミニストレーターに対してローカルデータベース認証が必要になる場合があります。また、同じインスタンスに SAML 2.0 とダイジェストトークン認証ソリューションを実装するような会社もあります。

Multi-Provider SSO プラグインのアクティブ化

この統合には、 Integration - Multiple Provider Single Sign-On Installer プラグイン (com.snc.integration.sso.msoi.installer) が必要です。

始める前に
必要なロール：admin。

com.snc.integration.sso.multi.installer プラグインは、OIDC、SAML、および Digest にも使用できます。

必要なロール：admin

手順

1. 移動先 すべて > システムアプリケーション > 利用可能なすべてのアプリケーション > すべて.
2. フィルター基準と検索バーを使用して、Integration - Multiple Provider Single Sign-On Installer プラグイン (com.snc.integration.sso.msoi.installer) を見つけます。

   名前または ID でプラグインを検索できます。プラグインが見つからない場合は、ServiceNow 担当者から要求する必要があります。
3. [インストール] を選択して、インストールプロセスを開始します。

   注: ドメインセパレーションと代理アドミンがインスタンスで有効になっている場合、管理ユーザーはグローバルドメインに含まれている必要があります。それ以外の場合、次のエラーが表示されます： 「別の操作が実行されているため、アプリケーションのインストールは利用できません： <プラグイン名> のプラグインの有効化 (Application installation is unavailable because another operation is running: Plugin Activation for <plugin name>)」

   インストールが完了するとメッセージが表示されます。プラグインとともにインストールされるコンポーネントの詳細については、「 アプリケーションとともにインストールされているコンポーネントの検索 」を参照してください。

複数プロバイダー SSO のプロパティ、テーブル、およびスクリプト

Integration - Multiple Provider Single Sign-On Installer プラグインには、次のようなシステムプロパティ、テーブル、およびスクリプトが含まれています。

プロパティ

複数プロバイダー SSO は、次のシステムプロパティを追加します。

複数プロバイダー SSO のプロパティ

名前	説明
glide.authenticate.multisso.debug	複数プロバイダー SSO 統合のデバッグログを有効 (true) または無効 (false) にします。 • タイプ：true | false • デフォルト値：false
glide.authenticate.multisso.enabled	複数プロバイダー SSO を有効 (true) または無効 (false) にします。

複数プロバイダー **SSO** のプロパティ (続く)

名前	説明
	• タイプ：true | false • デフォルト値：false 注: インスタンスでアカウント復旧 (ACR) が有効になっている場合、このプロパティを false に設定しても、マルチプロバイダー SSO は無効になりません。ユーザー名とパスワードでログインするには、 *glide.sso.acr.enabled* プロパティを使用して ACR も無効にする必要があります。このプロパティの詳細については、「アカウント復旧プロパティ」を参照してください。
glide.authenticate.multissov2_feature.enabled	このプロパティは、MultiSSOv2 バージョンがインスタンスで有効になっているかどうかを決定します。
glide.authenticate.show.max.sso.login.option	このプロパティは、ログイン画面に表示される SSO オプションの最大数を決定します。 注: デフォルト値は 5 です。プロパティの最大値は 10 です。
glide.authenticate.show.max.sso.login.option	このプロパティは、ログイン画面に表示される IdP の最大数を決定します。"glide.authenticate.show.max.sso.login.option" 注: デフォルト値は 10 です。

テーブル

複数プロバイダー SSO は、以下のテーブルを追加します。

複数プロバイダー **SSO** のテーブル

名前	説明
SSO プロパティ [sso_properties]	各 IdP、スキーマ、共通 SSO データなどを保存します。
SAML 2 Update 1 プロパティ [saml2_update1_properties]	SAML 2.0 Update 1 の構成データ (SAML 証明書など) を保存します。
ダイジェストプロパティ [digest_properties]	ダイジェストトークン認証の構成データを保存します。
SSO フェデレーション [sso_fedation]	各 SSO フェデレーションのデータを保存します。
OIDC ID プロバイダー [oidc_identity_provider]	Open ID Connect ベースの ID プロバイダーのデータを保存します。

スクリプト

複数プロバイダー SSO は、以下のスクリプトを追加します。

複数プロバイダー **SSO** スクリプト

名前	説明
MultiSSO	顧客が会社ごとに SSO タイプを定義できるようにします。
MultiSSOLogin	各ドメインが独自のログインスクリプトを保持できるようにします。
MultiSSOLogout	各ドメインが独自のログアウトスクリプトを保持できるようにします。
MultiSSO_OIDC_custom	ユーザーが OIDC 接続のカスタムシングルサインオンスクリプトを定義できるようにします。
MultiSSO_OIDC_logout_custom	ユーザーが OIDC 接続のカスタムログアウトスクリプトを定義できるようにします。
MultiSSO_Abstract_Core	すべての複数プロバイダー SSO クラスの基底クラスを提供します。
MultiSSO_ClientHelper	複数プロバイダー SSO 用のクライアント呼び出し可能ユーティリティ関数を提供します。
MultiSSO_DigestedToken	ダイジェストトークン認証の基本システムロジックを提供します。
MultiSSO_SAML2_Update1	マルチテナンシングルサインオン用の SAML 2.0 Update 1 認証を処理するロジックを提供します。

複数プロバイダー **SSO** の設定

複数プロバイダー SSO を設定するには、プロパティの設定、ID プロバイダー (IdP) の作成、SSO を使用するユーザーの設定など、いくつかの手順を実行する必要があります。

各構成の詳細については、次のトピックを参照してください。

- 複数プロバイダー SSO (SAML) IdP 認証フロー
- マルチプロバイダー SSO プロパティの設定
- 外部 ID プロバイダーの作成
- 複数プロバイダー SSO のユーザーの構成

マルチプロバイダー **SSO** プロパティの設定

SSO プロパティを設定し、さらにシステムのプロパティテーブルにプロパティを追加して IdP 包含リストを設定します。

始める前に
必要なロール：admin

手順

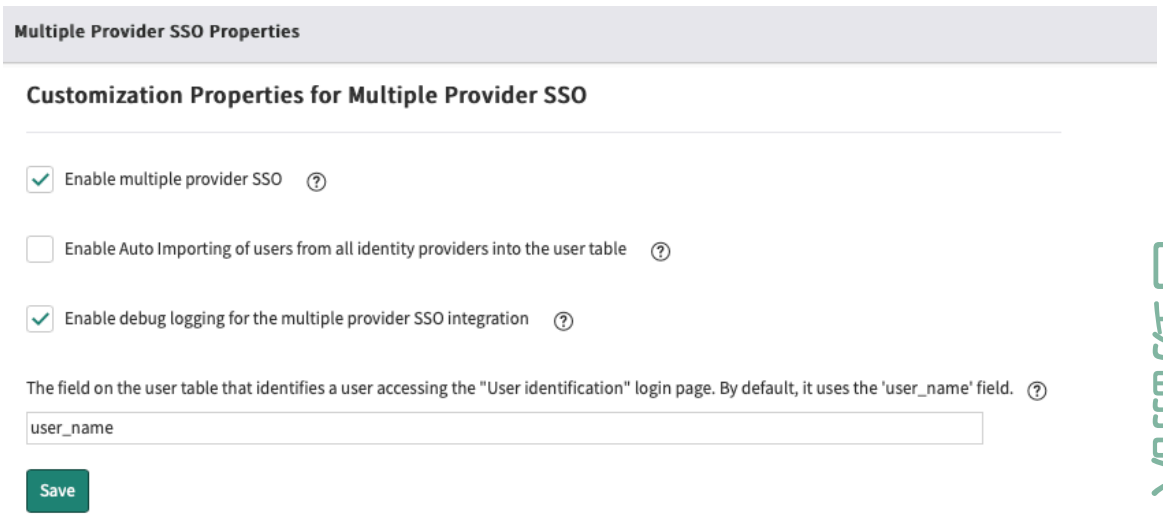
1. 移動先 すべて > マルチプロバイダー **SSO** > アドミニストレーション > プロパティ.
2. [マルチプロバイダー **SSO** の有効化 **(Enable Multi-Provider SSO)]** チェックボックスをオンにします。

この選択により、[外部ログインを使用] リンクがログインページに追加されます。

3. IdP のユーザーでユーザーテーブルを更新するには、[自動インポートの有効化 **(Enable Auto Importing)**] オプションを選択します。
4. コンテンツフレームの下部にデバッグメッセージを表示できるようにするには、[マルチプロバイダー **SSO** 統合のデバッグログの有効化 **(Enable debug logging for the Multi-Provider SSO integration)**] チェックボックスをオンにします。
有効にすると、デバッグログ機能によってパフォーマンスが低下し、ログ生成のためにディスクスペースが使用されます。
5. [ユーザー **ID** ログインページへのアクセス時にユーザーを識別するユーザーテーブル上のフィールド **(The field on the user table that identifies a user accessing the User identification login page)**] プロパティで、IdP がユーザーを識別するために使用する値を含むユーザーテーブルのフィールドを入力します。デフォルト値は **user_name** です。

   デフォルト値は **user_name** です。

   マルチプロバイダー **SSO** のプロパティ



6. [保存] をクリックします。
7. ユーザーがインスタンスにログインするときには、[外部ログインを使用] リンクをクリックするように指示してください。

関連トピック

SAML ユーザープロビジョニング

外部 **ID** プロバイダーの作成

複数プロバイダー SSO プロパティを設定した後、SAML 2.0 またはダイジェストトークン ID プロバイダーを更新または作成できます。

始める前に

必要なロール：admin

手順

1. 移動先 すべて > マルチプロバイダー **SSO** > **ID** プロバイダー.
2. ID プロバイダー レコードを編集するには、レコードをクリックします。

- ダイジェストトークン構成の場合は、プロパティを手動で更新します。
- SAML2 Update 1 構成の場合は、**[ID** プロバイダーのメタデータをインポート] 関連リンクを使用して ID プロバイダーのメタデータを自動的に更新するか、プロパティを手動で更新します。
- OpenID Connect 構成の場合は、プロパティを手動で更新します。

3. 新しい ID プロバイダーを作成するには、[新規] をクリックします。
   - ダイジェストトークン構成の場合：**[SSO** をダイジェスト **(Digest SSO)]** をクリックし、複数プロバイダーシングルサインオンのダイジェストプロパティを入力します。
   - SAML2 構成の場合：**[MultiSSOV2_SAML2_custom]** をクリックし、URL から ID プロバイダーのメタデータを XML としてインポートするか、ID プロバイダー情報を手動で入力します。

   - OpenID Connect の場合：**[OpenID Connect]** をクリックし、クライアント ID、クライアントシークレット、および既知の構成 URL を入力します。

4. IdP をフェイルオーバー IdP (デフォルトの IdP が利用できない場合に使用される) にするには、[デフォルト] チェックボックスをオンにします。

   SAML 2 Update 1 をアクティブにして Fuji リリースにアップグレードすると、SAML 2 Update 1 IdP がデフォルトのフェイルオーバーとして選択されます。新しいインスタンスの場合、または SAML 2 Update 1 がアクティブでないリリースからアップグレードする場合は、デフォルトのフェイルオーバー IdP は選択されていません。

   注: メタデータのインポートプロセスでは、ID プロバイダーの証明書レコードが自動的に作成されます。**x509** 証明書モジュールに移動して、証明書を表示します。

   注: Single Sign-on の証明書は、SAML 証明書を扱うため、常に PEM 形式になっている必要があります。

5. 電子署名がアクティブな場合は、[ID プロバイダー] フォームを設定し、**[eSignature** 認証用のアサーションコンシューマー **URL]** フィールドを追加します。
   ほとんどの場合、この URL は https://YOURINSTANCE.service-now.com/consumer.do です。ただし、電子署名の SAML 認証を処理する方法をカスタマイズして使用する場合は、独自のコンシューマー URL を設定できます。SAML 2.0 Update 1 のみを使用しており、複数プロバイダーシングルサインオンを使用していない場合は、電子署名の SAML プロパティを使用してアサーションコンシューマー URL を設定します。

SAML のインスタンスサービスプロバイダー (SP) メタデータの生成

SSO 構成の一部として、インスタンス SP メタデータを生成して IdP に提供できます。

始める前に
必要なロール：admin

このタスクについて
IdP には、要求の認証と転送を行うためにインスタンスの SP メタデータが必要です。

手順

1. インストールされている SSO プラグインを選択します。

オプション	説明
マルチプロバイダー SSO	移動先 マルチプロバイダー **SSO** > **ID** プロバイダー. IdP を選択し、[メタデータを生成] ボタンをクリックします。統合により、システムプロパティ設定からインスタンスの SP メタデータが自動的に生成されます。
SAML 2 SSO	移動先 **SAML 2** シングルサインオン > メタデータ. 統合により、システムプロパティ設定からインスタンスの SP メタデータが自動的に生成されます。

2. テキストボックスに SP メタデータをコピーします。

例：

```
<EntityDescriptor xmlns="urn:oasis:names:tc:SAML:2.0:metadata"
 entityID="https://yourinstance.service-now.com">
  <SPSSODescriptor AuthnRequestsSigned="false" WantAssertionsSigned="true"
 protocolSupportEnumeration="urn:oasis:names:tc:SAML:2.0:protocol">
  <SingleLogoutService Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect"
 Location="https://yourinstance.service-now.com/navpage.do" />

 <NameIDFormat>urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress</NameIDFor
mat>
  <AssertionConsumerService isDefault="true" index="0"
 Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
 Location="https://yourinstance.service-now.com/navpage.do" />
  <AssertionConsumerService index="1"
 Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
 Location="https://yourinstance.service-now.com/consumer.do"/>
 </SPSSODescriptor>
</EntityDescriptor>
```

3. インスタンス SP メタデータを IdP に提供します。
たとえば、SSOCircle を使用すると、ユーザーは SP メタデータをオンラインで提供できます。

複数プロバイダー SSO のユーザーの構成

アドミニストレーターは、会社に所属する個別のユーザーまたはユーザー全員に対して複数プロバイダー SSO を構成できます。グループに対して複数プロバイダー SSO を構成することはできません。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > マルチプロバイダー **SSO** > **ID** プロバイダー.
2. ID プロバイダーのレコードを右クリックし、**[sys_id** のコピー] を選択します。
3. データをクリップボードにコピーします。
4. ユーザーレコードまたは会社レコードに移動します。
5. フォームを設定し、**[SSO** ソース] フィールドを追加します。
6. **[SSO** ソース] フィールドに、次のいずれかを入力します。
   - **SAML** ユーザー：「**sso:**」に続けて ID プロバイダーのレコードの sys_id を入力します。
   - **SSO** フェデレーション ユーザー：「**federation:**」に続けてフェデレーションレコードの sys_id を入力します。
7. [更新] をクリックします。

IdP 接続をテスト

IdP への接続をテストすると、外部認証を有効にする前に設定が検証されます。

始める前に
必要なロール：admin

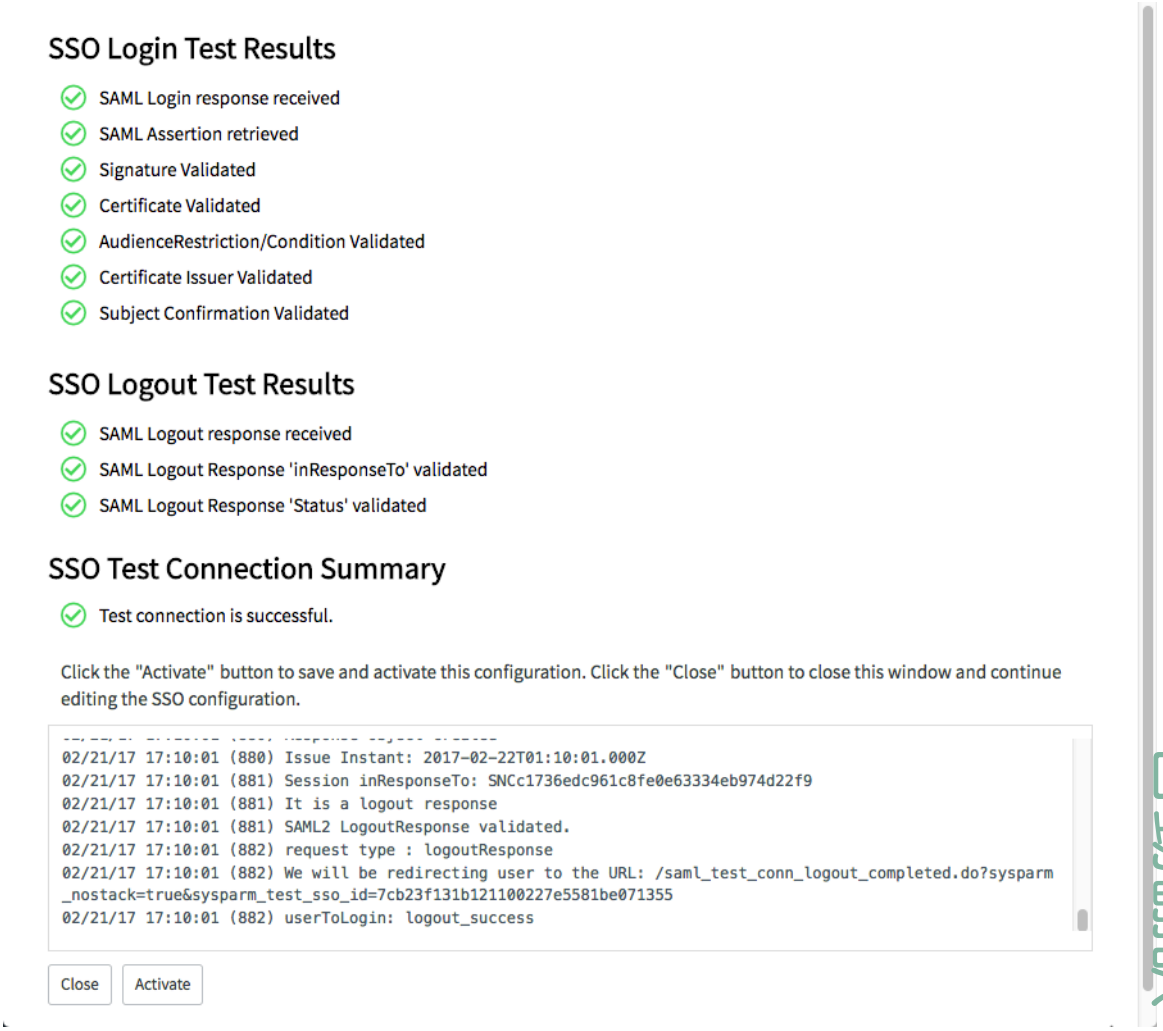
このタスクについて
Jakarta リリースでは、ポップアップウィンドウ内のテスト接続をサポートしています。このオプションで IdP が正しく機能しない場合は、このデフォルト設定をオフにできます。
MultiSSO IDP メタデータのリフレッシュジョブは、接続の作成、更新、またはテスト中に IdP の証明書をフェッチして更新します。

> **注:** 一部の idp ケースでは、テスト接続に失敗した場合、値が false の glide.authenticate.multisso.test.connection.mandatory を作成する必要があります。作成すると、テスト接続なしで idp をアクティブ化することができます。

手順

1. 移動先 すべて > マルチプロバイダー **SSO** > **ID** プロバイダー.
2. 定義済みの IdP を選択するか、[新規] をクリックして新しい IdP を定義します。
3. オプション: 新しい IdP を設定する場合は、ID プロバイダーを構成します。
4. [テスト接続] をクリックし、IdP のログイン認証情報を入力してログインを確認します。
   テスト接続が成功するまで、IdP をアクティブ化することはできません。テストに失敗した場合は、構成情報を更新して保存できますが、この構成をアクティブ化することはできません。
5. **[SSO** テスト結果/サマリー **(SSO Test Results/Summary)]** または **[SSO** ログのテスト中] セクションを使用してログメッセージを表示して、結果を確認します。
   エラーが発生した場合は、「マルチ SSO (SAML 2.0) のエラーと修正」を参照してください。

6. テストが完了したら [キャンセル] をクリックします。



一般的な IdP 接続エラー

次の表では、一般的な IdP 接続エラーとその解決策について説明します。

IdP テスト接続のトラブルシューティング

エラーメッセージ	解決策
ユーザーフィールドの検証に失敗しました。無効なユーザーフィールド「<フィールド名>」は sys_user テーブルのフィールドではありません。(Invalid User Field '<field name>' is not a field on sys_user table.)	選択したユーザーテーブルフィールドの内容が SAML NameID トークンと一致していることを確認します。
アサーションの発行者が無効です。	**[ID プロバイダー URL]** に IdP への有効な URL が含まれていることを確認します。各 IdP URL は一意である必要があります。
AudienceRestriction の検証に失敗しました。	[対象者 **URI]** にインスタンスへの有効な URL が含まれていることを確認します。
IdP のセッションからログアウトできません。	**SingleLogoutRequest** URL に IdP のログアウトサービスへの有効な

IdP テスト接続のトラブルシューティング (続く)

エラーメッセージ	解決策
	URL が含まれていることを確認します。
署名が認証情報のキーに対して検証されませんでした。 (Signature did not validate against the credential's key.)	IdP に有効な証明書がインストールされていることを確認します。

SAML によるスクリプトの問題のトラブルシューティング

SAML によるスクリプトの問題をトラブルシューティングします。複数の Single Sign-On をアクティブ化するときに SAML が既にアクティブであり、インストールを既にカスタマイズしている場合は、スクリプトの問題が発生する可能性があります。

始める前に

必要なロール：admin

手順

1. 変更されたインストール終了 SAML2SingleSignon_update1 とスクリプトインクルード SAML2_update1 をバックアップします。
2. インストレーションイグジットとスクリプトインクルードの両方を、ベースラインシステムで使用可能なバージョンに戻します。
3. **Integration - Multiple Provider Single Sign-On Installer** プラグインをアクティブ化するか、またはアップグレードします。
   SAML と必要なすべてのファイルが SAML 2 Update 1 にアップグレードされます。
4. [複数の SSO (Multiple SSO)] プロパティページを開き、[マルチプロバイダー **SSO** の有効化 **(Enable Multi-Provider SSO)]** チェックボックスをオンにして有効にします。
5. SAML2SingleSignon_update1 インストレーションイグジットの変更をベースラインスクリプトインクルード **MultiSSO_SAML2_Update1** に入力し、SAML2_update1 スクリプトインクルードの変更をベースライン SAML2_update1 スクリプトインクルードに入力します。

複数プロバイダー **SSO** を使用したログイン

ユーザーが複数プロバイダー SSO を使用してログインするために推奨される最も効率的な方法は、特別に設定された URL を使用する方法です。

始める前に

必要なロール：admin

このタスクについて

複数プロバイダー SSO の設定後は、パラメーター文字列で正しい IdP を使用して URL をユーザーに送信できます。例：

```
/login_with_sso.do?glide_sso_id=<sys_id of the sso configuration>
```

ユーザーが IdP ページに正常にログインすると、IdP sys_id を含む cookie がブラウザーに追加されます。次にユーザーがログインしようとすると、ユーザーは IdP サーバーへのログインにリダイレクトされ、インスタンスへの自動ログインが行われます。

URL パラメーターが設定されていないか、ブラウザーのキャッシュがクリアされている場合は、次のように操作することもできます。

手順

1. ログインページの [外部ログインを使用] リンクをクリックします。

   外部ログインページが表示されます。[ローカルログインを使用] をクリックすると、標準ログインページに戻ることができます。

2. 複数プロバイダー SSO プロパティで設定したユーザーテーブルの指定されたフィールドの値を入力します。

   ユーザーは IdP サーバーにリダイレクトされ、そこでログインします。

結果
ユーザーが IdP に正常にログインすると、インスタンスにアクセスしようとするたびにその IdP に自動的にリダイレクトされます。ユーザーが別の IdP にアクセスできるようにするには、 パラメーターに新しい IdP 情報を含む URL を送信します。ユーザーが正常にログインすると、新しい IdP が cookie 内の古い IdP を上書きします。ユーザーが正常にログインしない場合、古い IdP 情報は cookie に保持されます。

ユーザーによるログイン用 **ID** プロバイダーの選択の有効化

SSO フェデレーションのサポートにより、ユーザーはログインする IdP を選択できるようになります。

始める前に
必要なロール：admin

このタスクについて
SSO フェデレーションは、インスタンスを含む複数の IdP およびサービスプロバイダーからのメタデータをアグリゲートします。フェデレーションは、IdP 名や IdP 証明書などの情報を含むメタデータを XML ファイルとして公開します。その後、アドミニストレーターは XML ファイルを読み込むようにインスタンスに指示し、必要なすべての IdP 情報を SSO プロパティテーブルに自動的に入力することができます。

手順

1. 移動先 すべて > マルチプロバイダー **SSO** > フェデレーション.
2. [**New**] をクリックします。
3. 必要に応じて、フィールドに値を入力します (表を参照)。
4. [送信] をクリックします。
5. フェデレーションの設定後、[SSO メタデータのリフレッシュ (Refresh SSO Metadata)] スケジュール済みジョブを有効にしてから、フェデレーション IdP にアクセスするユーザーを設定します。作成したフェデレーションレコードの sys_ID を使用します。

   作成したフェデレーションレコードの sys_ID を使用します。

インスタンスが SSO プロパティテーブルに IdP 情報を入力します。フェデレーションを使用するように設定されたユーザーがログインすると、設定したディスカバリーサービス URL にリダイレクトされます。次に、IdP を選択して必要な認証情報を入力します。または、パラメーターで IdP を使用して URL をユーザーに送信します。

ユーザーにログイン用 **ID** プロバイダーの選択を許可

フィールド	説明
名前	フェデレーションの説明的な名前を入力します。
有効	インスタンスがフェデレーションから XML ファイルをプルできるようにするには、このチェックボックスをオンにします。
タイプ	このフェデレーションがサポートする認証のタイプを選択します。
ディスカバリーサービス URL	このフェデレーションのディスカバリーサービスの URL を入力します。これは、ユーザーが IdP を選択してログインするように指示されるサイトです。
メタデータ URL	フェデレーションメタデータを保持する XML ファイルの URL を入力します。
x509 証明書	フェデレーション証明書を選択します。
ドメイン	データが属するドメインを選択します。

注: InCommon フェデレーション ID 管理 IdP が事前設定されています。

複数プロバイダー **SSO** で サービスポータル を使用して **URL** をリダイレクトする

サービスポータル システムのプロパティとスクリプトインクルードを組み合わせて使用し、ポータルにログインするユーザーに対してシステムが URL リダイレクトを処理する方法を決定します。

始める前に
必要なロール：admin

このタスクについて
システムプロパティを使用してプライマリ ID プロバイダー (IdP) に自動的にリダイレクトする場合、サービスポータル によってその IdP に自動的にリダイレクトされます。IdP が複数ある場合、サービスポータル では、ログインページで [外部ログインを使用] へのリンクが表示されます 。

手順

1. Service Portal ログインページの設定
2. ログイン後にサービスポータルへリダイレクトする

アカウント復旧 **(ACR)**

アドミニストレーターは、アカウント復旧 (ACR) を設定して、SSO の設定ミスや期限切れの証明書への対処などの復旧アクティビティを実行できます。

注: ACR を有効にすると、インスタンスに対して SSO が有効になっている場合に、ローカルのインタラクティブログイン (ユーザー名またはパスワードベース) が無効になります。

ACR は次の機能を提供します。

- シングルサインオン (SSO) ログインをバイパスして、アドミニストレーターとして SSO 構成の問題に対処します。
- SSO を使用してログインし、アカウント復旧として設定されたアドミニストレーターアカウントでタスクを実行します。
- ACR フローにより、アドミニストレーターは、SSO の構成ミスや期限切れの証明書など復旧を必要とする場合に、セルフサービス機能を使用して、アカウントの復旧に対処できるようになります。
- インスタンスへの不正アクセスを減らし、SSO ユースケース外で ACR を使用するための強力な基盤を提供します。

新しいインスタンス

新しいインスタンスで ACR を使用するには、次の操作を行う必要があります。

- Mutli-SSO プラグイン (com.snc.integration.sso.multi.installer) をアクティブにします。
- ACR を有効にします (glide.sso.acr.enabled)。新しいインスタンスの場合、これはデフォルトで有効になっています。
- SSO プロパティ (glide.authenticate.multisso.enabled) を有効にする前に、アドミニストレーターは ACR ユーザーとして登録する必要があります。

  注: インスタンスでアカウント復旧 (ACR) が有効になっている場合、このプロパティを false に設定しても、マルチプロバイダー SSO は無効になりません。ユーザー名とパスワードでログインするには、 *glide.sso.acr.enabled* プロパティを使用して ACR も無効にする必要があります。このプロパティの詳細については、「アカウント復旧プロパティ」を参照してください。

- アドミニストレーターは、ACR ユーザーとして登録する前に、ローカルログインのパスワードを設定し、MFA を登録する必要があります。

アップグレードされたインスタンス

アップグレードされたインスタンスで ACR を使用するには、次の操作を行う必要があります。

- Mutli-SSO プラグイン (com.snc.integration.sso.multi.installer) をアクティブにします。
- ACR を有効にします (glide.sso.acr.enabled)。

  注: アップグレードされたインスタンスの場合、アドミニストレーターは ACR を有効にする必要があります。

- SSO プロパティ (glide.authenticate.multisso.enabled) を有効にする前に、アドミニストレーターは ACR ユーザーとして登録する必要があります。
- アドミニストレーターは、ACR ユーザーとして登録する前に、ローカルログインのパスワードを設定し、MFA を登録する必要があります。

アカウント復旧ユーザーの設定

アカウント復旧を使用するには、少なくとも 1 つのアドミンアカウントをアカウント復旧ユーザーとして登録する必要があります。少なくとも 1 つのアカウントが設定されるまで、インスタンスでシングルサインオンを有効にすることはできません。このプロセスの詳細については、「[アカウント復旧プロパティ] ページからのアカウント復旧ユーザーの設定」を参照してください。

自動翻訳

注: すでにシングルサインオンを使用しているインスタンスを Rome 以降のリリースにアップグレードする場合、シングルサインオンは復旧ユーザーが設定されていなくても引き続き機能します。

アカウント復旧の構成

このアカウント復旧機能は、**Integration - Multiple Provider Single Sign-On Installer (com.snc.integration.sso.msoi.installer)** プラグインに含まれています。この機能はデフォルトで有効になっています。このアカウントおよび他のアカウントの復旧設定は、システムプロパティを使用して変更できます。これらのプロパティの詳細については、「アカウント復旧プロパティ」を参照してください。

アカウント復旧ポリシーのコンテキスト

アカウント復旧ユーザーを登録し、シングルサインオン (SSO) を有効にすると、インスタンスはすべてのローカルログインを制限します。この制限は、**SSO - ACR** コンテキスト認証ポリシーコンテキストで定義されています。コンテキストの詳細については、「アカウント復旧のコンテキスト」を参照してください。

認証ポリシーとポリシーコンテキストの詳細、およびインスタンスでの動作の詳細については、「適応認証」を参照してください。

アカウント復旧ユーザーの設定

インスタンスでアカウント復旧アクティビティを実行するアカウント復旧ユーザーを設定します。

アカウント復旧ユーザーは、SSO の設定ミスへの対処や期限切れの証明書への対処などのアカウント復旧タスクを実行するために、アドミニストレーターが使用できるユーザーアカウントです。

注: インスタンスでアカウント復旧を使用する場合は、アカウント復旧ユーザーを設定する必要があります。この手順は、インスタンスで複数プロバイダーシングルサインオンを有効にする前に必要です。

[アカウント復旧プロパティ] ページからのアカウント復旧ユーザーの設定

[アカウント復旧プロパティ] ページからアカウントの復旧を設定します。

始める前に
必要なロール：admin

新しいインスタンスで ACR を設定するには、次の操作を行う必要があります。

- Mutli-SSO プラグイン (com.snc.integration.sso.multi.installer) をアクティブにします。
- ACR を有効にします (glide.sso.acr.enabled)。新しいインスタンスの場合、これはデフォルトで有効になっています。
- SSO プロパティ (glide.authenticate.multisso.enabled) を有効にする前に、アドミニストレーターは ACR ユーザーとして登録する必要があります。
- アドミニストレーターは、ACR ユーザーとして登録する前に、ローカルログインのパスワードを設定し、MFA を登録する必要があります。

アップグレードされたインスタンスで ACR を使用するには、次の操作を行う必要があります。

- Mutli-SSO プラグイン (com.snc.integration.sso.multi.installer) をアクティブにします。
- ACR を有効にします (glide.sso.acr.enabled)。

注: アップグレードされたインスタンスの場合、アドミニストレーターは ACR を有効にする必要があります。

- SSO プロパティ (glide.authenticate.multisso.enabled) を有効にする前に、アドミニストレーターは ACR ユーザーとして登録する必要があります。

  注: インスタンスでアカウント復旧 (ACR) が有効になっている場合、このプロパティを false に設定しても、マルチプロバイダー SSO は無効になりません。ユーザー名とパスワードでログインするには、 *glide.sso.acr.enabled* プロパティを使用して ACR も無効にする必要があります。このプロパティの詳細については、「アカウント復旧プロパティ」を参照してください。

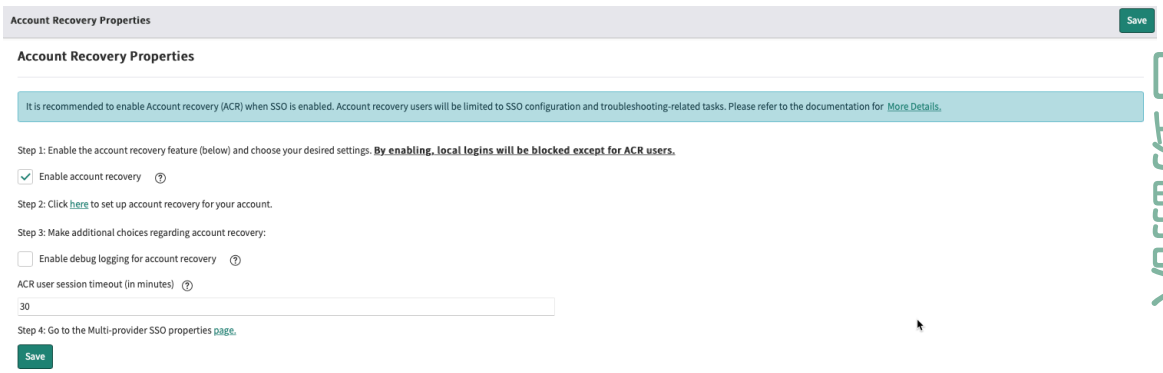
- アドミニストレーターは、ACR ユーザーとして登録する前に、ローカルログインのパスワードを設定し、MFA を登録する必要があります。

手順

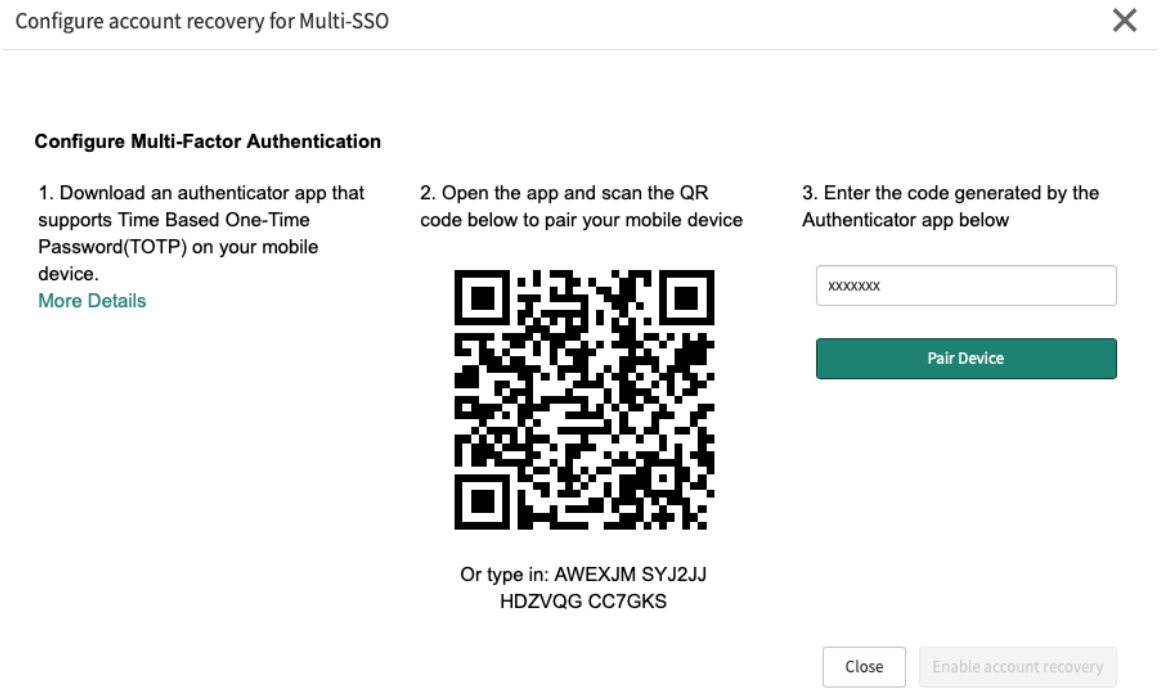
1. 移動先 すべて > アカウント復旧 > プロパティ.
2. [アカウント復旧の有効化] をクリックします。

   注: SSO が有効になっている場合は、アカウント復旧を有効にする必要があります。アカウント復旧ユーザーは、SSO 構成とトラブルシューティング関連のタスクに制限されます。

3. ステップ 2 にある文字 [ここ **(here)]** をクリックします。



4. [マルチ **SSO** のアカウント復旧を設定 **(Configure account recovery for Multi-SSO)]**モーダルの画面の指示に従って操作します。



画面の手順を完了すると、[アカウント復旧の有効化] が有効になります。

5. [アカウント復旧の有効化] をクリックします。

結果
ユーザーアカウントをアカウント復旧ユーザーとして設定しました。このアカウントを確認し、他の設定されているアカウント復旧ユーザーを確認するには、 マルチプロバイダー **SSO** > アカウント復旧 > ユーザー.

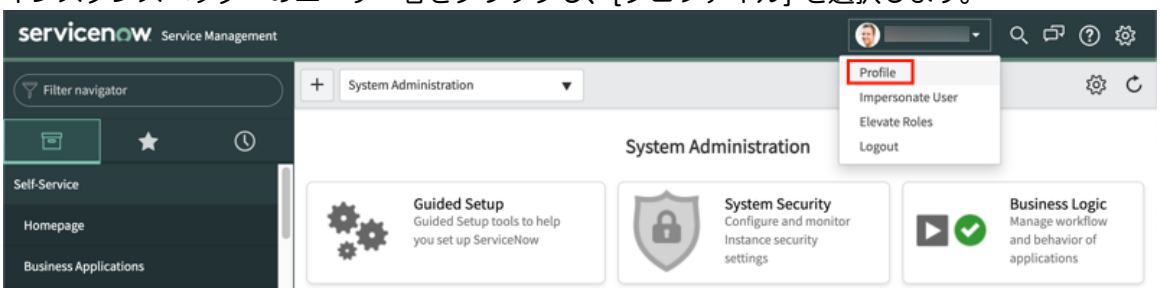
admin ユーザープロファイルからのアカウント復旧ユーザーの設定

アドミンユーザープロファイルから、アドミニストレーターをアカウント復旧ユーザーとして設定します。

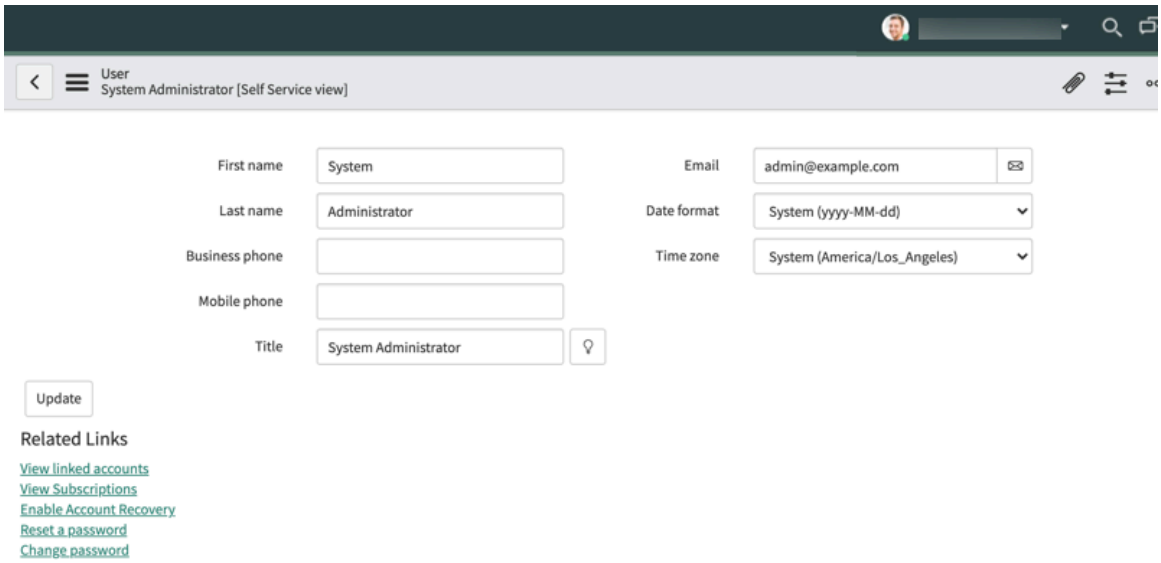
始める前に
必要なロール：admin

手順

1. アドミンアカウントを使用してインスタンスにログインします。
2. インスタンスヘッダーのユーザー名をクリックし、[プロファイル] を選択します。



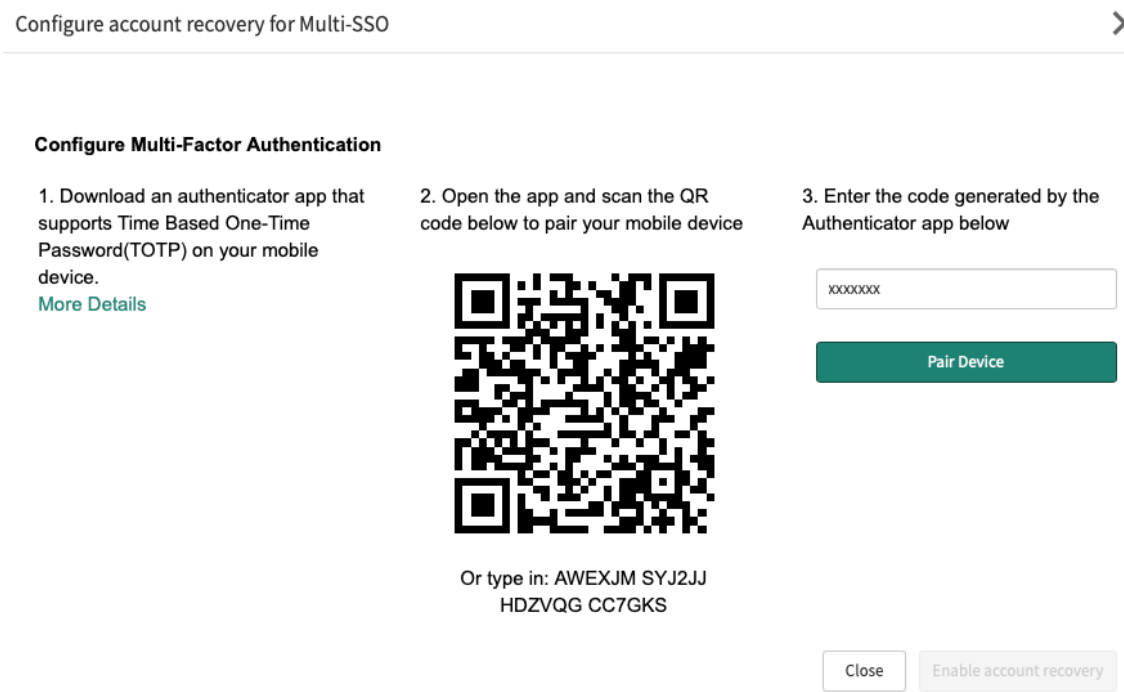
3. [ユーザー] フォームで、[関連リンク] セクションの [アカウント復旧の有効化] をクリックします。



注:

選択したユーザーが既にアカウント復旧を有効にしている場合、関連リンクに [アカウント復旧の有効化] は表示されません。代わりに [アカウント復旧の無効化] オプションが表示されます。

4. [マルチ **SSO** のアカウント復旧を設定 **(Configure account recovery for Multi-SSO)**]モーダルの画面の指示に従って操作します。



画面の手順を完了すると、[アカウント復旧の有効化] が有効になります。

5. [アカウント復旧の有効化] をクリックします。

結果
ユーザーアカウントをアカウント復旧ユーザーとして設定しました。このアカウントを確認し、他の設定されているアカウント復旧ユーザーを確認するには、 マルチプロバイダー **SSO** > アカウント復旧 > ユーザー.

アカウント復旧プロパティ

システムプロパティを使用して、インスタンスでアカウント復旧 (ACR) を設定します。

に移動して、インスタンスのアカウント復旧プロパティにアクセスします マルチプロバイダー **SSO** > アカウント復旧 > プロパティ.

アカウント復旧システムプロパティ

プロパティ	説明
アカウント復旧機能を有効にします [glide.sso.acr.enabled]	インスタンスでアカウント復旧機能が有効になっているかどうか。このプロパティはデフォルトで有効になっています。
アカウント復旧のデバッグログ記録を有効にします [glide.sso.acr.debug.log.enabled]	インスタンスのデバッグログにアカウント復旧情報が含まれているかどうか。このプロパティは、デフォルトでは無効になっています。
ACR ユーザーセッションタイムアウト (分) [glide.sso.acr.ui.session.timeout]	インスタンスがアカウント復旧ユーザーセッションを終了するまでの非アクティブ時間 (分) このプロパティのデフォルト値は 30 です。

マルチプロバイダー **SSO** の電子署名

マルチプロバイダー SSO を使用した電子署名では、SAML または OIDC プロパティの代わりに電子署名プロパティを認証に使用できます。

認証中のシングルサインオン (SSO) 検証で、電子署名を送信する前にユーザーに認証情報の提供を要求する場合は、署名を送信する前にユーザー認証情報の入力を求めるように認証を構成できます。

Approvals with e-signature (com.glide.e_signature_approvals) プラグインをインストールして、SSO ログインの電子署名を構成できます。

> **注:** 電子署名プラグインをインストールするには、コード署名の署名 (com.glide.code_signing.signatures) をインストールする必要があります。

電子署名による承認プラグインのアクティブ化

電子署名による承認プラグイン (com.glide.e_signature_approvals) を使用すると、ユーザーはログイン認証情報を再入力して要求を承認することができます。

始める前に

必要なロール：admin

> **注:** 電子署名プラグインをインストールするには、コード署名の署名 (com.glide.code_signing.signatures) をインストールする必要があります。

手順

1. 移動先 すべて > システムアプリケーション > 利用可能なすべてのアプリケーション > すべて.
2. フィルター基準と検索バーを使用してプラグインを検索します。

   名前または ID でプラグインを検索できます。プラグインが見つからない場合は、ServiceNow 担当者から要求する必要があります。
3. [インストール] を選択して、インストールプロセスを開始します。

注: ドメインセパレーションと代理アドミンがインスタンスで有効になっている場合、管理ユーザーはグローバルドメインに含まれている必要があります。それ以外の場合、次のエラーが表示されます： 「別の操作が実行されているため、アプリケーションのインストールは利用できません： <プラグイン名> のプラグインの有効化 (Application installation is unavailable because another operation is running: Plugin Activation for <plugin name>)」

インストールが完了するとメッセージが表示されます。プラグインとともにインストールされるコンポーネントの詳細については、「 アプリケーションとともにインストールされているコンポーネントの検索 」を参照してください。

複数プロバイダー SSO を使用した SAML 2.0 認証の SSO 承認の設定

電子署名による SSO 承認では、SAML IdP および ServiceNow インスタンスの設定が必要です。

始める前に
必要なロール：admin

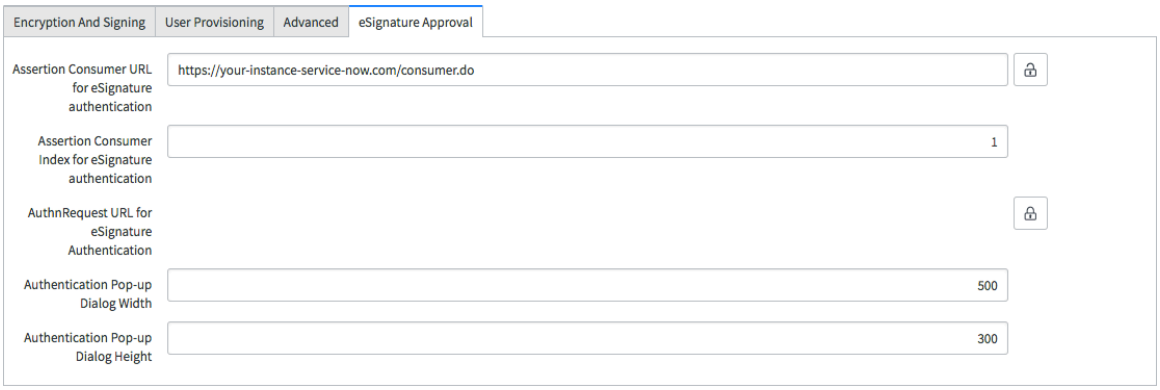
このタスクについて
SAML IdP は、SAML アサーション要求で forceAuthn 属性をサポートし、優先する必要があります。この IdP 設定がないと、電子署名は機能しません。SAML 2.0 認証の認証情報を使用して電子署名による承認を設定します。

手順

1. Activate Multi-Provider SSO プラグインを使用して、SAML 2.0 を有効にするか、SAML 2.0 にアップグレードします。
2. Approval with E-Signature プラグインをアクティブ化します。
3. 移動先 マルチプロバイダー **SSO** > **ID** プロバイダー をクリックし、2.0 SAML IdP 構成の [詳細] タブに [**AuthnRequest** の強制] 属性がオンになっていることを確認します。
   SAML 2.0 IdP が **[AuthnRequest** の強制] 属性をサポートしている必要があります。サポートしていない場合、電子署名はサポートされません。
4. [eSignature の承認] タブで、次の電子署名の SAML プロパティを入力します。

オプション	説明
eSignature 認証用の Assertion Consumer URL	このプロパティのデフォルトは、適切な URL です。このプロパティを設定するには、ロックアイコンを選択してこのフィールドを編集可能にします。編集後、そのアイコンを選択してフィールドをロックします。
eSignature 認証用の Assertion Consumer Index	サービス プロバイダーが AssertionConsumerURL の複数の URL セットを保有している場合は、eSignature で使用するためにインデックス 1 以上で始まるインデックスを設定できます。
eSignature 認証用の AuthnRequest URL	eSignature 認証用の SAML 2.0 IdP AuthnRequest URL を示す URL を入力できます。URL が Assertion Consumer URL と同じである場合は、この設定を空のままにすることができます。
認証ポップアップダイアログの幅	ユーザーが eSignature を使用して要求を承認すると、ポップアップが開き、認証情報を入力

オプション	説明
	できます。この設定は、ダイアログ ボックスの幅を制御します。デフォルトは 500 です。
認証ポップアップダイアログの高さ	ユーザーが eSignature を使用して要求を承認すると、ポップアップが開き、認証情報を入力できます。この設定は、ダイアログ ボックスの高さを制御します。デフォルトは 300 です。



5. タブの下にある [メタデータを生成] ボタンを選択して、サービス プロバイダーのメタデータを再生成します。
6. サービスプロバイダーのメタデータをコピーし、SAML IdP でそれを更新します。

複数プロバイダー **SSO** を使用した **OIDC** 認証の **SSO** 承認の設定

電子署名による SSO 承認では、SAML IdP および ServiceNow インスタンスの設定が必要です。

始める前に

必要なロール：admin

このタスクについて

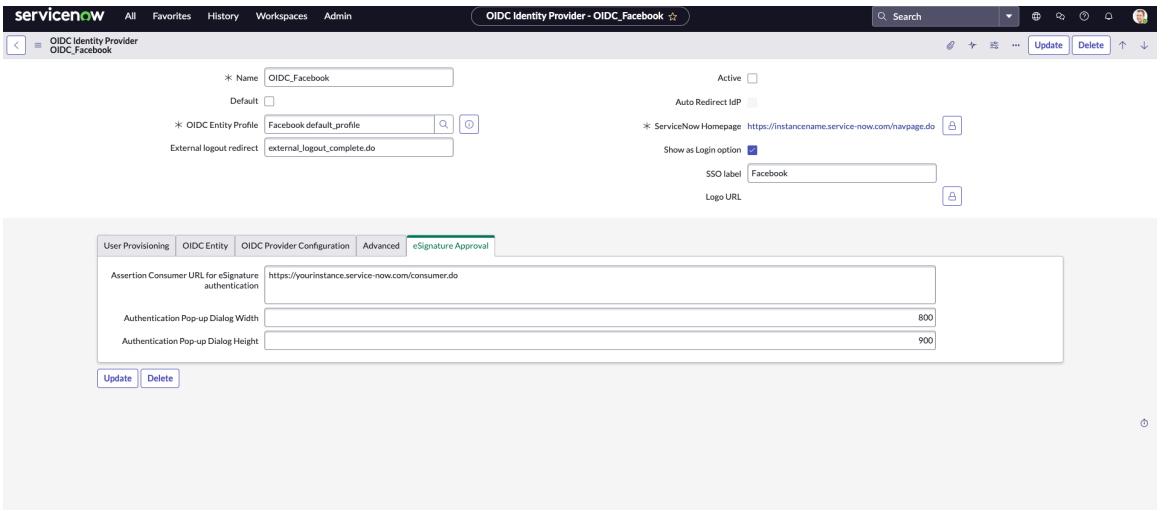
SAML IdP は、SAML アサーション要求で forceAuthn 属性をサポートし、優先する必要があります。この IdP 設定がないと、電子署名は機能しません。SAML 2.0 認証の認証情報を使用して電子署名による承認を設定します。

手順

1. Approval with E-Signature プラグインをアクティブ化します。
2. 移動先 マルチプロバイダー **SSO** > **ID** プロバイダー および OIDC プロバイダー構成の確認
3. [eSignature の承認] タブで、次の電子署名の SAML プロパティを入力します。

オプション	説明
eSignature 認証用の Assertion Consumer URL	このプロパティのデフォルトは、適切な URL です。このプロパティを設定するには、ロックアイコンを選択してこのフィールドを編集可能にします。編集後、そのアイコンを選択してフィールドをロックします。
認証ポップアップダイアログの幅	ユーザーが eSignature を使用して要求を承認すると、ポップアップが開き、認証情報を入力できます。この設定は、ダイアログボックスの幅を制御します。デフォルト値は 800 です。

オプション	説明
認証ポップアップダイアログの高さ	ユーザーが eSignature を使用して要求を承認すると、ポップアップが開き、認証情報を入力できます。この設定は、ダイアログボックスの高さを制御します。デフォルトは 900 です。



4. OIDC の初期セットアップ中に電子署名を設定する場合は [送信] を選択し、電子署名の詳細を更新する場合は [更新] を選択します。

シングルサインオン **(SSO) ID** プロバイダー **(IdP)** としての **OpenID Connect (OIDC)**

OpenID Connect (OIDC) は OAuth プロトコル上にビルドされた ID レイヤーで、最新の直感的なシングルサインオン (SSO) エクスペリエンスをユーザーやエンドユーザーに提供します。

OIDC は、ユーザーがソーシャル ID プロバイダーを使用して ServiceNow アプリケーションにログインできるようにすることで、モバイルアプリケーションのログインエクスペリエンスを向上させます。たとえば、アドミニストレーターは、OpenID Connect をサポートするサードパーティの ID プロバイダーを使用してシングルサインオンを設定できます。ユーザーは、ID プロバイダーの資格情報を使用してカスタム ServiceNow アプリケーションにログインできます。

企業/顧客間 (B2C) ユーザーには Google などのソーシャル ID プロバイダーを使用し、企業間ユーザーには Okta、 Microsoft Azure AD などのエンタープライズ ID プロバイダーを使用できます。

Single Sign-on (SSO) 用の **OpenID Connect (OIDC)** 構成の作成

Multi-Provider SSO プラグインを使用して、OpenID Connect (OIDC) 構成を作成または更新します。

シングルサインオン (SSO) 用の OpenID Connect (OIDC) の構成

始める前に

- OIDC アプリケーションを ID プロバイダー (IdP) に登録し、クライアント ID、クライアントシークレット、および既知の構成 URL をメモします。
- 複数プロバイダー SSO プラグインをアクティブ化します。複数プロバイダー SSO 機能はドメインセパレーションをサポートしており、ドメインごとに異なる IdP を指定できます。
- マルチプロバイダー SSO プロパティの設定

- Approval with e-Signature プラグインをアクティブ化して、OIDC IdP の電子署名を有効にします。
- 必要なロール：admin

ID プロバイダーのクライアント ID、クライアントシークレット、および既知の構成 URL がある場合は、SSO の OIDC 構成を直接インポートできます。

注:

- ドメインセパレーションプラグインがインストールされている場合、OIDC ベースの IDP のログインページに [**OIDC** でログイン] ボタンは表示されません。
- アドミニストレーターは、OIDC ベースの IdP を使用して、特定の会社またはドメインのユーザーの SSO を有効にすることができます。

ID プロバイダーに関する必要な情報がない場合は、手動で SSO 用に OIDC を設定できます。構成が完了すると、ユーザーは Google Okta などのサードパーティのソーシャル ID プロバイダーを使用して ServiceNow アプリケーションにログインできます。

手順

1. 移動先 すべて > マルチプロバイダー **SSO** > **ID** プロバイダー.
2. 次のいずれかのオプションを選択します。
   - 既存の構成を更新するには、OIDC ID プロバイダーレコードをクリックします。
   - 新しい構成を作成するには、[新規] をクリックし、**[OpenID Connect]** を選択します。
3. 新しい構成の場合は、次のいずれかの方法で OIDC 構成情報を入力します。

オプション	説明
OpenID Connect の既知の構成をインポート	関連するクライアント認証情報とともに既知の構成 URL がある場合は、OIDC 構成を直接インポートできます。 注: OIDC の既知の構成をインポートすると、関連するすべてのフィールドが自動入力されます。
OIDC ID プロバイダーフォームの手動構成	既存の OAuth OIDC エンティティがない場合は、[OpenID Connect の既知の構成をインポート] ポップアップを閉じて、[OIDC ID プロバイダー] フォームのフィールドに手動で入力します。

[OpenID Connect の既知の構成をインポート] のフィールド

プロパティ	説明
名前	OIDC ID プロバイダー構成の一意の名前。
クライアント ID	サードパーティの OIDC ID プロバイダーに登録されているアプリケーションのクライアント ID。
クライアントシークレット	サードパーティの OIDC ID プロバイダーに登録されているアプリケーションのクライアントシークレット。

プロパティ	説明
既知の設定 URL	サードパーティ OIDC ID プロバイダーに関するメタデータを含む URL。

[OIDC ID プロバイダー] フォームのすべての必須フィールドに入力する必要があります。

OIDC ID プロバイダーフォームに手動で入力する前に、OIDC IdP の OAuth エンティティプロファイルがあることを確認してください。

OAuth エンティティプロファイルがない場合は、Okta、Azure などのデフォルトの外部 OIDC プロバイダーテンプレートを使用して作成できます。

OAuth エンティティプロファイルの権限許可タイプには、認証コードが入っている必要があります。詳細については、次を参照してください。 での OAuth OIDC プロバイダーの構成 ServiceNow AI Platform.

注: Multiple Provider Single Sign-On Installer プラグインのデモデータでは、サードパーティの ID プロバイダーである Auth0、Azure AD、Google、および Okta のテンプレートを使用できます。

OIDC ID プロバイダーのフィールド

プロパティ	説明
名前	OIDC ID プロバイダーレコードの名前
有効	OIDC IdP 構成をアクティブにするオプション **注:** このオプションは、テスト接続が成功した後にのみアクティブに設定できます。
デフォルト	OIDC 構成が複数ある場合に、OIDC IdP 構成をデフォルトとして設定するオプション
自動リダイレクト IdP	ID プロバイダーのログインページへユーザーを自動リダイレクトできるようにするオプション。このフィールドは、[自動リダイレクト **IdP** として設定] オプションが [関連リンク] セクションで設定されている場合に表示されます。 **注:** 新しい自動リダイレクト IdP 構成をアクティブにすると、*glide_sso_id* cookie が新しい自動リダイレクト IdP で自動的に更新されます。*glide.authenticate.sso.update.idp.cookie* システムプロパティは、この機能を制御します。
OIDC エンティティプロファイル	OIDC 構成の OAuth エンティティプロファイル。
ServiceNow のホームページ	認証に使用されるログイン ページの URL。このフィールドは、自動的にインスタンス URL に設定されます。URL の形式は https://yourinstance.service-now.com/navpage.do です。
外部のログアウトのリダイレクト	ログアウト後に統合によりリダイレクトされる URL。通常は、SSO に使用されるポータルです。このフィールドは自動的に external_logout_complete.do に設定されます (例：https://yourinstance.service-now.com/external_logout_complete.do)。

自動翻訳

プロパティ	説明
ログインオプションとして表示	OIDC IdP をログインページのログインオプションとして表示するオプション。ログインオプションは 、**[ID** プロバイダーでログイン **(login with Identity provider)]** ボタンとして表示されます。
SSO ラベル	ログインページに表示される OIDC IdP のラベル。このフィールドは、[ログインオプションとして表示] が有効になっている場合にのみ表示されます。
ロゴ URL	OIDC IdP プロバイダーのロゴを含む公開 URL。このフィールドは、[ログインオプションとして表示] が有効になっている場合にのみ表示されます。

4. オプション: [ユーザープロビジョニング] タブ > [ユーザープロビジョニング] タブで自動ユーザープロビジョニングを有効にします。

   (Optional) ユーザーのログイン時に自動ユーザープロビジョニングを有効にするよう選択できます。自動ユーザープロビジョニングが有効になっている場合、ユーザーレコードが存在しなければ ServiceNow インスタンスに自動的に作成されます。

   [ユーザープロビジョニング] のフィールド

プロパティ	説明
ユーザーを自動プロビジョニング	自動ユーザープロビジョニングを有効にするオプション。このプロパティにより、ユーザーが IdP に存在するがユーザーテーブルに存在しない場合に、インスタンスユーザー (sys_user) テーブルにユーザーが作成されます。
プロビジョニングに使用	ID トークン、ユーザー情報エンドポイント、または ID トークンとユーザー情報の両方を ServiceNow ユーザーに変換するために使用するデータソース。ルックアップリストを使用して、事前定義されたデータソーステンプレートを選択し、レコードを開いて変換テーブルのマッピングを構成します。
データソースをプロビジョニング	ユーザーのプロビジョニングに使用される ID トークンデータソース。
ユーザー情報データソース	ユーザープロビジョニングに使用されるユーザー情報エンドポイントデータソース。このフィールドは、[プロビジョニングに使用] フィールドで [ユーザー情報] または [ID トークンとユーザー情報の両方 (Both ID Token and User Info)] が選択されている場合に表示されます。
次回のログイン時にユーザーを更新	次回のログイン時にユーザーを更新できるようにするオプション。
ユーザー更新の時間間隔 (秒)	後続のログインの間にユーザーレコードを更新するための最小時間間隔 (秒)。このフィールドは自動的に 3,600 秒に設定されます。たとえば、ユーザーがログインした後、次のログインまで 3,600 秒後にユーザーレコードが更新されます。このフィールドは、 [次回のログイン時にユーザーを更新] フィールドが有効になっている場合にのみ使用できます。
プロビジョニングされたユーザーに適用されるユーザーロール	新しくプロビジョニングされたユーザーに適用されるロールのリスト。

5. [OIDC エンティティ] タブ
   エンティティレコードを使用して、OIDC クライアント構成と OIDC 接続フローを表示および変更できます。
6. [OIDC プロバイダー構成] タブ
   OIDC IdP または ID トークン要求検証の既知の構成 URL を表示および変更できます。
7. オプション: [詳細] タブ

   (Optional) シングルサインオンおよびログアウト時に実行されるスクリプト

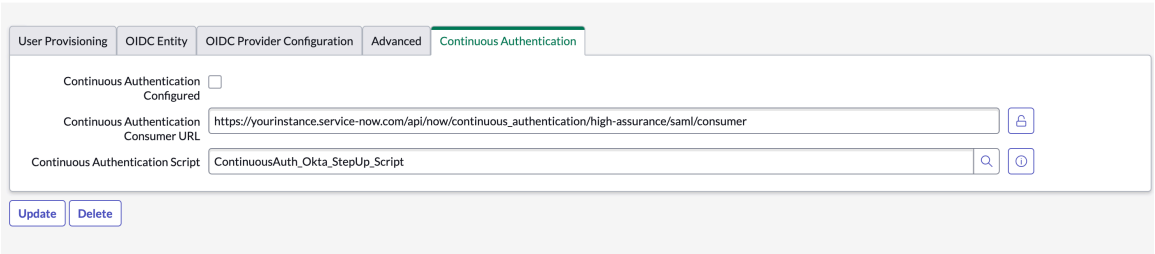
   [詳細] フィールド

プロパティ	説明
シングルサインオンスクリプト	シングルサインオン 中に実行されるスクリプト。このフィールドは自動的に MultiSSO_OIDC_custom に設定されます。
ログアウトスクリプト	ユーザーがログアウトした後に実行されるスクリプト。このフィールドは自動的に MultiSSO_OIDC_logout_custom に設定されます。

8. オプション: [継続認証] タブで、次のフィールドを設定します。

   注:

   - [継続的認証] タブは、ライセンスが必要な **Zero Trust - Continuous Authentication** (com.snc.zero_trust_continuous_authentication) プラグインをインストールした場合にのみ表示されます。
   - 継続的認証ポリシーを使用してテーブルまたはデータクラスへのアクセスを保護する場合は、「 継続認証 (CA)」を参照してください。



継続認証

フィールド	説明
継続認証が構成されました	チェックボックスをオンにして、構成をアクティブに設定します。
継続認証コンシューマー URL	ID プロバイダーからのコンシューマー URL を指定します。
継続認証スクリプト	(Optional) ルックアップアイコンを選択して、プラットフォームから提供されたスクリプトを選択します。この構成では、OIDC Okta の場合: **ContinuousAuth_Okta_StepUp_Script**

9. オプション: [電子署名の承認 (eSignature Approval)] タブで、OIDC IDp の電子署名を構成します。

注: [電子署名の承認 (eSignature Approval)] タブは、 **Approval with e-Signature** プラグイン (com.glide.e_signature_approvals) をインストールした場合にのみ表示されます。

[電子署名の承認 **(eSignature Approval)]** フィールド

プロパティ	説明
eSignature 認証用の Assertion Consumer URL	eSignature の OIDC 認証を処理する方法をカスタマイズして使用する場合は、独自のコンシューマー URL を設定できます。たとえば、マルチプロバイダー SSO を使用している場合は、このプロパティを使用する必要はありません。URL の形式は https://yourinstance.service-now.com/consumer.do です。
認証ポップアップダイアログの幅	認証ポップアップダイアログの幅。このフィールドは自動的に 800 に設定されます。
認証ポップアップダイアログの高さ	認証ポップアップダイアログの高さ。このフィールドは自動的に 900 に設定されます。

10. オプション: インスタンスのログインページに移動して、IdP がログインオプションとして表示されていることを確認します。
URL は、次の形式にする必要があります。https://yourinstance/login_with_sso.do?glide_sso_id=sysId_IdP

注: [ログインオプションとして選択 **(Selected as login Option)]** を有効にしている場合は、インスタンスのログイン URL に移動できます。

Facebook ベースのシングルサインオン **(SSO)** の使用

Facebook ベースの SSO で Facebook 認証情報を使用して、ServiceNow インスタンスにログインします。

始める前に
Facebook ベースの SSO は ServiceNow インスタンスとともに出荷されます。

ID プロバイダー (Idp) 構成を、使用する **ID** プロバイダーとして **OIDC_Facebook** IdP に定義できます。Idp 構成の詳細については、「Facebook ベースのシングルサインオン (SSO) の構成」を参照してください。

必要なロール：admin

手順

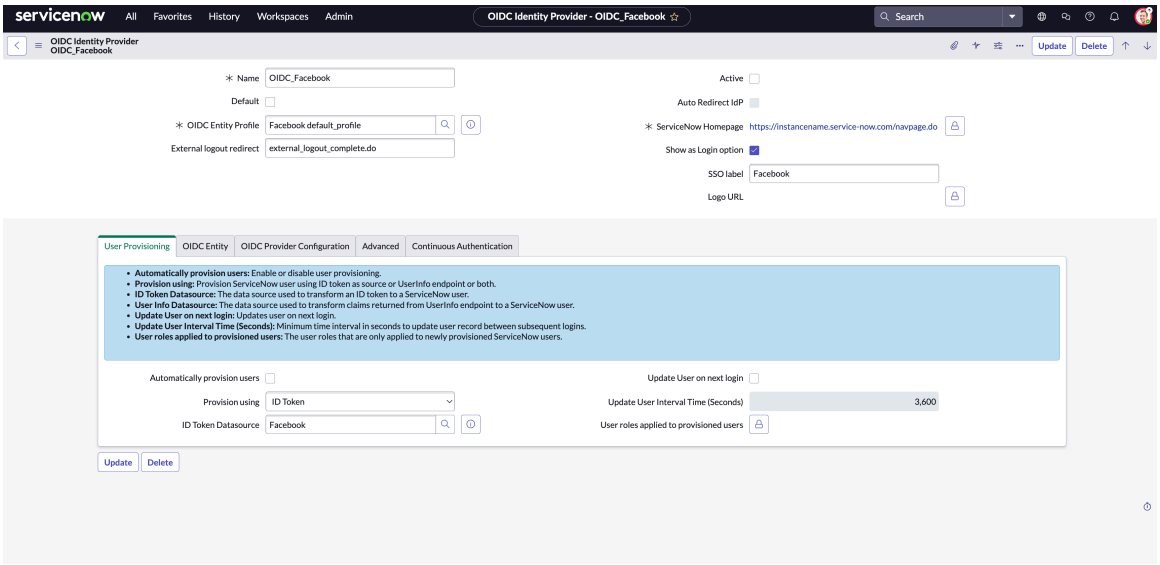
1. 移動先 すべて > マルチプロバイダー **SSO** > **ID** プロバイダー.
2. **[OIDC_Facebook]** を選択します。
3. [OIDC_Facebook] ページで、次のフィールドを指定します。

注:

- デフォルトの IdP を使用すると、ほとんどのフィールドが自動入力されます。
- ServiceNow ホームページの詳細を指定する必要があります。
- Facebook からのクライアント ID やクライアントシークレットなどのユーザー関連の詳細を、IdP 内に指定する必要があります。

OIDC_Facebook ID プロバイダーの詳細

フィールド	説明
名前	OIDC IdP レコードの名前を記録します。「OIDC_Facebook」と入力します。
デフォルト	OIDC IdP 構成をデフォルトとして設定するオプション。
OIDC エンティティプロファイル	OIDC 構成の OAuth エンティティプロファイル。「Facebook default_profile」と入力します。
外部のログアウトのリダイレクト	ログアウト後に統合によりリダイレクトされる URL。通常、この URL は SSO に使用されるポータルです。このフィールドは自動的に **external_logout_complete.do** に設定されます。例：https://<yourinstance>.service-now.com/external_logout_complete.do
有効	OIDC IdP 構成をアクティブにするオプション **注:** このオプションは、テスト接続が成功した後にのみアクティブに設定できます。
自動リダイレクト IdP	ID プロバイダーのログインページへユーザーを自動リダイレクトできるようにするオプション。
ServiceNow のホームページ	認証に使用されるログイン ページの URL。このフィールドは、自動的にインスタンス URL に設定されます。URL の形式は https://<インスタンス>.service-now.com/navpage.do です。
ログインオプションとして表示	OIDC IdP をログインページのログインオプションとして表示するオプション。この場合、ログインオプションは **[Facebook** でログイン **(Login with Facebook)]** ボタンとして表示されます。
SSO ラベル	ログインページに表示される OIDC IdP のラベル。このフィールドは、[ログインオプションとして表示] が有効になっている場合にのみ表示されます 。
ロゴ URL	OIDC IdP プロバイダーのロゴを含む公開 URL。このフィールドは、[ログインオプションとして表示] が有効になっている場合にのみ表示されます 。



4. オプション: [ユーザープロビジョニング] タブを開き、フィールドに入力します。

> **注:** Facebook から、ユーザーのクライアント ID やクライアントシークレットなどの OIDC 関連情報を設定する必要があります。

[ユーザープロビジョニング] タブ

フィールド	説明
ユーザーを自動プロビジョニング	自動ユーザープロビジョニングを有効にするオプション。このプロパティにより、ユーザーが IdP に存在するがユーザーテーブルに存在しない場合に、インスタンスユーザー [sys_user] テーブルにユーザーが作成されます。 **注:** ユーザーのログイン時に自動ユーザープロビジョニングを有効にするよう選択できます。自動ユーザープロビジョニングが有効になっている場合、ユーザーレコードが存在しなければ ServiceNow インスタンスに自動的に作成されます。
プロビジョニングに使用	ServiceNow ユーザーに変換するために使用するデータソース。選択肢は次のとおりです。 ○ **ID** トークン ○ ユーザー情報エンドポイント ○ **ID** トークンとユーザー情報の両方 ルックアップリストを使用して、事前定義されたデータソーステンプレートを選択し、レコードを開いて変換テーブルのマッピングを構成します。
データソースをプロビジョニング	ユーザーのプロビジョニングに使用される ID トークンデータソース
ユーザー情報データソース	ユーザープロビジョニングに使用されるユーザー情報エンドポイントデータソース。このフィールドは、[プロビジョニングに使用] フィールドで [ユーザー情報] または **[ID** トークンとユーザー情報の両方] が選択されている場合にのみ表示されます。

フィールド	説明
次回のログイン時にユーザーを更新	次回のログイン時にユーザーを更新できるようにするオプション。
ユーザー更新の時間間隔 (秒)	後続のログイン間でユーザーレコードを更新するための最小時間間隔 (秒)。このフィールドは自動的に 3,600 秒に設定されます。たとえば、ユーザーがログインした後、次のログインまで 3,600 秒後にユーザーレコードが更新されます。このフィールドは、 [次回のログイン時にユーザーを更新] フィールドが有効になっている場合にのみ使用できます。
プロビジョニングされたユーザーに適用されるユーザーロール	新しくプロビジョニングされたユーザーに適用されるロールのリスト。

5. **[OIDC** エンティティ] タブでエンティティレコードを使用して、OIDC クライアント構成と OIDC 接続フローを表示および変更できます。
   OIDC ベースの構成の詳細については、「サードパーティトークンを受け入れるための OAuth OIDC プロバイダーの構成」を参照してください。
6. **[OIDC** プロバイダー構成] タブで、OIDC IdP の既知の構成 URL を表示して変更します。
7. オプション: [詳細] タブを開き、フィールドに入力します。

   [詳細] タブ

プロパティ	説明
シングルサインオンスクリプト	シングルサインオン中に実行されるスクリプト。
ログアウトスクリプト	ユーザーがログアウトした後に実行されるスクリプト。

   注: スクリプトは、シングルサインオンおよびログアウト時に実行されます。

8. 構成を有効にしてテストするには、[アクティブ **(Active)]** をクリックします。
9. レコードを更新するには、[更新] をクリックします。
   Facebook ベースのログインオプションがログインフォームに表示されます。
10. ログインフォームでログインするときは、次の操作を行います。
    a. Facebook オプションを選択します。
    b. ServiceNow インスタンスにログインするには、Facebook 認証情報を指定します。

Facebook ベースのシングルサインオン **(SSO)** の構成

Facebook ベースの SSO を ServiceNow インスタンスに構成します。

始める前に

Facebook から IdP として構成された有効なクライアント ID が必要です。

以下のプロパティを有効にします。

- マルチプロバイダー SSO を有効にします。
- マルチプロバイダー SSO 統合に対してデバッグログを有効にします。

必要なロール：admin

手順

1. 移動先 すべて > マルチプロバイダー **SSO** > **ID** プロバイダー.
2. 新しい Facebook ID プロバイダーを作成するには、[新規] をクリックします。
3. **[OpenID Connect]** をクリックします。
4. フォームのフィールドに入力します。

[OpenID Connect の既知の構成をインポート] フォーム

フィールド	説明
名前	OIDC ID プロバイダー構成の一意の名前。
クライアント ID	サードパーティの OIDC ID プロバイダーに登録されているアプリケーションのクライアント ID
クライアントシークレット	サードパーティの OIDC ID プロバイダーに登録されているアプリケーションのクライアントシークレット
既知の設定 URL	サードパーティ OIDC ID プロバイダーに関するメタデータを含む URL。

5. [インポート] をクリックします。
   Facebook ベースの IdP が作成されます。
6. Facebook IdP を選択します。
7. Facebook idP で、次の操作を行います。
   a. [名前]、**[OIDC** エンティティプロファイル]、[外部のログアウトのリダイレクト]、**[ServiceNow** ホームページ] などのすべてのフィールドを検証します。
   b. **SSO** ラベルを入力します。
8. [ユーザープロビジョニング] タブで、ユーザーを特定のユーザープロビジョニングとロールに設定するために必要なフィールドを指定します。

   必要なのは必須フィールドのみです。その他のフィールドは、必要に応じて指定できます。
9. **[OIDC** エンティティ] タブで、次の操作を行います。
   a. エンティティをクリックします。
   b. [リダイレクト **URL]** フィールドを Facebook リダイレクト URL に設定します。
10. **[OAuth** エンティティプロファイル] タブで、次の操作を行います。
    a. プロファイルの詳細で、プロファイルをクリックします。
    b. スコープを選択し、詳細を確認します。
       たとえば、**[scope-1]** を選択します。
11. [**OAuth** エンティティスコープ] タブで、[スコープ **1**] リンクをクリックし、スコープを `メール`として追加します。

12. 構成を保存するには、ヘッダーを右クリックして [保存] をクリックします。
13. 構成をアクティブとして設定するには、[アクティブ **(Active)]** を選択します。

結果
ログインフォームが Facebook SSO オプションとともに表示されます。

SAML

セキュリティアサーションマークアップ言語 (SAML) は、セキュリティドメイン間で認証および承認データを交換するための XML ベースの規格です。

SAML は、ID プロバイダー (アサーションのプロデューサー) とサービスプロバイダー (アサーションのコンシューマー) の間でセキュリティ情報を交換します。SAML は OASIS セキュリティサービス技術委員会の製品です。正しく実装された場合、SAML は利用可能なシングルサインオン (SSO) の最も安全な方法のひとつになります。

SAML 2.0 統合では、XML トークンを外部の ID プロバイダー (IdP) と交換することで SSO が有効になります。IdP はユーザーを認証し、NameID トークンをシステムに渡します。一致する NameID トークン (メールアドレスなど) を持つユーザーが見つかると、インスタンスはそのユーザーをログインさせます。

SSO 認証に SAML 2.0 プラグインを使用している場合は、glide.ui.rotate_sessions プロパティを false に設定する必要があります。設定しない場合、インスタンスと ID プロバイダーの間で行われるセッション情報の共有が妨げられます。昇格された security_admin 権限を持つユーザーは、このプロパティにアクセスできます。

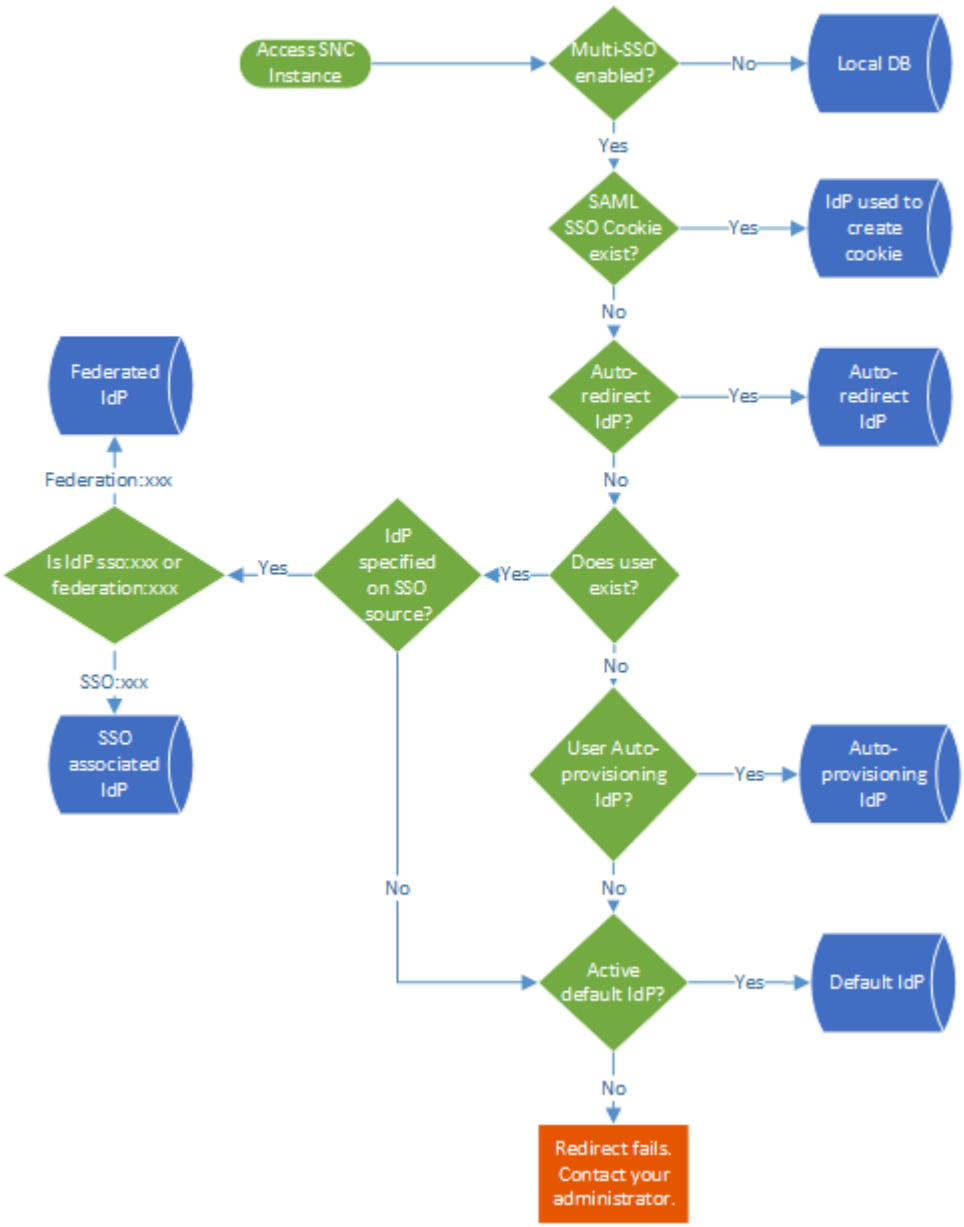
注: 既存の SAML 2.0 統合を使用しているお客様には、Multi-Provider SSO プラグインへのアップグレードをお勧めします。

複数プロバイダー **SSO (SAML) IdP** 認証フロー

SAML マルチ SSO を介してユーザーを認証できるさまざまなエンティティについて説明します。

認証フローに従うと、エンティティがマルチ SSO を使用してユーザーを認証するタイミングを把握できます。

Multi-SSO (SAML) IdP authentication flow



ローカル DB

マルチ SSO が有効になっていない場合、認証はローカル DB に送信されます。

SAML SSO Cookie IdP

SAML SSO cookie が存在する場合、この cookie で指定された IdP がユーザーを認証します。

自動リダイレクト IdP

自動リダイレクト IdP が有効になっている場合、この IdP がユーザーを認証します。

フェデレーション IdP

ユーザーのブラウザーが外部認証 (login_locate_sso.do) のログイン画面にリダイレクトされ、**[SSO** ソース] フィールドで IdP が [フェデレーション：xxx (federation:xxx)] に設定されたユーザーがユーザーテーブル内に存在する場合、フェデレーション IdP がユーザーを認証します。

自動翻訳

関連 IdP

ユーザーのブラウザーが外部認証 (login_locate_sso.do) のログイン画面にリダイレクトされ、**[SSO** ソース] フィールドで IdP が [SSO：xxx] に設定されたユーザーがユーザーテーブル内に存在する場合、関連 IdP がユーザーを認証します。

自動プロビジョニング IdP

ユーザーのブラウザーが外部認証 (login_locate_sso.do) のログイン画面にリダイレクトされ、ユーザーテーブル内にユーザーが存在しないが自動プロビジョニングが有効になっている場合は、自動プロビジョニング IdP がユーザーを認証します。

注: 複数の自動プロビジョニング IdP が有効になっている場合、ユーザーは使用可能な自動プロビジョニング IdP を選択できます。

デフォルト IdP

ユーザーのブラウザーが外部認証 (login_locate_sso.do) ログイン画面にリダイレクトされ、ユーザーが次のいずれかの場合：

- ユーザーテーブル内に存在せず、自動プロビジョニングが有効になっておらず、アクティブなデフォルト IdP がある場合
- ユーザーテーブル内に存在し、SSO ソースユーザーまたは会社レコードに IdP が指定されておらず、アクティブなデフォルト IdP がある場合

デフォルトの IdP がユーザーを認証します。

ID プロバイダー (IdP) システムのプロパティ

一般に、IdP は認証およびログアウトのメタデータを含む XML ドキュメントを提供します。

たとえば、SSOCircle はメタデータ をオンラインで公開しています。

IdP メタデータを参照して、次のエントリを見つけます。

- HTTP-Redirect の値を含む Binding 属性を持つ SingleSignOnService 要素。Location 属性には、AuthnRequest サービスのために統合に必要な URL がリストされます。
- HTTP-Redirect の値を含む Binding 属性を持つ SingleLogoutService 要素。Location 属性には、SingleLogoutRequest サービスに必要な URL がリストされます。

注: SAML 2.0 統合は、HTTP リダイレクトによる IdP サービスへのバインディングのみをサポートしています。

例：

```
<SingleSignOnServiceBinding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect"Location="https://idp.ssocircle.com:443/sso/SSORedirect/metaAlias/ssocircle"/>
```

```
<SingleLogoutServiceBinding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect"Location="https://
idp.ssocircle.com:443/sso/IDPSloRedirect/metaAlias/ssocircle"ResponseLocation="https://
idp.ssocircle.com:443/sso/IDPSloRedirect/metaAlias/ssocircle"/>
```

IdP 発行者 URL の設定

セキュリティトークンを発行する IdP の URL を指定します。

始める前に
必要なロール：admin

このタスクについて
統合では、各 SAML 応答に含まれ、このシステムプロパティにリストされている URL が、*Issuer* 要素にリストされている URL と同じであることが確認されます。たとえば、次のようになります。

```
<samlp:Responsexmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
 Destination="https://demoi2.service-now.com/navpage.do"
 ID="s28da6774c88ae1eab292bf25fe625db81919d8e1e"
InResponseTo="SNC841720c227c81948cfd68cadcad235c6"
IssueInstant="2012-01-30T20:07:10Z"  Version="2.0"><saml:Issuer
xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion">http://idp.ssocircle.com</saml:Issuer>
  ...
  <saml:Assertion xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
    ID="s2f347f973c063836cf70ea38302d94976f9c5b851"
IssueInstant="2012-01-30T20:07:10Z"
Version="2.0"><saml:Issuer>http://idp.ssocircle.com</saml:Issuer>
    ...
  </saml:Assertion></samlp:Response>
```

手順

1. 移動先 すべて > **SAML 2** シングルサインオン > プロパティ.
2. [ユーザー情報と一緒に *SAML2* セキュリティトークンを発行する *ID* プロバイダの *URL]* プロパティで IdP への URL を入力します。

   各 IdP URL は一意である必要があります。デフォルトでは、SSOCircle への URL が統合に含まれます。詳細については、「http://idp.ssocircle.com 」を参照してください。

AuthnRequest サービス **URL** の設定

IdP のメタデータを使用して、統合の IdP の要求サービス URL を設定します。

始める前に
必要なロール：admin

手順

1. プロパティ *[ID* プロバイダーの *AuthnRequest* サービスのベース URL。AuthnRequest は、この *URL* に *SAMLRequest* パラメーターとしてポストされます] で、SingleSignOnService 要素から取得した HTTP リダイレクトバインディングへの URL を入力します。
2. *[AuthnRequest* に署名] の横にあるチェックボックスをオンにすると、ID プロバイダーの Single Sign-on サービスが署名済みの AuthnRequest を受信できるようになります。
3. プロパティ [セッションが認証されないために *SAML 2.0* シングルサインオンに失敗した場合、またはこれが最初のログインである場合は、この *URL* にリダイレクトされます。これは、最初の *SAML 2.0 AuthnRequest* が *SAMLRequest* パラメーターを使用して送信されたベース *URL* です。] で、SingleSignOnService 要素から取得した HTTP リダイレクトバインディングへの URL を入力します。

   デフォルトでは、SSOCircle サービスへの URL が統合に含まれます。

SingleLogoutRequest サービス **URL** の設定

IdP のメタデータを使用して、統合の IdP の要求サービス URL を設定します。

始める前に
必要なロール：admin

自動翻訳

手順

1. *[ID プロバイダーの SingleLogoutRequest サービスのベース URL。LogoutRequest は、この URL に SAMLRequest パラメーターとしてポストされます]* プロパティで、SingleLogoutService 要素から取得された URL を入力します。
   LogoutRequest は、この URL に SAMLRequest パラメーターとしてポストされます。デフォルトでは、SSOCircle サービスへの URL が統合に含まれます。
2. [ログアウト後にユーザーをリダイレクトする URL。通常は *SSO* を有効にしたポータルに戻る *(http://portal.companya.com/logout* など)] プロパティで、ユーザーが正常にログアウトした後にリダイレクトする URL を入力します。

   IdP でフォームベースの認証を使用する場合は、IdP のログインフォームへの URL を入力します。IdP で Kerberos などのフォームベース以外の認証方法を使用する場合は、URL を静的ログアウトページに設定する必要があります。このようにすると、ログアウトしたユーザーがすぐに IdP にリダイレクトされて再度ログインさせられることはありません。デフォルトでは、統合には静的 UI ページ external_logout_complete.do への URL が含まれています。

(オプション) 署名付きログアウト要求の有効化

一部の IdP では、サービスプロバイダーが証明書を使用してログアウト要求に署名する必要があります。

始める前に
必要なロール：admin

このタスクについて
IdP で署名されたログアウト要求が必要な場合は、IdP のメタデータを使用して次のシステムプロパティを設定します。

手順

1. [詳細] タブで、プロパティ *[LogoutRequest* の署名] 署名済みの *LogoutRequest* を *ID* プロバイダーの *SingleLogoutRequest* サービスが必要とする場合は、このプロパティを *True* に設定します] では、[はい] を選択すると IdP には署名されたログアウト要求が必要であることが指定され、[いいえ] を選択すると署名なしのログアウト要求が使用されます。
2. *[LogoutRequest* の署名] に対して [はい] を選択した場合は、*[ID* プロバイダーの *SingleLogoutRequest* サービスのプロトコルバインディング(値には、「urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect」または「urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST」のいずれかを指定できます)] プロパティに、SingleLogoutService 要素の Binding 属性にリストされているサポート対象の値を 1 つ入力します。

   デフォルトでは、統合には HTTP リダイレクトバインディングが使用されます。

3. [更新] をクリックします。
4. サービスプロバイダー (SP) キーストアをインストールします。

サービスプロバイダー **(SP)** システムプロパティ

これらのシステムプロパティは、インスタンスがサービスプロバイダーとして IdP とやり取りする方法を定義します。

順序に従って操作し、IdP をサービスプロバイダーとして定義します。

SAML のインスタンス **URL** の設定

インスタンス固有の URL を設定して、IdP がユーザーを認証できるようにします。

始める前に
必要なロール：admin

手順

1. *[Service-now* インスタンス (通常はこのインスタンス) *(The URL to the Service-now instance (usually this instance))]* プロパティに、IdP が認証するインスタンスの URL (ログインページを含む) を入力します。
   例：https://yourinstance.service-now.com/navpage.do
2. プロパティ [エンティティ ID、または発行者] で、IdP が認証するインスタンスのベース URL (ログインページを除く) を入力します。
   例：https://yourinstance.service-now.com/

SAML の対象 URL の設定

対象者プロパティを使用して、SAML 応答の意図した受信者であることをインスタンスで確認できます。

始める前に
必要なロール：admin

このタスクについて
統合では、各 SAML 応答に含まれ、このシステムプロパティにリストされている URL が、*Audience* 要素にリストされている URL と同じであることが確認されます。たとえば、次のようになります。

```
<samlp:Responsexmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
 ID="s2cdc74f37f923e26fe1aeec42b70a93d24230334f"
InResponseTo="90AA6073F01567BFB0DF194F596314E2"
 Version="2.0"  IssueInstant="2010-04-29T23:21:51Z"
Destination="https://dloomac.service-now.com/navpage.do">
...
<saml:Conditions NotBefore="2012-01-30T19:57:10Z"
 NotOnOrAfter="2012-01-30T20:17:10Z"><saml:AudienceRestriction><saml:Audience>http
s://
demoi2.service-now.com</saml:Audience></saml:AudienceRestriction></saml:Conditions>
...
</samlp:Response>
```

手順

1. 移動先 すべて > **SAML 2** シングルサインオン > プロパティ.
2. プロパティ *[SAML2* トークンを受け入れる対象者 *URI* (通常、インスタンス URI。例：*https://<instance name>.service-now.com)]* で、インスタンスの URL を入力します。
   たとえば https://demoi2.service-now.com などです。この URL は、SAML 応答の *Audience* 要素の値と一致する必要があります。
3. [更新] をクリックします。

SAML の NameID ポリシーの設定

SAML の NameID ポリシーを設定します。SAML 2.0 では、IdP がサービスプロバイダーと NameID トークンを交換する必要があります。

始める前に
必要なロール：admin

このタスクについて
SAML 2.0 統合の場合、NameID トークンをユーザーテーブルの特定のフィールドにマッピングする必要があります。統合では、NameID トークンの値を使用して、IdP が認証するユーザーを決定します。

手順

1. IdP メタデータを参照して、emailAddress の値を含む NameIDFormat 要素を見つけます。この要素の値は、統合で使用されるデフォルトの形式です。
2. 他の NameIDFormat 要素を確認して、ユーザーテーブルの他のフィールドに一致する形式があるかどうかを判断します。

NameID トークンに一致するユーザーテーブルフィールドの判別

ID プロバイダーは、NameID トークンの形式を指定します。

始める前に
必要なロール：admin

このタスクについて
SAML 2.0 を設定するには、 NameID トークンの形式に一致するフィールドをユーザーテーブルから選択する必要があります。通常、IdP には NameID トークンとしてメールアドレスを使用するオプションがあります。ユーザーテーブルにはメールフィールドが含まれているため、このフィールドが NameID トークンとして使用するための論理的な選択肢となります。ユーザーテーブルの別のフィールドを NameID トークンとして使用する場合は、まず IdP がユーザーテーブルのフィールドの値と一致する NameID 形式を提供していることを確認します。この場合、ユーザーテーブルへフィールドの追加が必要になることがあります。

手順

1. IdP の NameIDFormat 要素で利用可能な形式を、ユーザーテーブルのフィールドと比較します。
2. ユーザーテーブルに一致する値がある NameID 形式を選択します。
3. *[SAMLResponse* の件名の *NameID* 要素と照合する [ユーザー] テーブルフィールド] フィールドに、NameID トークンで一致する値を検索するための [ユーザー] テーブルフィールドの名前を入力します。

   デフォルトでは、統合でメールフィールドを使用します。

IdP NameID ポリシーの設定

IdP が NameID トークンに使用する形式を指定します。

始める前に
必要なロール：admin

このタスクについて
この形式は、IdP のメタデータの一部としてリストされます。

手順

1. プロパティ *[SAMLResponse* の件名の *NameID* を返すために使用する *NameID* ポリシー。SAML *ID* プロバイダーは、そのメタデータでポリシーを宣言して、これをサポートする必要があります。NameID 値は、[ユーザー] テーブルの特定のフィールドと照合し、ユーザーを検索するのに使用されます。] に、統合で使用される NameIDFormat 要素の値を入力します。

デフォルトでは、統合でメールアドレスに SSOCircle NameIDFormat を使用します。

2. [保存] をクリックします。

SAML の [ユーザー] テーブルフィールドの値

統合の [ユーザー] テーブルフィールドには、適切な一致する値が確実に含まれるようにします。

たとえば、統合がメールフィールドを NameID トークンとして使用する場合は、インスタンスに IdP と同じメールアドレスがリストされていることを確認してください。統合では、NameID トークンに一致する値を持たないユーザーを認証できません。

(オプション) SAML の認証コンテキストクラスを提供可能にする

インスタンスの優先認証要求形式を含む認証コンテキストクラス要求を IdP に送信できるようにすることができます。

始める前に
必要なロール：admin

このタスクについて
AuthContextClass メッセージの作成を可能にする場合は、認証コンテキストクラス参照形式も指定する必要があります。

> **注:** IdP の中には、サービスプロバイダーによる認証コンテキストクラスの設定を許可しないものがあります。この設定を無効にすると、IdP が認証コンテキストクラスを選択できるようになります。

手順

1. *[AuthnRequest* ステートメントに *AuthnContextClass* 要求を作成します] プロパティから、[はい] を選択して Password Protected Transport などの特定のコンテキストクラスを指定するか、[いいえ] を選択して最も適切なコンテキストクラスを IdP が選択するようにします。
2. *[AuthnRequest* ステートメントに *AuthnContextClass* 要求を作成します] に [はい] を選択した場合は、 *[SAML 2.0 AuthnRequest to the Identity Provider* で要求する *AuthnContextClassRef* メソッド *(The AuthnContextClassRef method that we will request in our SAML 2.0 AuthnRequest to the Identity Provider)]* プロパティに、認証に使用するコンテキストクラスの URN を入力します (表を参照)。

AuthnContextClass URN オプション

認証タイプ	認証コンテキストクラス URN
フォームベースの認証	urn:oasis:names:tc:SAML:2.0:ac:classes:PasswordProtectedTransport
Kerberos ベースの認証	urn:federation:authentication:windows

デフォルトでは、Password Protected Transport 認証メソッドが統合で使用されます。

3. [更新] をクリックします。

(オプション) SAML のログアウト要求に署名するためのキーストアプロパティの設定

キーストアプロパティを設定すると、統合で署名済みサーバーと署名済み CA 証明書を使用してログアウト要求に署名できます。

始める前に
必要なロール：admin

手順

1. プロパティ *[SAML 2* 要求の署名に使用する *SAML 2.0 SP Keystore* に保存されているキーエントリのエイリアス *(The alias of key entry stored in SAML 2.0 SP Keystore used to sign SAML 2 requests)]* に、SAML 2.0 Keystore 用に作成したエイリアス名を入力します。
   デフォルトでは、統合はエイリアス *saml2sp* を検索します。
2. プロパティ *[SAML 2* 要求の署名に使用する *SAML 2.0 SP Keystore* に保存されているキーエントリのパスワード *(The password of key entry stored in SAML 2.0 SP Keystore used to sign SAML 2 requests)]* に、SAML 2.0 Keystore のパスワードを入力します。
   デフォルトでは、パスワードはデフォルトのエイリアス名と同じです。
3. [更新] をクリックします。
4. SP メタデータを再生成します。
   詳細については、「SP メタデータ」を参照してください。

SAML のサービスプロバイダーキーストアの作成

インスタンスがログアウト要求に署名できるように、次のアイテムを含む Java キーストアを作成します。

始める前に
必要なロール：admin

このタスクについて

- インスタンスの署名済みサーバー証明書
- 署名済み CA 証明書
- 公開鍵と秘密鍵のペア

プライベート認証局で独自の署名付き証明書を作成することも、パブリック認証局から購入することもできます。

次の手順は、新しい Java Keytool キーストアファイルを生成し、証明書の署名要求 (CSR) を作成し、証明書をインポートする方法を示しています。ルート証明書や中間証明書はいずれも、ドメインのプライマリ証明書をインポートする前にインポートする必要があります。コマンドラインインターフェイスで次のコマンドを入力します。

> **注:** これらの手順はプラットフォーム固有のものではなく、完了するためにはセキュリティ証明書に関する技術的な知識が必要です。テクニカルサポートは証明書の作成を支援できません。

手順

1. Java キーストアとキーペアを生成します。

```
keytool -genkey -alias mydomain -keyalg RSA -keystore my.keystore
```

2. 既存の Java キーストアの CSR を生成します。

```
keytool -certreq -alias mydomain -keystore my.keystore -file mydomain.csr
```

3. ルートまたは中間認証局 CA 証明書を既存の Java キーストアにインポートします。

```
keytool -import -trustcacerts -alias root -file Thawte.crt -keystore my.keystore
```

4. 署名済みプライマリ証明書を既存の Java キーストアにインポートします。

```
keytool -import -trustcacerts -alias mydomain -file mydomain.crt -keystore my.keystore
```

SAML 要求に署名するためのサービスプロバイダーキーストアのインストール

次の手順では、既存のサンプルキーストアを削除し、公開鍵と秘密鍵のペアを含む独自のサービスプロバイダーキーストアをインストールします。

始める前に
必要なロール：admin

手順

1. サービスプロバイダーキーストアを作成します。
2. 移動先 **SAML 2** シングルサインオン > 証明書 または マルチプロバイダー > 管理者 > **x509** 証明書.
3. **[SAML 2.0 Keystore_Key2048_SHA256]** をクリックします。
4. [添付ファイルを管理] リンクをクリックします。
5. *[saml2sp_key2048withsha256.jks]* の横にある [削除] チェックボックスをオンにします。
6. [削除] をクリックします。
7. [ファイルを選択] をクリックし、署名付き証明書を含むキーストアを選択します。
8. [添付] をクリックします。
9. [添付ファイル] ポップアップを閉じます。

   注: 新しく添付された証明書には別の名前を指定することをお勧めします。

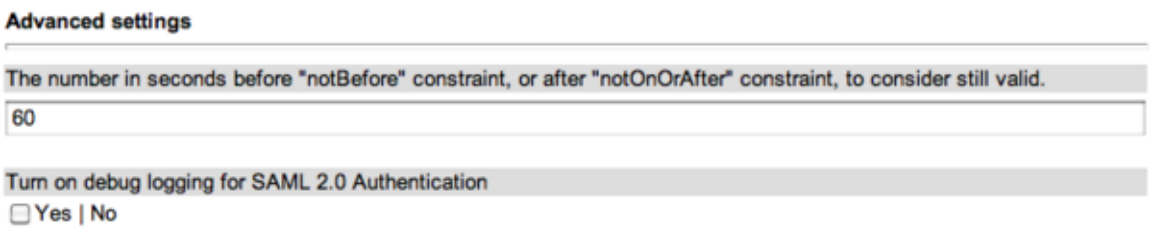
10. [キーストアパスワード] に、SAML 2 エイリアスにアクセスするためのパスワードを入力します。
11. [更新] をクリックします。

(オプション) 詳細な SAML プロパティ

次の詳細設定を使用すると、セキュリティを強化し、統合をデバッグできます。

詳細設定

移動先 すべて > **SAML 2** シングルサインオン > プロパティ.



詳細システムプロパティテーブル

プロパティ	説明
「notBefore」制約の前、または「notOnOrAfter」制約の後に、まだ有効であるとみ	IdP クロックと SP クロックの時差を考慮して、*NotBefore* および *NotOnOrAfter* の制約に追加する秒数を入力します。これらの制約は、指定された期間内に行われない要求を拒否することによって、リプレイ攻撃を防止します。IdP クロックと SP クロックが大幅に異なる場合、ネットワーク遅延により SAML 要求が認可されないことがあります。このプロパティ

詳細システムプロパティテーブル (続く)

プロパティ	説明
なされる時間の秒数。	は、SAML 要求と応答がまだ有効であると見なされる猶予期間を追加します。
SAML 2.0 認証のデバッグロギングをオンにします	[はい] を選択すると、SAML 2.0 イベントの追加のログ情報が有効になります。

ID プロバイダー証明書のインストール

ID プロバイダーがサービスプロバイダーとの通信を検証できるように、PEM 証明書を X.509 証明書フォームに貼り付けることができます。

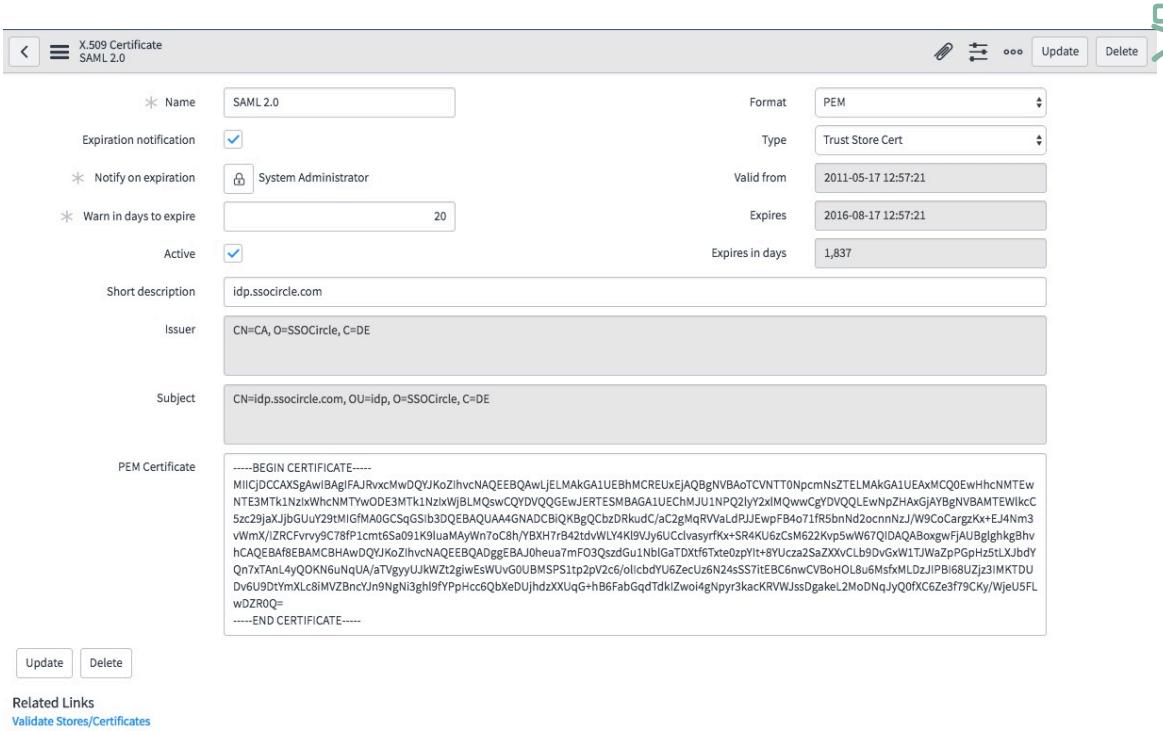
始める前に
必要なロール：admin

このタスクについて
IdP の証明書は IdP のメタデータ内にあります。IdP 開発者は、ローカル IdP を作成するときに証明書のメタデータをどこに置くか決定します。

注: Single Sign-on の証明書は、SAML 証明書を扱うため、常に PEM 形式になっている必要があります。

手順

1. 移動先 すべて > **SAML** シングルサインオン > 証明書.
2. フォームフィールドに入力します (表を参照)。
3. [保存] をクリックします。



注: 統合は現在、インスタンスと IdP 間の通信で証明書に署名しません。

フィールド	説明
名前	証明書名。[名前] エントリは変更しないでください。統合で X.509 証明書を使用するには、その名前が *SAML 2.0* である必要があります。この要件は、複数プロバイダーのシングルサインオン (SSO) を使用していない場合にのみ当てはまります。
有効期限通知	[期限切れ時に通知] フィールドで選択したユーザーに通知を送信するには、このオプションを選択します。デフォルトでは、これは有効になっています。
期限切れ時に通知	証明書の有効期限に関する通知を有効にするユーザーを選択します。ユーザーが選択されていない場合、ログインしているユーザーは、最後にログインした 2 人のアドミニストレーターロールとともに追加されます。
有効期限切れ前に警告	インスタンスが通知を送信する期限切れまでの日数。20 以上の値を入力してください。Istanbul 以降のリリースにアップグレードしたインスタンスでは、より大きい値を指定しない限り、この値は 20 に設定されます。
アクティブ	この証明書がアクティブであることを示すチェックボックス。
フォーマット	証明書の形式。SAML は PEM 形式を使用します。
タイプ	証明書コンテナー。インスタンスは、トラストストア、Java キーストア、および PKCS#12 キーストアからの証明書を認識します。
有効開始日	インスタンスは、証明書の有効開始日を自動的にこのフィールドに追加します。証明書を X.509 証明書レコードに添付して、このフィールドに入力します。
有効期限	インスタンスは、証明書の有効期限をこのフィールドに自動的に追加します。証明書を X.509 証明書レコードに添付して、このフィールドに入力します。
数日中に期限切れ	有効期限までの計算された日数。
簡単な説明	証明書の説明。
問題	インスタンスは、証明書発行者をこのフィールドに自動的に追加します。証明書を X.509 証明書レコードに添付して、このフィールドに入力します。
件名	インスタンスは、このフィールドの対象となる証明書を自動的に追加します。証明書を X.509 証明書レコードに添付して、このフィールドに入力します。
PEM 証明書	X509 証明書の値を入力します。

次のタスク
[ストア/証明書を検証] をクリックして、トラストストアと証明書をテストします。

不足している **SAML** の証明書を置き換える

証明書モジュールに空白のページが表示される場合は、SAML 2.0 証明書レコードが削除されています。証明書レコードを手動で作成することで、欠落している証明書を置き換えることができます。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > システム定義 > 証明書.
2. SAML 2.0 という新しいレコードを作成します。

   重要: この名前を使用する必要があります。この要件は、複数プロバイダーのシングルサインオン (SSO) を使用していない場合にのみ当てはまります。

3. クリック **SAML 2** シングルサインオン > 証明書.
4. *[PEM* 証明書] フィールドで、IdP のメタデータから ds:X509Certificate 要素の値を入力します。
5. [保存] をクリックします。

SAML 統合のテスト

他のすべてのセットアップタスクを完了した後、SAML 統合をテストします。

始める前に
必要なロール：admin

手順

1. admin ロールを持つユーザーとしてインスタンスにログインします。
2. 移動先 **SAML 2** シングルサインオン > プロパティ.
3. [外部認証を有効にします] プロパティで、[はい] を選択します。

   注:

   外部認証を有効にするには、すべてのユーザーが SAML 2.0 Single Sign-on を使用する必要があります。誰かが未認証の状態でアプリケーションにアクセスしようとすると、インスタンスは自動的に (IdP に) 認証要求を送信し、ユーザーを [SAML IdP 認証 (SAM IdP Authentication)] ページにリダイレクトします。

4. [保存] をクリックします。
5. インスタンスからログアウトします。
6. インスタンス URL を参照します。
   統合が正常に機能している場合、IdP はユーザーの認証情報を要求します。

関連トピック

マルチ SSO (SAML 2.0) のエラーと修正

マルチ SSO (SAML 2.0) のエラーと修正

マルチ SSO (SAML 2.0) のセットアップと構成に関する一般的なエラーと関連する修正のリスト。

マルチ **SSO (SAML 2.0)** セットアップ中のエラー

インスタンスログのエラー	テスト接続メッセージ	**SAML** プロパティ	診断	修正
NotAfter：<Thu Jun 05 22:57:44 PDT 2014>	IDP x509 証明書が存在し、有効かつアクティブであることを確認してください。	適用外	現在の証明書または SAML アサーションの有効期限が切れました。	• SNC クロックを SAML IdP サーバークロックと同期します。 • SAML 2.0 証明書レコードを更新します。
• SAML 2.0 証明書を特定できません。 • ServiceNow インスタンスに保存されているデジタル署名が見つかりませんでした。(Could not find a digital signature stored in the ServiceNow instance.)	IDP x509 証明書が存在し、有効かつアクティブであることを確認してください。	PEM 形式の文字列を [PEM 証明書] フィールドに入力する必要があります。	SAML 証明書が存在しません。非アクティブの可能性があります。	正しい PEM 形式の証明書がインスタンスにアップロードされていることを確認します。
証明書が一致しません。(Certificates do not match.) 想定：<certStr>、実際：<inboundCert> (Expect: <certStr>, actual: <inboundCert>.)	IDP x509 証明書が存在し、有効かつアクティブであることを確認してください。	適用外	SNC で利用可能な証明書がアサーションの証明書と一致しません。次のような原因が考えられます。 • IdP では証明書が更新されているが、ServiceNow インスタンスで更新されていない。 • 証明書の形式が正しくない。	SAML 2.0 証明書レコードの PEM 形式の文字列が、ユーザー IdP の SAMLResponse の X509 証明書と一致していることを確認します。
証明書の有効性の確認に失敗しました。(Failure to check the validity of the certificate.)	IDP x509 証明書が存在し、有効かつアクティブであることを確認してください。	適用外	現在の証明書の有効期限が切れている可能性があります。	SAML 2.0 証明書レコードを更新します。
署名プロファイルの検証に失敗しました。(Failure to validate signature profile.)	IDP x509 証明書が存在し、有効かつ	適用外	アサーションが別の証明書で署名されている可能性があります。	IdP に SNC インスタンスと同じ証明書があるかどうかを確認します。

マルチ **SSO (SAML 2.0)** セットアップ中のエラー (続く)

インスタンスログのエラー	テスト接続メッセージ	**SAML** プロパティ	診断	修正
	アクティブであることを確認してください。			
SubjectConfirmationData の InResponseTo 属性が一致しません。(InResponseTo attribute in SubjectConfirmationData mismatch.) 想定：<inResponseTo>、実際：<inResponseTo> (Expect: <inResponseTo>, actual: <inResponseTo>.)	SubjectConfirmation の検証に失敗しました。	適用外	このエラーは、次のいずれかの状況が発生した場合に表示されます。 • IdP が別の SAMLRequest の SAMLResponse を返す場合 • ユーザーがインスタンス URL ではなく SAMLRequest を使用して URL をブックマークする場合 • null 値が想定される場合、インスタンスに複数のノードがあると、応答が別のノードに送信される可能性があります。	IdP admin は、想定される SAMLReponse が返されていることを確認する必要があります。この状況は、ロードバランサーまたはインフラストラクチャの問題である可能性があります。
SessionIndex 値が見つかりません：<メッセージ>... (SessionIndex value not found: <message>...)	SessionIndex が有効ではありません。(SessionIndex not valid.)	適用外	SNC インスタンスには SessionIndex が必要です。IdP はこれを SAML 応答で返して正常に認証します。	IdP admin は、SessionIndex が SAMLResponse で定義されていることを確認する必要があります。
有効な SubjectConfirmation が見つかりません。	SubjectConfirmation の検証に失敗しました。	適用外	IdP のエラーにより、条件が欠落している可能性があります。 応答の StatusCode には、想定される成功ではなくレスポンダーが含まれます。	SAMLResponse を確認して、SAMLResponse に条件が含まれているかどうかを判断します。 有効な件名確認データの有効期限が切れているか、対象者に対して適切でない可能性があります。
アサーション対象者が一致しません。(Assertion audience	[対象者 URI] フィールドが正し	SAML2 トークンを受け入れる対象	SNC インスタンスで構成された対象者 URI は、IdP の値と一致する必要があります。	ログの SAMLResponse で <saml2:Audience> を探し、値がインスタン

マルチ **SSO (SAML 2.0)** セットアップ中のエラー (続く)

インスタンスログのエラー	テスト接続メッセージ	**SAML** プロパティ	診断	修正
mismatch.) 想定：<propAudience>、実際：<audienceUri> (Expect: <propAudience>, actual: <audienceUri>.) または AudienceRestriction の検証に失敗しました。一致する対象者が見つかりません。(No matching audience found.)	く設定されていることを確認します (Ensure that the 'Audience URI' field is set correctly)	者 URI (通常、インスタンス URI。たとえば https://demo.service-now.com)。		スの値と一致することを確認します。
アサーションの発行者が無効です。想定：<value on instance>、実際：<value returned by IdP> (Expect: <value on instance>, actual: <value returned by IdP>)	アサーションの発行者が無効です。	ユーザー情報と一緒に SAML2 セキュリティトークンを発行する ID プロバイダの URL。	IdP エンティティ ID (発行者) が SNC インスタンスで定義された値と一致しません。	• IdP または SP の設定が不適切でないか確認します。 • SAML プロパティ (ユーザー情報と一緒に SAML2 セキュリティトークンを発行する ID プロバイダの URL) が正しく設定されているか確認します。
件名は将来有効になります。(Subject is valid in the future.) 現在：<now>、NotBefore：<notBefore> (Now: <now>, NotBefore: <notBefore>) または 件名の有効期限が切れました。(Subject is expired.) 現在：<now>、NotOnOrAfter：<notOnOrAfter> (Now: <now>, NotOnOrAfter: <notOnOrAfter>)	件名の検証確認に失敗しました。(Subject validation confirmation failed.)	notBefore 制約の前、または notOnOrAfter 制約の後に、まだ有効であるとみなされる時間の秒数。	IdP クロックが SP クロックと同期していません。	SAML プロパティ glide.authenticate.sso.saml2.clockske をより大きな値に更新します。デフォルトは 180 秒です。場合によっては 300 以上の設定が必要になります。IdP サーバーの時間も確認する必要があります。

マルチ **SSO (SAML 2.0)** セットアップ中のエラー (続く)

インスタンスログのエラー	テスト接続メッセージ	**SAML** プロパティ	診断	修正
アサーションは将来有効になります。現在：<now>、notBefore：<notBefore> (Assertion is valid in the future, now: <now>, notBefore: <notBefore>) または アサーションの有効期限が切れました。現在：<now>、notOnOrAfter：<notOnOrAfter>	アサーションが無効です。	notBefore 制約の前、または notOnOrAfter 制約の後に、まだ有効であるとみなされる時間の秒数。	IdP クロックが SP クロックと同期していません	SAML プロパティをより大きな値に更新します。デフォルトは 60 秒です。場合によっては 300 以上の設定が必要になります。IdP サーバーの時間も確認する必要があります。

一般的なログインおよび **IdP** エラー

エラーまたは症状	診断	修正
高セキュリティがアクティブな場合、ログイン要求はシステムと IdP の間で無限ループを生成します。	• 通常、URL エンドポイントはエラーページまたはログアウトページになります。 • IdP ホスト名をプロパティ値に追加せずに *glide.security.url.whitelist* を定義すると、logout_redirect.do によってこのループが作成される可能性があります。 **注:** このプロパティの詳細については、「インスタンスセキュリティ強化設定」の「URL 許可リストのチェックを強制する (セキュリティセンター 1.3、1.5、および 2.0 で更新)」を参照してください。	失敗した認証要求をこの URL にリダイレクトするには、システムプロパティ glide.authenticate.failed_redirect を設定 (または作成) します。
ユーザーまたは要求の認証に使用されるトークンが、署名アルゴリズム http://www.w3.org/2001/04/xmldsig-more#rsa-sha256 で署名されています。これは、想定される署名アルゴ	イベントの詳細については、[アラートコンテキスト (Alert Context)] タブを確認してください。	[証明書利用者信頼設定 (Relying Party Trust configuration)] ダイアログの [詳細] タブに移動し、アルゴリズムが SHA-256 ではなく

一般的なログインおよび **IdP** エラー (続く)

エラーまたは症状	診断	修正
リズム http://www.w3.org/2000/09/xmldsig#rsa-sha1 とは異なります。		SHA-1 に設定されていることを確認します。
エラーメッセージ urn:oasis:names:tc:SAML:2.0:sta がシステムログ (syslog) テーブルに表示されます。	IdP (ADFS など) の応答が oasis:names:tc:SAML:2.0:status:Requester のステータスの場合は、送信された要求に関する問題が原因で IdP がログインを拒否したことを意味します。ほとんどの場合、IdP から受信した SAML 応答にはエラーの詳細が記載されていません。	IDP に送信された SAML 要求を確認し、IDP アドミニストレーターと協力してインスタンスの SAML 設定を更新して、エラーを回避してください。ログイン失敗の理由については、IDP プロバイダーにお問い合わせください。

シングルサインオン **(SSO)** ログインのリダイレクト

SSO を有効にすると、ユーザーを特定のページにリダイレクトしたり、ユーザーにローカルログインを誘導したりすることができます。

たとえば、ユーザーが https://customerX.service-now.com にアクセスしようとすると、デフォルトのログインページの代わりに社内ポータルが表示されます。または、ユーザーがアプリケーションからログアウトするときに、ブラウザーはユーザーを特定の内部ページにリダイレクトできます。インスタンス内でリダイレクトプロパティを設定して、ユーザーにデフォルトのログインページではなく SSO ログインページが確実に表示されるようにできます。

注: 次のプロパティは SSO を強制しません。login.do ページには引き続きアクセスでき、ユーザーは、ローカルパスワードを設定していればシステムにログインできます。

リダイレクトのプロパティ

ユーザーがログアウトした場合、または SSO を使用したサインオンに失敗した場合に、ユーザーが次に移動する場所 (メインポータルページや SSO ログイン情報を含むナレッジベース記事) を定義できます。URL を指定するには、次のプロパティを使用します。これらのプロパティのいずれかがインスタンスに存在しない場合は、プロパティを作成できます。

glide.authenticate.failed_requirement_redirect

ユーザーがプライベートページにアクセスしようとしたときに (インシデントを表示しようとした場合など)、SSO 認証情報を提供しない場合にリダイレクトされる URL。通常、このプロパティは顧客のログインポータル (http://portal.companya.com/ など) に設定されます。

glide.authenticate.failed_redirect

SSO 試行の失敗後にユーザーをリダイレクトする URL。エラーについての説明と役に立つリンクがある公開ナレッジ記事にリダイレクトできます (例：http://portal.companya.com/error)。

glide.authenticate.external.logout_redirect

ログアウト後にユーザーをリダイレクトする URL。通常はシングルサインオンログインを有効にしたポータルに戻ります (http://portal.companya.com/logout など)。

glide.authentication.external.disable_local_login

true に設定すると、メインログインページの SSO 認証情報が必要になります。デフォルト：false このプロパティは、*glide.authenticate.failed_requirement_redirect* プロパティと組み合わせて使用する必要があります。

次の表は、インストレーションイグジットの戻り値、プロパティ、および予想される動作の関係を示しています。

SSO のみを使用したログインの強制

値を返す	プロパティ	動作
failed_missing_requirement	*glide.authenticate.failed_requirement_redirect*	この値が返された場合に認証情報がセッションとを示しています。ログセッションはプロパテ URL にリダイレクトさ常、ログインが要求さ集される SSO プロバイす。
failed_authentication	*glide.authenticate.failed_redirect*	この値が返される場合にSSO 認証情報が認証にザーが存在しないか、アウトされていることインに失敗し、セッシで指定された URL にリます。これは通常、ログれ、認証情報が収集されイダーの URL です。
<user_id>	適用外	*<user_id>* で指定されザーにログインが許可すす。この値は、SSO プ *glide.authenticat* (「受信ヘッダーと照合用するインスタンスの (the instance's field against the incoming 義されたフィールド名

ローカルログインの制限

セキュリティ上の予防措置として、ローカルログインを防ぐために、リダイレクトプロパティを使用する以上のことをする必要があります。ユーザーがローカルにログインせず、常に内部のシングルサインオンシステムによって認証される場合は、インスタンスにインポートする各ユーザーにランダムなパスワードを割り当てる必要があります。ランダムなパスワードは、ユーザーのインポート時に最も簡単に設定できます。ユーザーデータがインポートセットを介してシステムにインポートされる場合は、次のコードを使用して onBefore 変換スクリプトを作成できます。

```
var r  = new Packages. java. util. Random ( ) ;

 var str1  = Packages. java. lang. Long. toString (Packages. java. lang.
 Math. abs (r. nextLong ( ) ) , 36 ) ; var str2  = Packages. java. lang.
 Long. toString (Packages. java. lang. Math. abs (r. nextLong ( ) ) , 36
 ) ;

 var newPass  = str1  + str2 ;

target. user_password = newPass ;

 //password now set to a random string like this:
 //qvm81zdrn7cwwylpvw94eebk
```

SAML 統合によるインスタンスのクローン作成

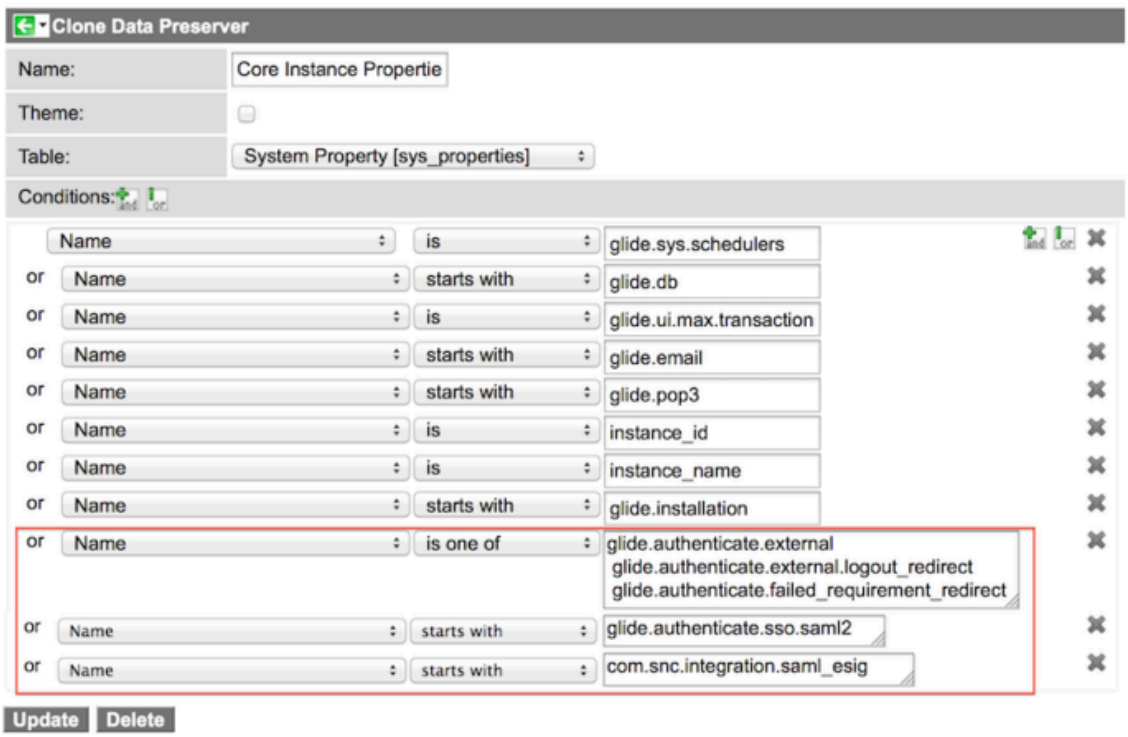
SAML 統合によりインスタンスのクローンを作成します。SAML 2.0 を使用するインスタンスのクローンを作成する前に、ターゲットインスタンスの SAML SSO 関連の設定を保存します。そうしないとターゲットインスタンスがアクセスできなくなる場合があります。

始める前に

必要なロール：admin

手順

1. ソースインスタンスで、次の場所に移動します: システムクローン > データを保存 > コアインスタンスのプロパティ.
2. 条件を使用して、次の SAML SSO 関連プロパティが保持されていることを確認します。
   - glide.authenticate
   - glide.security
   - glide.entry
   - glide.script
   - glide.session
   - glide.saml2
   - com.glide.communications
   - com.snc.integration.saml_esig



注: クローンを作成するときには、証明書がターゲットインスタンスに引き継がれるように添付ファイルを含めます。また、[テーマ] チェックボックスをオフにして、インスタンステーマを保持するかどうかに関係なく、これらのプロパティが保持されるようにします。

3. ソースインスタンスで、次の場所に移動します: システムクローン > データを保存 SAML/SSO/マルチ SSO に関連するsys_userで、sys_certificate および SAML ユーザーの SAML 証明書を保持します。

   必要な場合は、XML にエクスポートしてから、ターゲットに手動でインポートします。

   ⚠ 警告: あるシステムから別のシステムに SAML/SSO/マルチ SSO セットアップのクローンを作成しようとしないでください。ほとんどの場合、SAML/SSO またはマルチ SSO 設定の転送は、ID プロバイダーで構成する必要があるため機能しません。作業セットアップを上書きすると、ターゲットインスタンスが認証に失敗するため、ターゲットインスタンスにアクセスできなくなります。また、マルチ SSO プロバイダーレコードの sys_id を変更しないでください。変更すると、ユーザーは cookie を強制的にフラッシュします。クローンする際の注意事項の詳細については、「インスタンスをクローンする前のチェックリスト (Checklist before cloning an instance)」を参照してください。

4. マルチ SSO テーブルの sso_properties、digest_properties、および saml2_update1_properties を除外します。
5. 各インスタンスで SAML/SSO/マルチ SSO レコードを手動で作成し、ID プロバイダーでもレコードを設定します。
6. ターゲットインスタンスの sys_user (LDAP でも SAML でもない) レコードで、ソースインスタンスに存在しない sys_id を使用して、LOCAL アドミンアカウントを手動で作成します。
7. [更新] をクリックします。

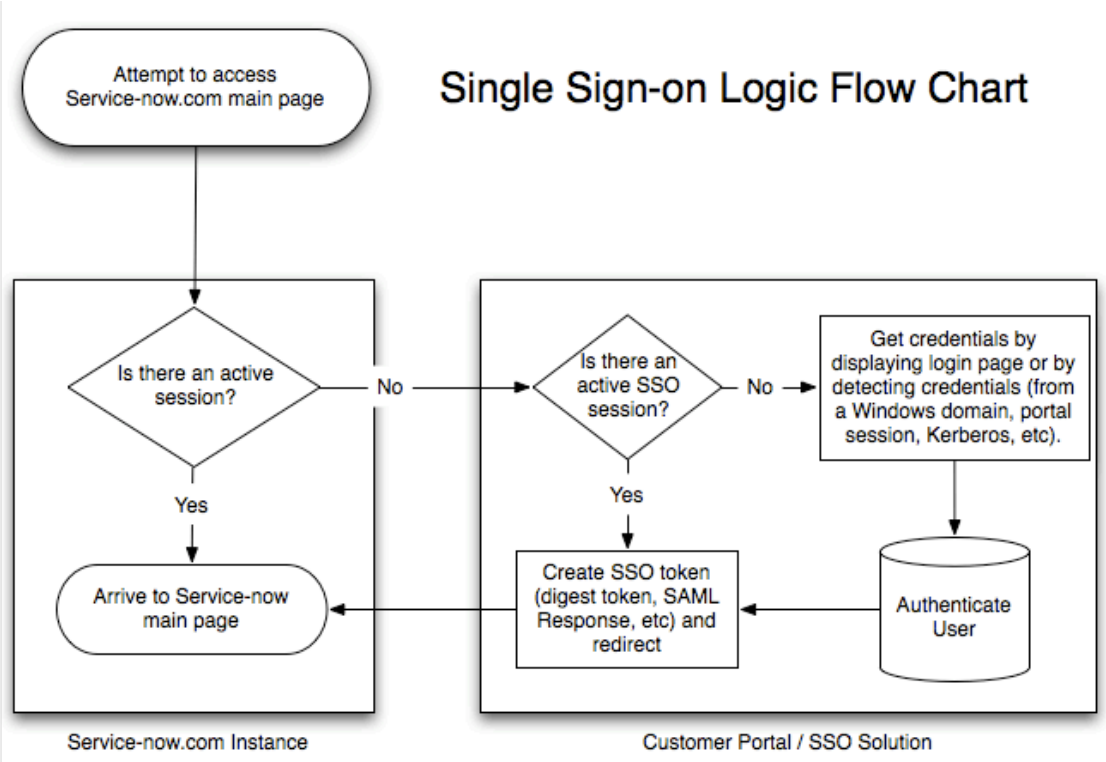
SAML 2.0 のコンセプト

次の SAML のコンセプトについて説明します。

一般的な SAML プロセスフロー (図)

一般的な SSO ロジックフローには、アクティブなセッションの検索、ユーザー認証情報の確認、必要なトークンの作成が含まれます。

SSO 標準



ログイン (AuthnRequest) プロセスフロー

SAML 2.0 は、Web ブラウザで ID プロバイダー (IdP)、サービスプロバイダー (SP)、プリンシパル (ユーザー) の間で情報を交換する Web ブラウザ SSO プロファイルを指定します。

ID プロバイダーは、SAML 認証サービスを提供する任意の SSO サービス (SSOCircle など) にすることができます。サービスプロバイダーは常にインスタンスです。メッセージフローは、サービスプロバイダーの保護されたリソースの要求から始まります。

SP でのターゲットリソースの要求

プリンシパルは、サービスプロバイダーでターゲットリソースを要求します。

https://instance.service-now.com/

インスタンスは要求を確認して、SAMLRequest および RelayState URL パラメーターが存在するかどうかを把握します。存在する場合、ユーザーは IdP で既に検証済みであるため、ステップ 2 ～ 6 をスキップできます。

ID プロバイダーに AuthnRequest を発行

インスタンスは、SAMLRequest 値を使用して IdP に送信される AuthnRequest を作成します。インスタンスは、さらに RelayState URL パラメーター値も作成して送信します。

RelayState トークンは、サービスプロバイダーで維持されるステータス情報への不透過型参照です。SAMLRequest パラメーターの値は、<samlp:AuthnRequest> 要素を縮小した base64 エンコード値です。

```
<samlp:AuthnRequest    xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
   xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"    ID="identifier_1"
   Version="2.0"    IssueInstant="2004-12-05T09:21:59Z"
 AssertionConsumerServiceIndex="0"><saml:Issuer>https://
sp.example.com/SAML2</saml:Issuer><samlp:NameIDPolicy      AllowCreate="true"
 Format="urn:oasis:names:tc:SAML:2.0:nameid-format:transient"/></samlp:AuthnRequest>
```

その後、統合は <samlp:AuthnRequest> 要素を URL エンコードし、それを SAMLRequest URL パラメーターとして送信します。

SSO サービスは、URL デコード、base64 デコード、拡張の順に <samlp:AuthnRequest> 要素を処理します。次に、セキュリティチェックを実行します。ユーザーが有効なセキュリティコンテキストを持っていない場合、IdP はログイン認証情報を要求してユーザーを識別します。ユーザーがすでにログインしている場合、IdP は単に SAMLResponse<tt> および <tt>RelayState URL パラメーターで応答します (ステップ 3 を参照)。

SAMLResponse と RelayState での応答

必要なログイン認証情報を収集した後、SSO サービスは要求を検証し、XHTML フォームを含むドキュメントで応答します。

```
<formmethod="post"action="https://instance.service-now.com/navpage.do" ...><input
 type="hidden" name="SAMLResponse" value="response ..." /><input type="hidden"
 name="RelayState" value="token ..." />
   ...
   <input type="submit" value="Submit" /></form>
```

RelayState パラメーターの値はこのステップから取得されます。SAMLResponse パラメーターの値は、次の <samlp:Response> 要素の base64 エンコードです。

```
<samlp:Responsexmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
 ID="s2cdc74f37f923e26fe1aeec42b70a93d24230334f"
```

```
InResponseTo="90AA6073F01567BFB0DF194F596314E2"
 Version="2.0"  IssueInstant="2010-04-29T23:21:51Z"
Destination="https://dloomac.service-now.com/navpage.do"><saml:Issuer
xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion">http://
idp.ssocircle.com</saml:Issuer><samlp:Status
xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"><samlp:StatusCode
xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
Value="urn:oasis:names:tc:SAML:2.0:status:Success"></samlp:StatusCode></samlp:Status>
<saml:Assertion xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
   ID="s23e536bfc51b8487d4d3299dec162d9c2e338823b"
IssueInstant="2010-04-29T23:21:51Z"
Version="2.0"><saml:Issuer>http://idp.ssocircle.com</saml:Issuer><Signature
xmlns="http://www.w3.org/2000/09/xmldsig#">

...
    </Signature><saml:Subject><saml:NameID
Format="urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress"
        NameQualifier="http://idp.ssocircle.com"
SPNameQualifier="https://
dloomac.service-now.com/navpage.do">david.loo@service-now.com</saml:NameID><saml:Su
bjectConfirmation
Method="urn:oasis:names:tc:SAML:2.0:cm:bearer"><saml:SubjectConfirmationData
          InResponseTo="90AA6073F01567BFB0DF194F596314E2"
          NotOnOrAfter="2010-04-29T23:31:51Z"
Recipient="https://dloomac.service-now.com/navpage.do" /
></saml:SubjectConfirmation></saml:Subject><saml:Conditions
NotBefore="2010-04-29T23:11:51Z"
NotOnOrAfter="2010-04-29T23:31:51Z"><saml:AudienceRestriction><saml:Audience>http
s://
dloomac.service-now.com</saml:Audience></saml:AudienceRestriction></saml:Conditions>
<saml:AuthnStatement AuthnInstant="2010-04-29T23:21:51Z"
SessionIndex="s2dbf89ab99001e0e8cdaed67266d9d4b21b968a04"><saml:AuthnContext><sa
ml:AuthnContextClassRef>urn:oasis:names:tc:SAML:2.0:ac:classes:PasswordProtectedTranspo
rt</saml:AuthnContextClassRef></saml:AuthnContext></saml:AuthnStatement></saml:Asse
rtion></samlp:Response>
```

SAMLResponse の検証

SAMLResponse の値は base64 でデコードされて拡張され、ステップ 3 の XML ドキュメントが表示されます。ログインスクリプトは //Subject/NameID 要素から XML 値を抽出し、それを使用してユーザーテーブルで既存のユーザーを検索します。

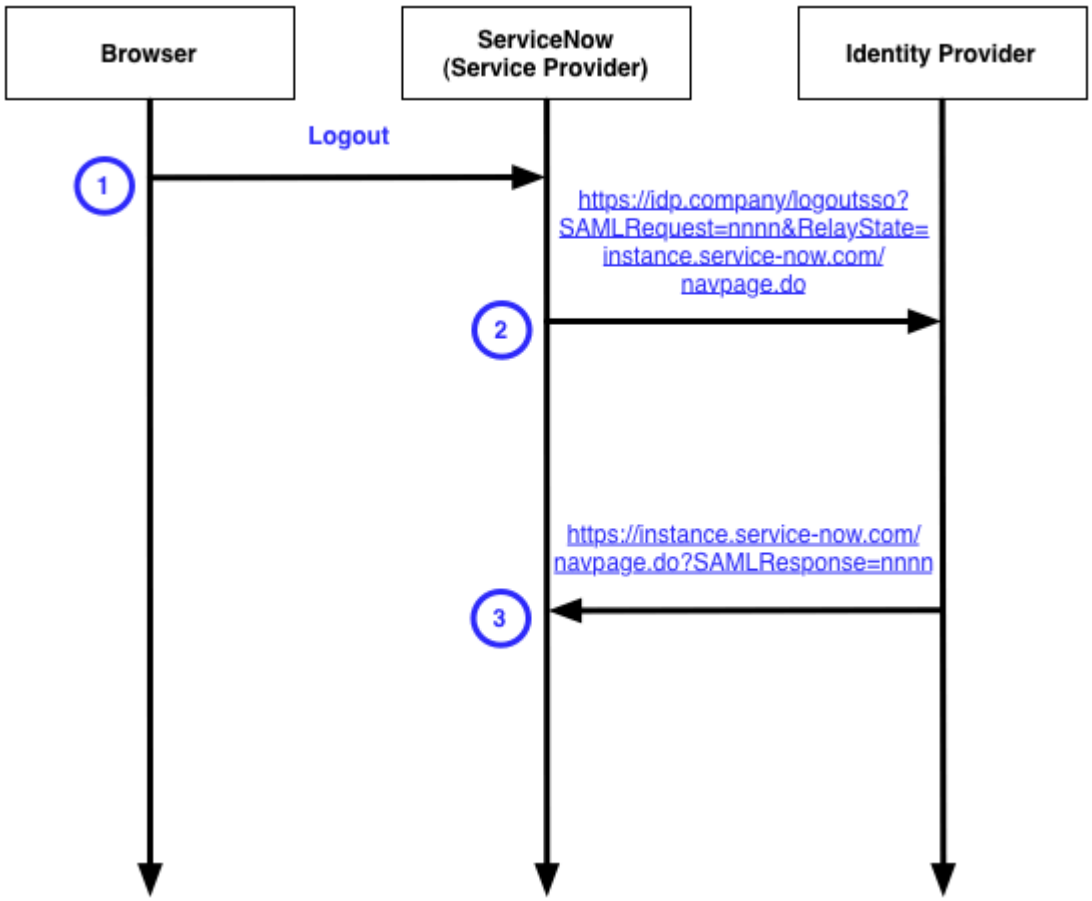
ログインスクリプトは、//AuthnStatement/@SessionIndex 要素からセッション ID を抽出し、LogoutRequest 用に格納します。

ログアウト (LogoutRequest) プロセスフロー

ログアウト時に、インスタンスは SAML 2.0 *LogoutRequest* サービス コールを IdP に発行します。

このサービスは、ユーザーをログアウトさせた後、指定されたログアウト URL にリダイレクトします。

SAML 2 ログアウト



ユーザーが [ログアウト] ボタンをクリック

ユーザーが [ログアウト] ボタンをクリックすると、インスタンスはログアウトスクリプトを実行します。

LogoutRequest の発行

ログアウトスクリプトは SAML 2.0 LogoutRequest を作成し、それを IdP で事前設定された SingleLogoutRequest SAML 2.0 サービスに投稿します。IdP は要求を縮小し、それを base64 エンコードします。LogoutRequest の例は次のようになります。

```
<saml2p:LogoutRequestxmlns:saml2p="urn:oasis:names:tc:SAML:2.0:protocol"
 ID="21B78E9C6C8ECF16F01E4A0F15AB2D46"  IssueInstant="2010-04-28T21:36:11.230Z"
 Version="2.0"><saml2:Issuer
 xmlns:saml2="urn:oasis:names:tc:SAML:2.0:assertion">https://dloomac.service-now.com
 </saml2:Issuer><saml2:NameID xmlns:saml2="urn:oasis:names:tc:SAML:2.0:assertion"
        Format="urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress"
        NameQualifier="http://idp.ssocircle.com"
 SPNameQualifier="https://
dloomac.service-now.com/navpage.do">david.loo@service-now.com</saml2:NameID><saml2
p:SessionIndex>s211b2f811485b2a1d2cc4db2b271933c286771104
 </saml2p:SessionIndex></saml2p:LogoutRequest>
```

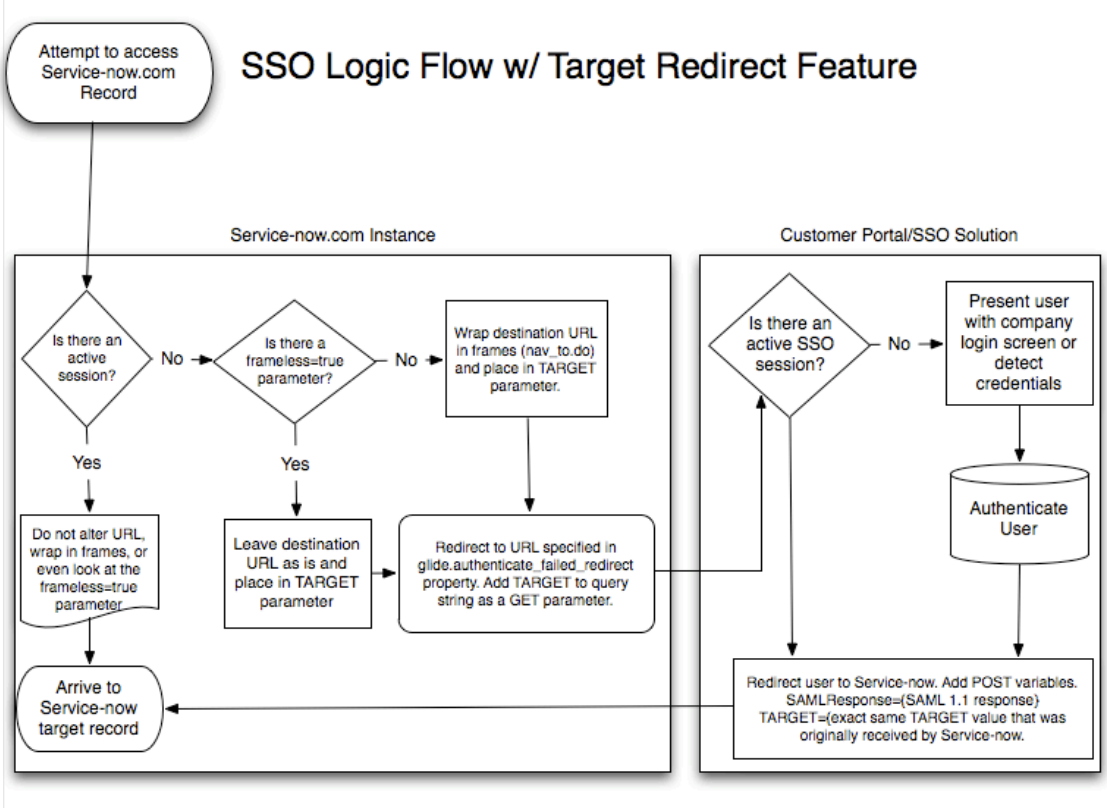
ユーザーのログアウト

ユーザーが IdP からログアウトします。IdP はリダイレクトされてインスタンスに戻りますが、ユーザーがログインしていないため、再度 IdP にリダイレクトされます。

SSO プロバイダーの URL 情報

SSO セッションを必要とするインスタンスへの URL リンクによってログインが阻止される際、SSO プロバイダーに対して参照 URL を指定し、認証後に URL をインスタンスに戻して正しいリソースにリンクできるようにする必要がある場合があります。

SSO ターゲットリダイレクト



インストレーションイグジットの戻り値は、プロパティで定義された URL の代わりに、または URL に加えて渡すように拡張されました。通常は、ユーザー名または事前定義された文字列値を返して、SSO セッションを許可するか阻止するかを制御します。次の例は、URL を渡す拡張動作を示しています。

```
return
 "failed_missing_requirement:%26amp;TARGET=https://
instance.service-now.com/nav_to.do?uri=incident.do?sys_id=12345";
```

上記の例では、URL https://instance.service-now.com/nav_to.do?uri=incident.do?sys_id=12345 を、TARGET という名前の URL パラメーターの形式で SSO プロバイダーに渡します。

注: SSO プロバイダーは、ユーザー認証情報が収集されて認証が渡されると、TARGET パラメーターのその情報を使用してインスタンスにリダイレクトすると想定されます。

コロン (:) は 2 つの戻り値を区切り、エンコードされた & (%26amp;) はプロパティ *glide.authenticate.failed_missing_requirement* と TARGET パラメーターで定義された URL を連結します。

マルチプロバイダー SSO を使用した SAML 2.0 構成

マルチプロバイダー SSO 機能から SAML 2.0 SSO 構成を作成または更新できます。

始める前に
必要なロール：admin

このタスクについて

注: Jakarta リリースから新たに、IdP 構成をアクティブ化する前にテスト接続機能を使用して構成を検証することが必要になりました。更新機能を使用して構成データを保存することはできますが、テスト接続が成功しないとアクティブな構成になりません。

手順

1. 移動先 すべて > マルチプロバイダー **SSO** > **ID** プロバイダー.
2. 次のいずれかのオプションを実行します。
   - 構成を更新するには、SSO 設定レコードをクリックします。
   - 新しい構成を作成するには、次をクリックします 新規 > **SAML**.
3. 次のいずれかの方法で IdP 情報を入力します。

オプション	説明
メタデータ記述子 URL を使用	[URL] チェックボックスをクリックし、使用している IdP の URL を入力します。
メタデータ記述子 XML ファイルを使用	[XML] チェックボックスをクリックし、使用している IdP から生成された XML データを貼り付けます。
メタデータを手動入力	ポップアップウィンドウを閉じて、プロパティフィールドに手動でデータを入力します。

注: [ID プロバイダー] フォームのすべての必須フィールドに入力する必要があります。

[マルチプロバイダー **SSO**] フィールド

プロパティ	必須	説明
名前	あり	IdP の名前を入力します。この IdP は自動リダイレクト Sys ID です。
有効	はい	IdP を認証に使用するには、[有効] を true に設定する必要があります。 注: このプロパティを設定するオプションは、テスト接続が成功した後にのみ表示されます。
デフォルト	なし	自動リダイレクト IdP (旧称「プライマリ IdP」) は、ユーザーをベースインスタンス URL に自動的にリダイレクトします。このプロパティは、この IdP 構成をデフォルトとして設定します。
自動リダイレクト IdP	なし	この IdP 構成を自動リダイレクト IdP として設定します。 注: 新しい自動リダイレクト IdP 構成をアクティブにすると、 `glide_sso_id` cookie が新しい自動リダイレクト IdP で更新されます。自動的に有効になる `glide.authenticate.sso.update.idp.cookie` システムプロパティは、この機能を制御します。
ID プロバイダー URL	あり	IdP への URL を入力します。各 IdP URL は一意である必要があります。

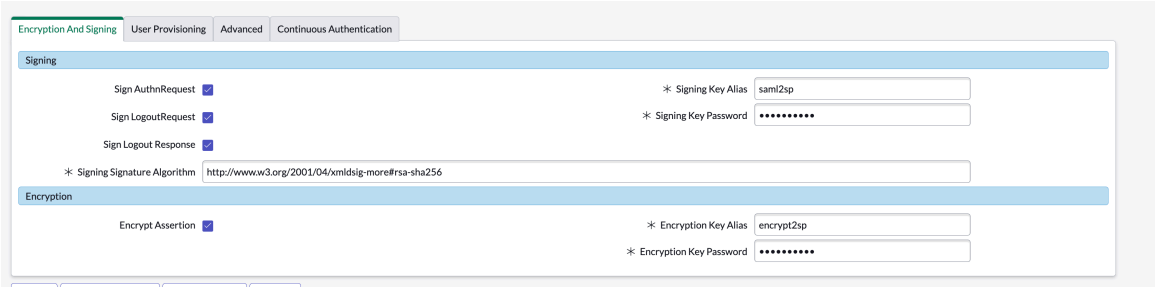
自動翻訳

プロパティ	必須	説明
ID プロバイダーの AuthnRequest	あり	SingleSignOnService 要素から取得した HTTP リダイレクトバインディングへの URL を入力します。
ID プロバイダーの SingleLogoutRequest	なし	SingleLogoutService 要素から取得した URL を入力します。
ServiceNow のホームページ	あり	IdP が認証するインスタンスの URL (ログインページを含む) を入力します。例：https://yourinstance.service-now.com/navpage.do
エンティティ ID/発行者	あり	IdP が認証するインスタンスのベース URL (ログインページを除く) を入力します。例：https://yourinstance.service-now.com/
対象者 URI	あり	IdP が認証するインスタンスのベース URL (ログインページを除く) を入力します。例：https://yourinstance.service-now.com/
NameID ポリシー	あり	統合で使用される NameIDFormat 要素の値を入力します。
外部のログアウトのリダイレクト	なし	ログアウト後に統合がリダイレクトする URL を入力します。
要件のリダイレクトに失敗	なし	失敗した認証要求をリダイレクトするための URL を入力します。デフォルトでは、これは IdP で設定されたエラーページまたはログアウトページの URL エンドポイントです。この値は glide.authenticate.failed_requirement_redirect フィールドに入力できます。
クライアントタイプ	いいえ	クライアントのタイプに基づいて、クライアントタイプを選択します。オプション：**iFrame** 埋め込み。 **注:** 構成に [クライアントタイプ] フィールドが必要な場合は、フォームを編集してフィールドを追加できます。詳細については、「OAuth および SSO レコードのクライアントタイプの構成」を参照してください。

4. オプション: [暗号化と署名] タブ

注:

- 暗号化と署名には独自の証明書を使用することをお勧めします。
- **FIPS** 承認済みモードでは、暗号化と署名に異なる証明書が必要です。
- 証明書の使用中は、次のシステムプロパティを証明書 (x.509 証明書) の sys_id に更新してください。
  - 署名 (glide.authenticate.sso.saml2.keystore)
  - 暗号化 (glide.authenticate.sso.saml2.encryption.keystore)
- ID プロバイダーレコードの署名キーストアと暗号化キーストアのキーエイリアスとキーパスワードを必ず更新し、メタデータを生成してください ([メタデータを生成] を選択)。
- 生成されたメタデータ (XML) に存在する署名証明書と暗号化証明書を ID プロバイダーにアップロードします。



[暗号化と署名] フィールド

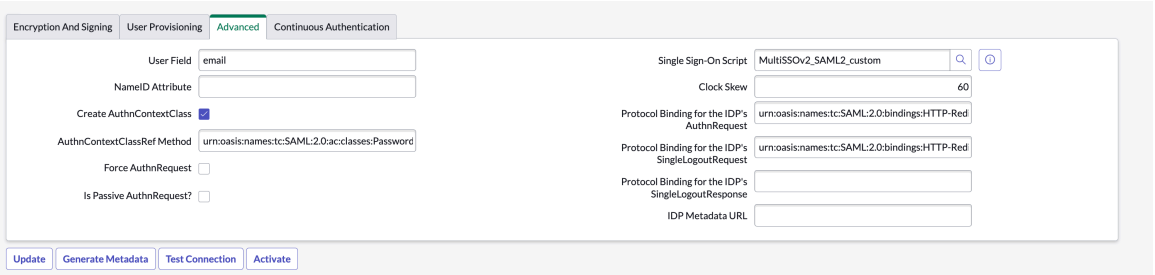
プロパティ	説明
署名キーエイリアス	SAML 2.0 SP Keystore に保存されているキーエントリの署名エイリアスを入力します。
署名キーパスワード	SAML 2.0 SP Keystore に保存されているキーエントリの署名パスワードを入力します。
暗号化キーのエイリアス	SAML 2.0 SP Keystore に保存されているキーエントリの暗号化エイリアスを入力します。
暗号化キーのパスワード	SAML 2.0 SP Keystore に保存されているキーエントリの暗号化パスワードを入力します。
アサーションを暗号化	SAML 応答のアサーションを暗号化するには、このチェックボックスをオンにします。IDP 用に生成されたメタデータには、IDP が生成する SAML 応答のアサーションを暗号化するために使用する x509 証明書が埋め込まれています。
署名アルゴリズムの署名	eSignature 認証用の SAML 2.0 Identity Provider AuthnRequest Consumer を示す URL を入力します。
AuthnRequest に署名	このチェックボックスをオンにすると、IdP Single Sign-on サービスが署名済みの AuthnRequest を受信できるようになります。
LogoutRequest の署名	このチェックボックスをオンにすると、IdP Single Sign-on サービスが署名済みの LogoutRequest を受信できるようになります。
ログアウト応答に署名	このチェックボックスをオンにすると、IdP シングルサインオンサービスが署名済みのログアウト応答を受信できるようになります。

5. オプション: [ユーザープロビジョニング] タブ

[ユーザープロビジョニング] フィールド

プロパティ	説明
自動プロビジョニングユーザー	自動ユーザープロビジョニングを有効にします。IdP によって提供される情報に基づいてユーザーがインスタンスユーザーテーブルに存在しない場合は、ユーザーを作成します。
各ログイン時のユーザーレコードの更新	ユーザーが SAML を使用してログインするたびに、インスタンスユーザーテーブルのユーザー情報を IdP の情報で更新します。

6. オプション: [詳細] タブ



[詳細] フィールド

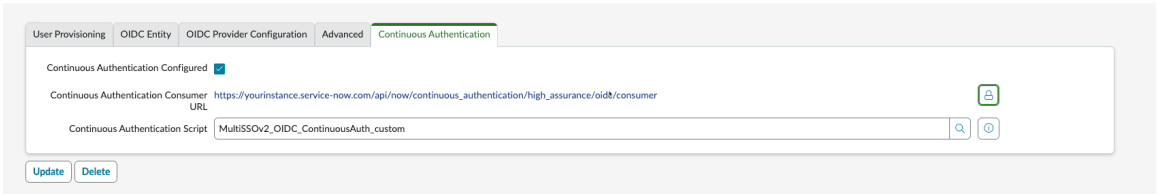
プロパティ	説明
ユーザーフィールド	ユーザーを識別するために IdP に必要な値を含むユーザーテーブルのフィールドに入力します。これは応答の一部としての一意の ID です。たとえば、ユーザー名、従業員 ID などがあります。システムユーザーテーブルで、この一意の ID がユーザーの詳細と照合されます。
NameID 属性	新しい NameID ポリシーを設定する場合を除き、このフィールドは空白のままにします。新しいポリシーを設定する場合、システムではログインするユーザーを識別するためのユーザーテーブルを使用する必要があります。ここで、システムは NameID トークンを、そのユーザーテーブルフィールドの名前と照合します。
AuthnContextClass の作成	[パスワードで保護されたトランスポート (Password Protected Transport.)] などの特定のコンテキストクラスを指定するには、このチェックボックスをオンにします。チェックボックスをオフにすると、IdP によって最も適切なコンテキストクラスが選択されます。
AuthnContextClassRef メソッド	IdP がユーザーを認証するために使用するログインメカニズムの URN を入力します。
AuthnRequest の強制	AuthnRequest を強制的に発生させるには、このチェックボックスをオンにします。
Passive AuthnRequest か	AuthnRequest がパッシブの場合は、このチェックボックスをオンにします。
シングルサインオンスクリプト	シングルサインオンスクリプトを選択します。デフォルトは *MultiSSOV2_SAML2_custom* です。
ログアウト応答に署名	このフィールドにログアウト応答の詳細を入力します。
クロックスキュー	SAMLResponse ノンスを構成する 2 つの属性間の秒数を入力します。デフォルトは 60 です。有効な SAMLResponse は、*notBefore* と *notOnOrAfter* の日付と時刻の値の範囲内である必要があります。SAMLResponse メッセージのサンプルについては、「Sample SAML 2 Response with the SubjectConfirmation and SubjectConfirmationData Elements (SubjectConfirmation および SubjectConfirmationData 要素を含む SAML 2 応答のサンプル)」と、「Sample SAML 2 Response with the AudienceRestrictions and Audience Elements (AudienceRestrictions および Audience 要素を含む SAML 2 の応答サンプル)」を参照してください。
IDP の SingleLogoutRequest のプロトコルバインディング	SingleLogoutService 要素の [バインディング] 属性に記載されているサポート対象の値を 1 つ入力します。

プロパティ	説明
IDP プロパティのインポート元のメタデータ URL	IdP プロパティはこの URL からインポートされます。設定すると、以前の証明書の有効期限が切れた場合に、IdP からの SAML 証明書の自動インポートが有効になります。 注: SAML2 Update 1 からマルチプロバイダー SSO にアップグレードする場合、または SSO 接続を手動で設定する場合、IdP メタデータ URL は自動的に入力されません。
要求	要求の一部としての一意の ID。ID にはユーザー名、従業員 ID などがあります。 注: 要求では、リダイレクトとポストバインディングの両方がサポートされています。このフィールドを設定するオプションは、テスト接続が成功した後にのみ表示されます。詳細については、「IdP 接続をテスト」を参照してください。
応答	応答の一部としての一意の ID。ID にはユーザー名、従業員 ID などがあります。 注: 応答では、リダイレクトとポストバインディングの両方がサポートされています。このフィールドを設定するオプションは、テスト接続が成功した後にのみ表示されます。詳細については、「IdP 接続をテスト」を参照してください。

7. オプション: [継続認証] タブで、次のフィールドを設定します。

注:

- [継続的認証] タブは、ライセンスが必要な **Zero Trust - Continuous Authentication** (com.snc.zero_trust_continuous_authentication) プラグインをインストールした場合にのみ表示されます。
- 継続的認証ポリシーを使用してテーブルまたはデータクラスへのアクセスを保護する場合は、「 継続認証 (CA)」を参照してください。



継続認証

フィールド	説明
継続認証が構成されました	チェックボックスをオンにして、構成をアクティブに設定します。
継続認証コンシューマー URL	ID プロバイダーからのコンシューマー URL を指定します。

自動翻訳

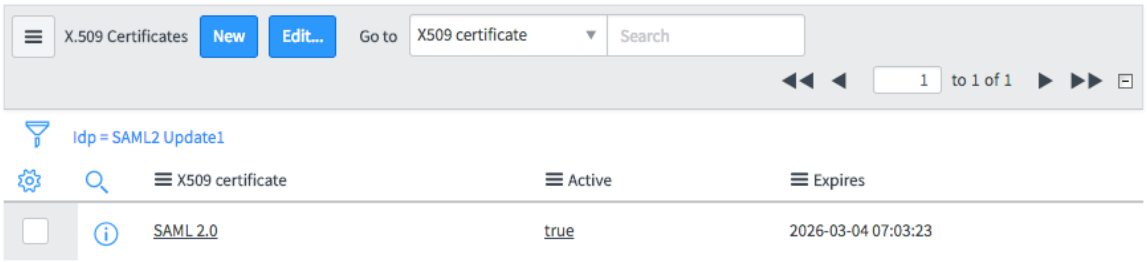
フィールド	説明
継続認証スクリプト	(Optional) ルックアップアイコンを選択して、プラットフォームから提供されたスクリプトを選択します。この構成では、SAML の場合: **MultiSSOv2_SAML2_ContinuousAuth_custom**

SAML の X.509 証明書

SAML 構成に必要な IdP 証明書を保存してアクティブ化します。

X.509 証明書は、SAML 構成で使用される IdP 証明書です。証明書をインストールした後、必要な数の証明書を追加できます。複数の証明書がある場合は、最初に見つかったアクティブな証明書が使用されます。**IDP** プロパティのインポート元のメタデータ **URL** フィールドに URL を設定すると、証明書が有効でなくなった場合に現在の有効な証明書のために IdP が自動的にポーリングされます。これにより証明書がインスタンスに追加され、アクティブな SAML 構成に使用されます。

注: ポーリングは、ネットワークの外部から IdP にアクセスできる場合に発生します。



SAML ガイド付きツアー

SAML チュートリアルを使用して、Single Sign-On 用の SAML を設定します。

SAML ガイド付きツアーは、ServiceNow® インスタンスの SSO のトレーニングと設定に役立ちます。アドミンはガイド付きツアーを選択して、インスタンスの SSO を設定するときに必要なアクションをすばやく把握できます。

ガイド付きツアーを使用する前に、Okta、Microsoft Azure、ADFS などの優先 ID プロバイダーで SAML アプリケーションを構成する必要があります。SAML アプリケーションの構成方法については、次のドキュメントを参照してください。

- Okta：Okta の場合、次を実行する必要があります。
  - Okta SAML アプリケーションの追加
  - 用の SAML 2.0 の構成 ServiceNow®
- Microsoft Azure: とのシングルサインオン (SSO) 統合 ServiceNow®
- Ping: を使用した SAML SSO の構成 ServiceNow®
- ADFS：SAML 2.0 との統合

SAML チュートリアルを使用するには、Integration - Multiple Provider Single Sign-On Installer プラグインをアクティブ化します。詳細については、「Multi-Provider SSO プラグインのアクティブ化」を参照してください。

SAML ガイド付きツアーを使用するには、次の手順を実行します。

1. 移動先 すべて > マルチプロバイダー **SSO** > **ID** プロバイダー.
2. [**New**] をクリックします。
3. **[SAML]** を選択します。
4. [ヘルプ] アイコンをクリックします。
5. **[Take Tour]** をクリックします。

SAML ガイド付きツアーは、複数のページにまたがる一連の手順を使用して構成を完了し、ツアーの一部として提供される手順と指示に従ってチュートリアルを完了します。SAML構成の詳細については、「マルチプロバイダー SSO を使用した SAML 2.0 構成」を参照してください。

SAML 2.0 と他の機能の統合

SAML 2.0 ソリューションを、電子署名、ディープリンク、ADFS などの他の機能と統合できます。

SAML のディープリンクサポートの追加

ディープリンクにより、インスタンスはシステム内の特定のレコードへの直接メールリンクをサポートできます。

SAML 2.0 統合を有効にすると、IdP が最初に要求された URL にユーザーをリダイレクトする前に、ディープリンク URL が認証チェックに合格する必要があります。たとえば、https://<instance name>.service-now.com/nav_to.do?uri=incident.do?sys_id=46c88ac1a9fe1981014de1c831fbcf6d という URL がメールに含まれているとします。

インスタンスは IdP に認証要求を送信し、RelayState URL パラメーターを使用して最初に要求されたリソース (この場合は uri=incident.do?sys_id=46c88ac1a9fe1981014de1c831fbcf6d) を保持します。IdP がユーザーを認証した後、インスタンスは RelayState URL パラメーターの値を読み取り、要求されたリソースにユーザーをリダイレクトします (インスタンスに存在する場合)。

ディープリンクのサポートを追加するには、ID プロバイダーが RelayState URL パラメーターをサポートしていることを確認します。

ADFS と SAML 2.0 との統合

ServiceNow Multi-Provider SSO プラグインは、SAML 2.0 シングルサインオン (SSO) と Microsoft ADFS との統合をサポートしています。

ADFS のインストールと構成の詳細については、「Active Directory Federation Services Overview (Active Directory フェデレーションサービスの概要)」を参照してください。Multi-Provider SSO プラグインは、SAML 2.0 SSO と ADFS 2.0、3.0、Azure AD との 統合を使用して構成およびテストされています。複数行の属性は、ADFS に関する ServiceNow の SAML 応答ではサポートされていません。

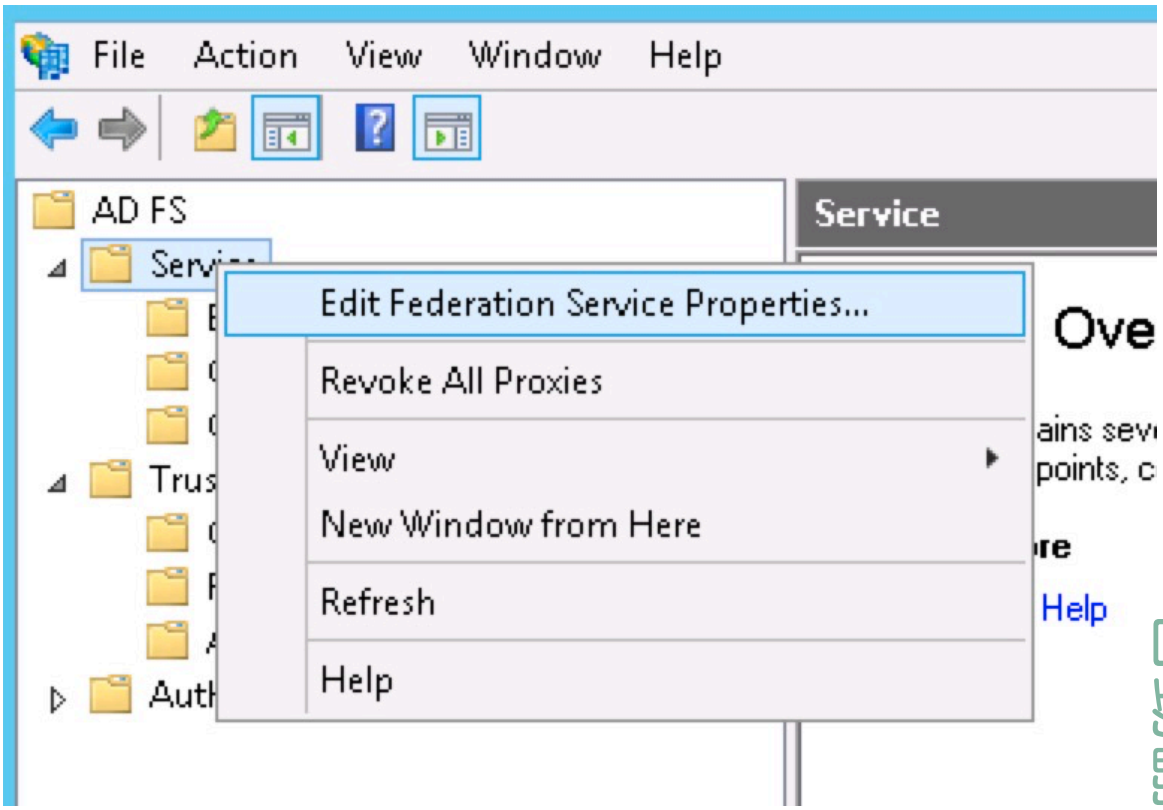
SAML 向け ADFS の設定

SAML 向けの ADFS を設定します。この手順では、ADFS 2.0 を使用し、ADFS Web サイトとして samportal.example.com を表示します。これを ADFS の Web サイトアドレスに置き換えます。

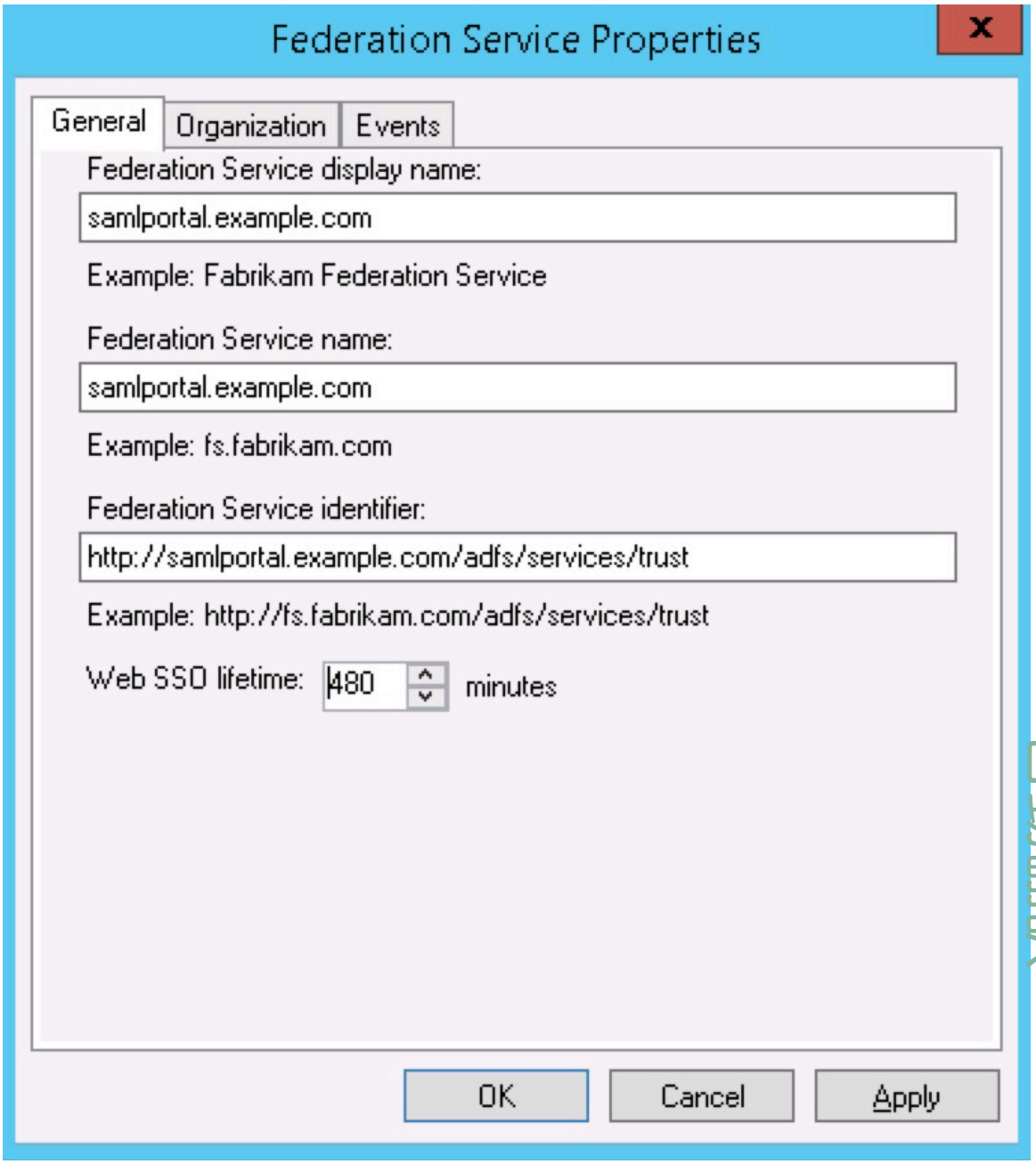
始める前に
必要なロール：admin

手順

1. ADFS 3.0 サーバーにログインして管理コンソールを開きます。
2. [サービス] を右クリックし、[フェデレーションサービスプロパティの編集 **(Edit Federation Service Properties)]** を選択します。



3. 全般設定が DNS エントリと証明書名と一致していることを確認します。



4. 証明書を参照し、トークン署名証明書をエクスポートします。

a. 証明書を右クリックし、[証明書を表示 **(View Certificate)]** を選択します。

b. [詳細] タブを選択します。

c. [ファイルにコピー **(Copy to File)]** をクリックします。
証明書のエクスポートウィザードが開きます。

d. [次へ] を選択します。

e. [いいえ、秘密鍵をエクスポートしません **(No, do not export the private key)]** オプションが選択されていることを確認し、[次へ] をクリックします。

f. **[DER** エンコード バイナリ **X.509 (.cer) (DER encoded binary X.509 (.cer))]** を選択し、[次へ] をクリックします。

g. ファイルを保存する場所を選択して名前を付け、[次へ] をクリックします。

h. [完了] を選択します。
インスタンスでは、この証明書が PEM 形式である必要があります。この証明書は、クライアントツールや、SSL Shopper のようなオンラインツールを使用して変換できます。

5. 作成した DER/バイナリ証明書を使用し、標準 PEM 形式でエクスポートします。

ADFS のインスタンス設定

インスタンスと SAML 2.0 設定を構成して、ADFS を使用できるようにします。

始める前に
これらの手順は、SAML 用の ADFS を設定した後にのみ実行してください。このプロセスの詳細については、「SAML 用の ADFS の設定」を参照してください。

必要なロール：admin

手順

1. まだアクティブでない場合は、 Multi-Provider SSO プラグインのアクティブ化を行います。
2. SAMLを設定します。ただし、IdP 証明書をインストールする際には、SAML 向け ADFS の設定の実行時に作成した PEM 証明書を添付してください。
3. [保存] をクリックします。
4. [問題] および [件名] フィールドに値が入力され、エラーがないことを確認します。
エラーが発生した場合は、保存した PEM 形式の証明書をメモ帳で開き、証明書をコピーして [PEM 証明書] フィールドに貼り付けます。
5. SAML2SingleSignon_update1 インストレーションイグジットがアクティブであることを確認します。
6. SAML 2.0 の構成を続行します。

> 注: ADFS サーバーで証明書が更新されている場合は、更新された証明書へのインスタンスもアップロードする必要があります。

ADFS 証明書利用者の構成

インスタンスメタデータを取得して ADFS サーバーにインポートします。ただし、証明書利用者を手動で構成する方が簡単に実装できます。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > マルチプロバイダー **SSO** > **ID** プロバイダー > **SAML2 Update1** > 暗号化と署名 をクリックし、SAML プロパティ Sign AuthnRequest (`glide.authenticate.sso.saml2.require_signed_authnrequest`) がアクティブでないことを確認します。
署名された要求が必要であることを ADFS アドミニストレーターが確認できる場合にのみ、このプロパティをアクティブにしてください。

2. SAML 2 メタデータリンクを介して生成したメタデータをコピーし、ファイルに保存します。
3. ADFS サーバーにログインして管理コンソールを開きます。
4. [証明書利用者信頼 **(Relying Party Trusts)]** を選択します。
5. ウィンドウの右上隅から [証明書利用者信頼の追加 **(Add Relying Party Trusts)]** を選択します。

   追加ウィザードが表示されます。
6. [開始] をクリックして開始します。
7. [ファイルをインポート] オプションを使用して、メタデータファイルをインポートします。
8. ServiceNow などの表示名を設定し、必要であればメモを入力します。
9. **[ADFS 3.0** プロファイル **(ADFS 3.0 Profile)]** を選択します。
10. トークン暗号化証明書を選択しないでください。
    すでにエクスポートされているサービスで定義されている証明書が使用されます。証明書を定義すると、インスタンスと適切に通信できなくなります。
11. **[URL** の構成 **(Configure URL)]** の設定を有効にしないでください。
12. 証明書利用者信頼の識別子として、接続したインスタンスサイトを入力します。
    この場合は、https://company.service-now.com を使用して、[追加] をクリックします。
13. すべてのユーザーにこの証明書利用者へのアクセスを許可します。
14. [次へ] をクリックし、[終了時に要求を開く **(Open the Claims when this finishes)]** チェックボックスをオフにします。
15. このページを閉じます。
    新しい証明書利用者信頼がウィンドウに表示されます。
16. 証明書利用者信頼を右クリックし、[プロパティ] を選択します。
17. [詳細] タブを参照して、[セキュアハッシュアルゴリズム **(Secure hash algorithm)]** を SHA-256 または SHA-1 に設定します。
18. [エンドポイント] タブを参照し、**SAML Assertion Consumer** をポストバインディングと https://company.service-now.com/navpage.do の URL で追加します。

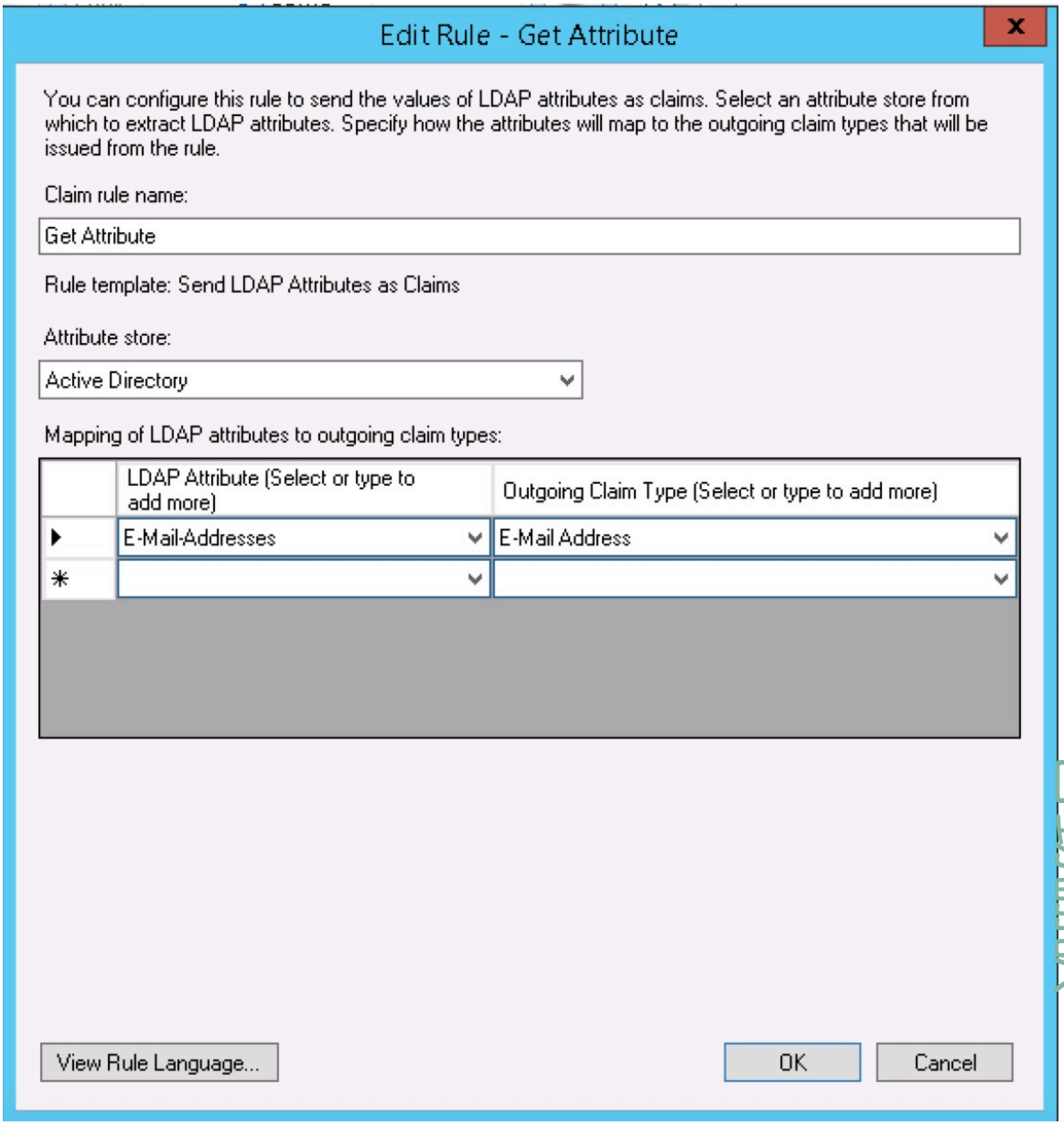
ADFS 証明書利用者要求ルールの構成

要求ルールを編集して、インスタンスと適切に通信できるようにします。

始める前に
必要なロール：admin

手順

1. ADFS サーバーにログインして管理コンソールを開きます。
2. 証明書利用者信頼を右クリックして、[要求ルールの編集 **(Edit Claim Rules)]** を選択します。
3. [発行変換ルール **(Issuance Transform Rules)]** タブをクリックします。
4. [ルールの追加 **(Add Rules)]** を選択します。
5. 使用する要求ルールのテンプレートとして **[LDAP** 属性を要求として送信 **(Send LDAP Attribute as Claims)]** を選択します。
6. 要求に「Get LDAP Attributes」などの名前を付けます。
7. [属性ストア **(Attribute store)]** を [Active Directory]、**[LDAP** 属性] を [E-Mail-Addresses]、[送信要求タイプ **(Outgoing Claim Type)]** を [E-Mail-Addresses] に設定します。



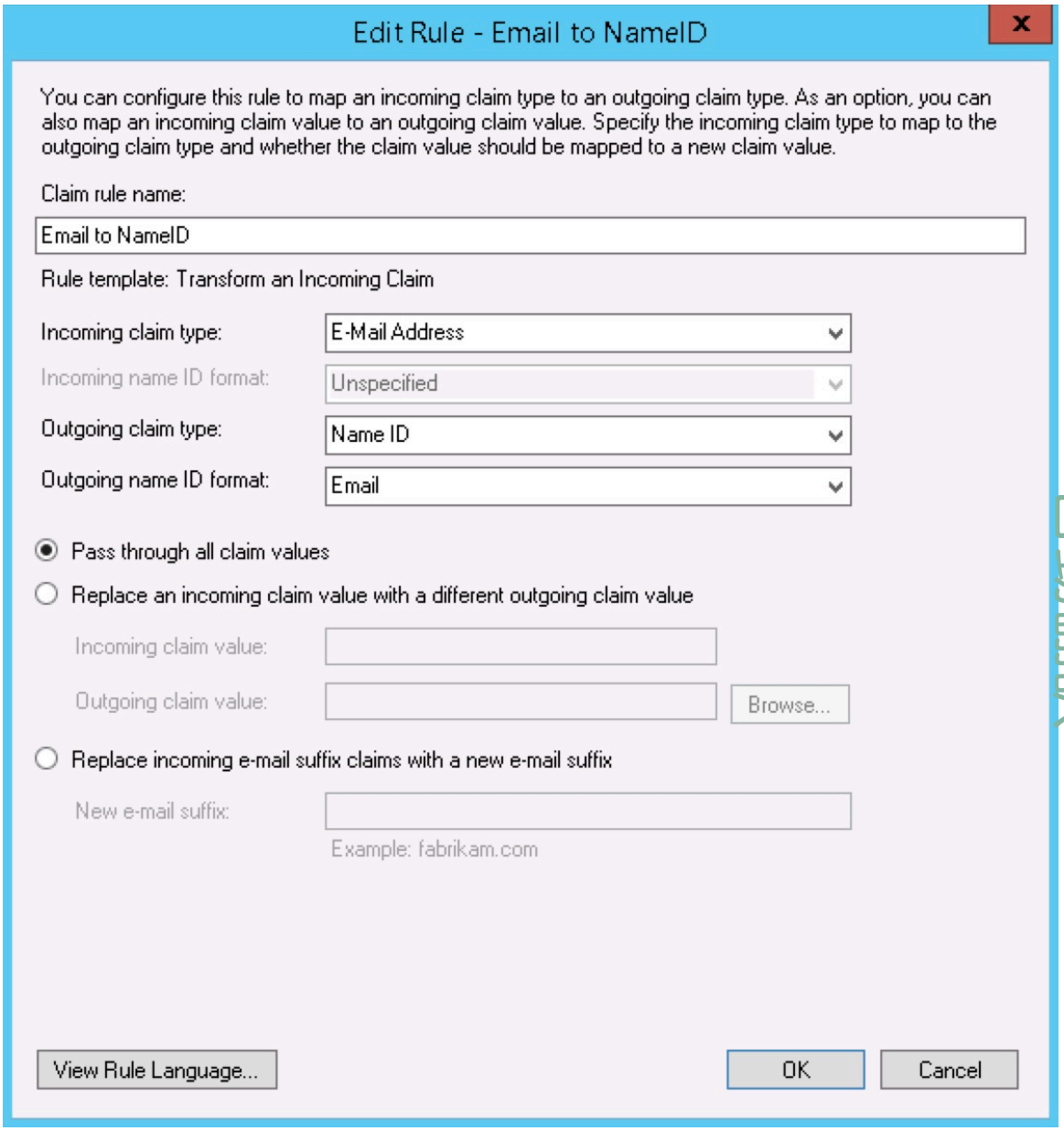

```
c:[Type ==
 "http://schemas.microsoft.com/ws/2008/06/identity/claims/windowsaccountname", Issuer
 == "AD AUTHORITY"]
=> issue(store = "Active Directory",
types = ("http://schemas.xmlsoap.org/ws/2005/05/identity/claims/emailaddress"),
query = ";mail;{0}", param = c.Value);
```

8. [完了] を選択します。
9. [ルールの追加 **(Add Rules)**] を選択します。
10. 使用する要求ルールテンプレートとして [受信要求を変換 **(Transform an Incoming Claim)**] を選択します。
11. 要求に、「Email to Name ID」などの名前を付けます。
12. [受信要求タイプ **(Incoming claim type)**] を前のルールの [送信要求タイプ **(Outgoing Claim Type)**] に設定します。
    たとえば「E-Mail Address」などとします。

13. [送信要求タイプ **(Outgoing claim type)]** を [Name ID] に設定し、[送信名 **ID** 形式 **(Outgoing name ID format)]** を [Email] に設定します。

    注: これらの値は、SAML 2.0 の構成時に定義する名前 ID ポリシーと一致する必要があります。

14. [すべての要求値をパススルー] を選択します。



この要求ルールは、次のルール言語のようになります。

```
c:[Type == "http://schemas.xmlsoap.org/ws/2005/05/identity/claims/emailaddress"]
 => issue(Type = "http://schemas.xmlsoap.org/ws/2005/05/identity/claims/nameidentifier",
Issuer = c.Issuer, OriginalIssuer = c.OriginalIssuer, Value = c.Value, ValueType =
 c.ValueType,
Properties["http://schemas.xmlsoap.org/ws/2005/05/identity/claimproperties/format"]
= "urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress");
```

15. [完了] をクリックします。

SAML ログアウトエンドポイントの作成

SAML ログアウトエンドポイントを作成してシングルログアウトを許可します。

始める前に
必要なロール：admin

このタスクについて
詳細については、ADFS サインアウトに関するこの記事 を参照してください。

手順

1. 検索項目 **ADFS** マネージャー > 信頼関係 > 証明書利用者信頼 > プロパティ.
2. [エンドポイント] タブで、**[SAML** を追加 **(Add SAML)]** をクリックします。
3. 設定の構成：
   - エンドポイントタイプ：SAML ログアウト
   - バインディング：POST
   - 信頼できる **URL (Trusted URL)**：ADFS サーバーの URL。例を次に示します。

     ```
     https://myadfsserver.domain.net/adfs/ls/?wa=wsignout1.0
     ```

   - 応答 **URL**：アプリケーションのログアウト URL。例を次に示します。

     ```
     https://{instancename}.service-now.com/external_logout_complete.do
     ```

ADFS 構成のテスト

ADFS 構成をテストして、ID プロバイダーとして適切に機能していることを確認します。

始める前に
必要なロール：admin

手順

1. Internet Explorer ブラウザを開きます。
2. ADFS ポータルに移動します。

   例：https://samportal.example.com/adfs/ls/idpinitiatedsignon.aspx このページには、構成されている証明書利用者信頼をすべて含むドロップダウンリストがあります。

3. インスタンスに関連付けられている証明書利用者を選択します。
4. [サインインを続行 **(Continue to Sign In)]** をクリックします。

   SAML 2.0 外部認証を適切に設定している場合は、インスタンスに自動的にログインします。

5. https://samportal.example.com/adfs/ls/idpinitiatedsignon.aspx?logintoRP=https://company.service-now.com に移動して、直接ログイン URL をテストします。

(ワークアラウンド) サービスプロバイダーが開始する認証の有効化

SAML 2.0 Update 1 がないために認証に失敗した場合は、ワークアラウンドを使用します。この問題は、ユーザーが IdP 認証をスキップしてインスタンスに直接移動しようとした場合に発生する可能性があります。

始める前に
必要なロール：admin

このタスクについて
このエラーは、SAMLResponse の SPNameQualifier 属性に必要な定義とセマンティクスを、インスタンスが ADFS に提供しないときに発生します。

次のいずれかを実行し、サービスプロバイダーが開始する認証を有効にします。

手順

- SAML 2.0 Update 1 にアップグレードし、 AuthnContextClass 要求を作成するオプションをオフにします。
- SAML 2.0 がアクティブ (SAML 2.0 Update 1 ではない) になっている場合は、**SAML2** スクリプトインクルードを変更して、SPNameQualifier 属性の定義をコメントアウトします。

  createNameID および createNameIDPolicy 関数の次の行をコメントアウトします。

```
//nid.setSPNameQualifier (serviceURL ) ;

 //nameIdPolicy. setSPNameQualifier (serviceURLStr ) ;
```

次のタスク
インスタンスへのアクセス時に ADFS サーバーからのログインプロンプトを表示したくない場合は、次の SAML 2.0 Update 1 プロパティを false に設定します：**AuthnRequest** ステートメントで **AuthnContextClass** 要求を作成する (*glide.authenticate.sso.saml2.createrequestedauthncontext*)。

(ワークアラウンド) **Kerberos** 認証のサポート

SAML 2.0 統合には、認証コンテキストをフォームベースの認証から Windows ベースの認証に変更するワークアラウンドがあります。

始める前に
必要なロール：admin

このタスクについて
現在、SAML 2 統合では、PasswordProtectedTransport または「フォームベースの認証」認証コンテキストを使用しています。この認証コンテキストでは、ユーザーに認証情報のフォームを表示するために IdP が必要です。Kerberos では、確立された Windows ログインを通じて SAML セッションが既にアクティブになっているため、ユーザーは IdP で認証する必要はありません。

手順

1. 移動先 すべて > マルチプロバイダー **SSO** > **ID** プロバイダー.
2. **[SAML2 Update1]** IdP レコードを開きます。
3. **[SAML 2.0 AuthnRequest to the Identity Provider** に含める **AuthnContextClassRef** メソッド] を、次のいずれかに設定します。

AuthnContextClassRef メソッドの値
urn:oasis:names:tc:SAML:2.0:ac:classes:PasswordProtectedTransport (デフォルト)
urn:federation:authentication:windows

4. [更新] をクリックします。

Azure AD と **SAML 2.0** との統合

ServiceNow を Azure Active Directory (Azure AD) と統合します。

ServiceNow を Azure AD と統合すると、次のことが可能になります。

- ServiceNow にアクセスできる Azure AD をコントロールする
- ユーザーが Azure AD アカウントで ServiceNow に自動的にサインインできるようにする
- Azure ポータルでアカウントを管理する

前提条件

開始するには、次を実行する必要があります。

- Azure AD のサブスクリプション。サブスクリプションがない場合は、無料のアカウントを取得できます。
- ServiceNow シングルサインオン (SSO) が有効になっている。
- San Diego バージョン以降をサポートする ServiceNow の ServiceNow インスタンスまたはテナント。
- ServiceNow テナントで Multiple Provider Single Sign On プラグインを有効にする必要があります。
- 自動構成の場合は、ServiceNow のマルチプロバイダープラグインを有効にする必要があります。

構成に対する一連のアクション

Azure AD を構成するために実行する必要がある一連のアクションは次のとおりです。

- ギャラリーから ServiceNow を Azure AD に追加する
- Azure AD SSO を構成する
- Azure AD テストユーザーを作成する
- Azure AD テストユーザーをアサインする
- ServiceNow を構成する

ギャラリーから **ServiceNow** を追加する

ギャラリーから Azure AD の管理対象 SaaS アプリのリストに ServiceNow を追加します。

始める前に

必要なロール：Azure admin

手順

1. Microsoft アカウントを使用して Azure ポータルにサインインします。
2. 左ペインから **[Azure Active Directory]** サービスを選択します。
3. 移動先 エンタープライズアプリケーション > すべてのアプリケーション.
4. 新しいアプリケーションを追加するには、[新しいアプリケーション] を選択します。
5. [ギャラリーから追加する] セクションで、検索ボックスに「ServiceNow」と入力します。
6. 結果のパネルから ServiceNow を選択し、アプリを追加します。
7. アプリがテナントに追加されるまで数秒待ちます。

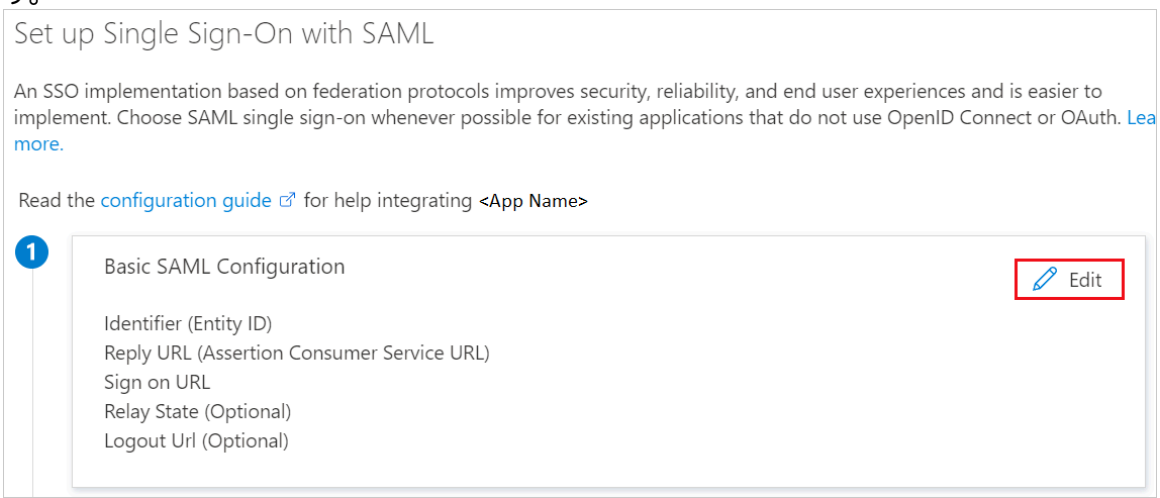
Azure AD SSO の構成

Azure ポータルで Azure AD SSO を構成します。

始める前に
必要なロール：Azure admin

手順

1. Azure ポータルの ServiceNow アプリケーション統合ページで、[管理] セクションを見つけます。
2. [シングルサインオン] を選択します。
   [シングルサインオン方式の選択 single sign-on method] ページで [SAML] を選択します。
3. [SAML シングルサインオンのセットアップ (Set up single sign-on with SAML)] ページで、[基本的な SAML 構成 (Basic SAML Configuration)] の鉛筆アイコンを選択して設定を編集します。



4. [基本的な SAML 構成 (Basic SAML Configuration)] セクションで、次の手順を実行します。

   a. [サインオン **URL (Sign on URL)**] に、次のいずれかの URL パターンを入力します。

   ```
   https://<instancename>.service-now.com/navpage.do
   https://<instance-name>.service-now.com/login_with_sso.do?glide_sso_id=<sys_id of the sso configuration>
   ```

   注: この URL 内に sys_id を指定する必要があります。

   b. [識別子 (エンティティ ID) (Identifier (Entity ID))] で、パターンが `https://<インスタンス名>.service-now.com` の URL を入力します。

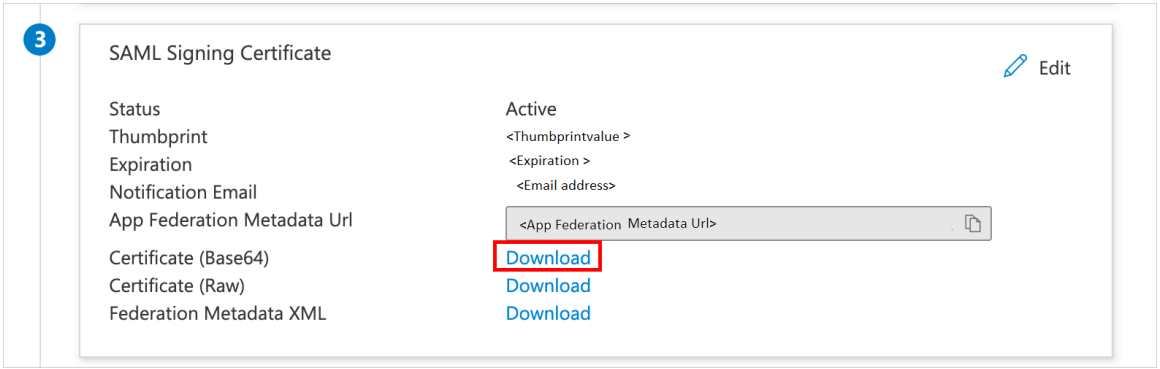
   c. [応答 **URL (Reply URL)**] に、次のいずれかの URL パターンを入力します。

   ```
   https://<instancename>.service-now.com/navpage.do
   https://<instancename>.service-now.com/customer.do
   ```

   d. [ログアウト URL] に、パターンが `https://<インスタンス名>.service-now.com/navpage.do` の URL を入力します。

   注: 実際のサインオン URL、応答 URL、ログアウト URL、および識別子を更新する必要があります。これらの URL に表示される値はデモ用です。

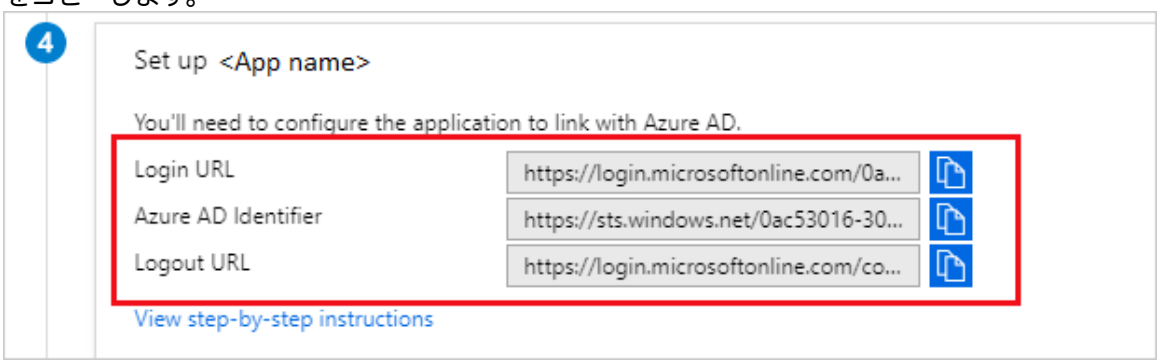
5. [SAML シングルサインオンのセットアップ (Set up single sign-on with SAML)] ページの [SAML 署名証明書 (SAML Signing Certificate)] セクションで、[証明書 **(Base64) (Certificate (Base64))**] を見つけます。



a. [コピー] ボタンを選択して [アプリフェデレーションメタデータ **URL (App Federation Metadata Url)]** をコピーし、メモ帳に貼り付けます。
この URL は追加の構成で必要です。

b. [ダウンロード] を選択して証明書 (Base64) をダウンロードします。

6. [ServiceNow のセットアップ (Set up ServiceNow)] セクションで、要件に基づいて適切な URL をコピーします。



Azure AD テストユーザーを作成する

Azure AD でテストユーザーを作成します。

始める前に
必要なロール：Azure admin

手順

1. Azure ポータルで、 すべて > **Azure Active Directory** > ユーザー > 全ユーザー.

2. [新規ユーザー] を選択します。

3. [ユーザープロパティ] で、次の情報を入力します。

- 名前
- ユーザー名
- パスワード

4. [作成] をクリックします。

Azure AD テストユーザーをアサインする

ServiceNow へのアクセス権を付与することで、Azure シングルサインオンを使用するために作成された Azure AD テストユーザーをアサインします。

始める前に
必要なロール：Azure admin

手順

1. Azure ポータルで、 すべて > エンタープライズアプリケーション > すべてのアプリケーション.
2. アプリケーションリストから、ServiceNow を選択します。
3. アプリのページの [管理] セクションで、[ユーザーとグループ] を選択します。
4. [ユーザーを追加] を選択します。
5. [アサインを追加] ウィンドウで、[ユーザーとグループ] を選択します。
6. [ユーザーとグループ] ウィンドウで、ユーザーリストから作成された [test user] を選択します。
7. 必要に応じて、[ロールの選択 **(Select a role)]** ドロップダウンからアサインされたユーザーのロールを選択します。
8. [アサインを追加] ウィンドウで、[アサイン **(Assign)]** を選択します。

ServiceNow の構成

SSO を使用するには、ServiceNow で Azure AD の詳細を構成します。

始める前に
プラグイン：Integration - Multiple Provider single sign-on Installer

マルチプロバイダー SSO のプロパティを有効にします。

- [マルチプロバイダー **SSO** を有効化 **(Enable multiple provider SSO)]** を選択します。
- [デフォルトの **ID** プロバイダーからユーザーテーブルへのユーザーの自動インポートを有効化] を選択します。
- [マルチプロバイダー **SSO** 統合のデバッグログを有効化 **(Enable debug logging for the multiple provider SSO integration)]** を選択します。
- ユーザーテーブル上のフィールドに「**email**」と入力します。

必要なロール：admin

手順

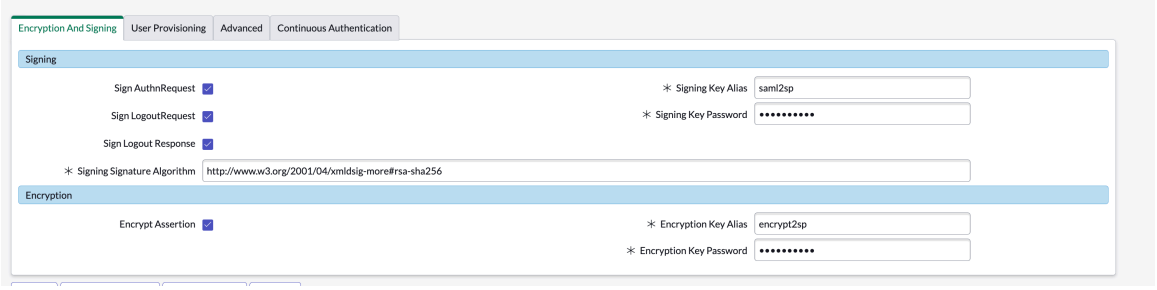
1. 移動先 すべて > マルチプロバイダー **SSO** > **ID** プロバイダー.
2. [ID プロバイダー] ページで、[新規] を選択します。
3. [ID プロバイダー] ウィンドウで、**[SAML]** を選択します。
4. [ID プロバイダーのメタデータをインポート] で、次のいずれかを実行できます。
   - **URL**：ID プロバイダー構成ページに詳細を自動入力するためのアプリフェデレーションメタデータ URL。
   - インポート：XML をインポートして、[ID プロバイダー構成] ページの詳細をインポートします。
5. 画面の上部を右クリックし、**[sys_id** のコピー] をクリックし、この値を [基本的な **SAML** 構成] セクションの [サインオン **URL]** で使用します。
6. フォームのフィールドに入力します。

[マルチプロバイダー **SSO]** フィールド

プロパティ	必須	説明
名前	はい	IdP の名前。この IdP は自動リダイレクト Sys ID です。
有効	はい	IdP を認証に使用するには、[有効] を true に設定する必要があります。 **注:** このプロパティを設定するオプションは、テスト接続が成功した後にのみ表示されます。
デフォルト	いいえ	自動リダイレクト IdP (旧称「プライマリ IdP」) は、ユーザーをベースインスタンス URL に自動的にリダイレクトします。このプロパティは、この IdP 構成をデフォルトとして設定します。
自動リダイレクト IdP	いいえ	自動リダイレクト IdP として設定できる IdP 構成。 **注:** 新しい自動リダイレクト IdP 構成をアクティブにすると、 `glide_sso_id` cookie が新しい自動リダイレクト IdP で更新されます。自動的に有効になる `glide.authenticate.sso.update.idp.cookie` システムプロパティは、この機能を制御します。
ID プロバイダー URL	はい	IdP への URL。各 IdP URL は一意である必要があります。
ID プロバイダーの AuthnRequest	はい	SingleSignOnService 要素から取得した HTTP リダイレクトバインディングへの URL。
ID プロバイダーの SingleLogoutRequest	いいえ	SingleLogoutService 要素から取得した URL。
ServiceNow のホームページ	はい	IdP が認証するインスタンスの URL (ログインページを含む)。例：https://yourinstance.service-now.com/navpage.do
エンティティ ID/発行者	はい	IdP が認証するインスタンスのベース URL (ログインページを除く)。例：https://yourinstance.service-now.com/
対象者 URI	はい	IdP が認証するインスタンスのベース URL (ログインページを除く)。例：https://yourinstance.service-now.com/
NameID ポリシー	はい	統合で使用される NameIDFormat 要素の値。
外部のログアウトのリダイレクト	いいえ	ログアウト後に統合によりリダイレクトされる URL。
要件のリダイレクトに失敗	いいえ	失敗した認証要求をリダイレクトするための URL。デフォルトでは、これは IdP で設定されたエラーページまたはログアウトページの URL エンドポイントです。この値は glide.authenticate.failed_requirement_redirect フィールドに入力できます。

7. オプション: [暗号化と署名] タブ

注: 暗号化と署名には独自の証明書を使用します。

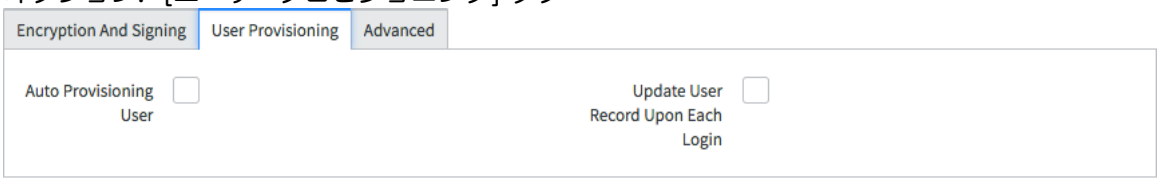


タブ

[暗号化と署名] フィールド

プロパティ	説明
署名/暗号化キーのエイリアス	SAML 2.0 SP Keystore に保存されているキーエントリのエイリアス。
署名キーパスワード	SAML 2.0 SP Keystore に保存されているキーエントリのパスワード。
アサーションを暗号化	SAML 応答のアサーションを暗号化するためのチェックボックス。IDP 用に生成されたメタデータには、IDP が生成する SAML 応答のアサーションを暗号化するために使用する x509 証明書が埋め込まれています。
署名アルゴリズムの署名	eSignature 認証用の SAML 2.0 Identity Provider AuthnRequest Consumer を示す URL。
AuthnRequest に署名	IdP Single Sign-on サービスが署名済みの AuthnRequest を受信できるようにするためのチェックボックス。
LogoutRequest の署名	IdP Single Sign-on サービスが署名済みの LogoutRequest を受信できるようにするためのチェックボックス。

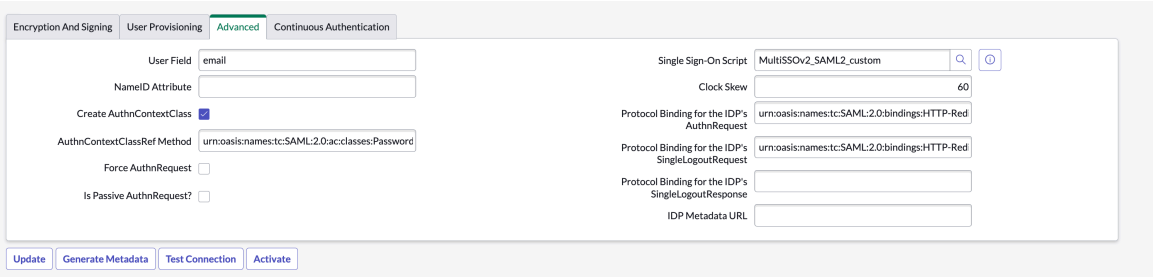
8. オプション: [ユーザープロビジョニング] タブ



[ユーザープロビジョニング] フィールド

プロパティ	説明
自動ユーザープロビジョニング	自動ユーザープロビジョニングでは、IdP によって提供される情報に基づいてユーザーがインスタンスユーザーテーブルに存在しない場合に、ユーザーを作成します。
各ログイン時のユーザーレコードの更新	ユーザーが SAML を使用してログインするたびに、インスタンスユーザーテーブルのユーザー情報を IdP の情報で更新します。

9. オプション: [詳細] タブ



[詳細] フィールド

プロパティ	説明
ユーザーフィールド	ユーザーを識別するために IdP に必要な値を含むユーザーテーブルのフィールド。これは応答の一部としての一意の ID です。たとえば、ユーザー名、従業員 ID などがあります。システムユーザーテーブルで、この一意の ID がユーザーの詳細と照合されます。
NameID 属性	新しい NameID ポリシーを設定する場合を除き、空白のままにするフィールド。新しいポリシーを設定する場合、システムではログインするユーザーを識別するためのユーザーテーブルを使用する必要があります。システムは NameID トークンを、そのユーザーテーブルフィールドの名前と照合します。
AuthnContextClass の作成	[パスワードで保護されたトランスポート (Password Protected Transport)] などの特定のコンテキストクラスを指定するためのチェックボックス。チェックボックスをオフにすると、IdP によって最も適切なコンテキストクラスが選択されます。
AuthnContextClassRef メソッド	IdP がユーザーを認証するために使用するログインメカニズムの URN。
AuthnRequest の強制	AuthnRequest を強制的に発生させるためのチェックボックス。
Passive AuthnRequest か	AuthnRequest がパッシブの場合にオンにするチェックボックス。
シングルサインオンスクリプト	シングルサインオンスクリプト。デフォルトは *MultiSSOV2_SAML2_custom* です。
ログアウト応答に署名	このフィールドにログアウト応答の詳細入力します。
クロックスキュー	SAMLResponse ノンスを構成する 2 つの属性間の秒数。デフォルトは 60 です。有効な SAMLResponse は、*notBefore* と *notOnOrAfter* の日付と時刻の値の範囲内である必要があります。SAMLResponse メッセージのサンプルについては、「Sample SAML 2 Response with the SubjectConfirmation and SubjectConfirmationData Elements (SubjectConfirmation および SubjectConfirmationData 要素を含む SAML 2 応答のサンプル)」と、「Sample SAML 2 Response with the AudienceRestrictions and Audience Elements (AudienceRestrictions および Audience 要素を含む SAML 2 の応答サンプル)」を参照してください。
IDP の SingleLogoutRequest のプロトコルバインディング	SingleLogoutService 要素の [バインディング] 属性に記載されているサポート対象の値の 1 つ。
IDP プロパティのインポート元のメタデータ URL	IdP プロパティはこの URL からインポートされます。設定すると、以前の証明書の有効期限が切れた場合に、IdP からの SAML 証明書の自動インポートが有効になります。

自動翻訳

プロパティ	説明
	注: SAML2 Update 1 からマルチプロバイダー SSO にアップグレードする場合、または SSO 接続を手動で設定する場合、IdP メタデータ URL は自動的に入力されません。
要求	要求の一部としての一意の ID。ID には、ユーザー名、従業員 ID などがあります。 注: 要求では、リダイレクトとポストバインディングの両方がサポートされています。このフィールドを設定するオプションは、テスト接続が成功した後にのみ表示されます。詳細については、「IdP 接続をテスト」を参照してください。
応答	応答の一部としての一意の ID。ID には、ユーザー名、従業員 ID などがあります。 注: 応答では、リダイレクトとポストバインディングの両方がサポートされています。このフィールドを設定するオプションは、テスト接続が成功した後にのみ表示されます。詳細については、「IdP 接続をテスト」を参照してください。

10. ページの右上隅にある [テスト接続] を選択します。
11. 認証情報を入力します。
    [SSO ログアウトのテスト結果] が表示されます。
12. [アクティブ化] を選択して、構成をアクティブ化します。

外部認証を使用したメールリンク

ダイジェストトークン外部認証を使用する場合はメールリンクを使用できますが、メール通知のリンクの処理方法を確立する必要があります。

デフォルトのリンクには、SSO 認証情報を取り込まずに、インシデントや変更要求などのインスタンス内の特定の場所に誘導する URL が含まれています。以下は、インスタンスのログインページでログインせずにユーザーをインスタンス内の場所に誘導する例です。

- /demo インスタンスにコネクトするための非暗号化 HTTP テクニック (特定のレコードに移動しない):

```
https://<instance
 name>.service-now.com/?SM_USER=user_name&DE_USER=lQjIVp7aRJtyPx5+2O/vgU24tbE=
```

- ユーザーが最初に自社のログインポータルに移動するための特定のレコードへのリンク (メール通知内):

```
https://login.company_portal_page.com/nav_to.do?uri=incident.do?sys_id=009f8eda0a0a0b2b01ab4eb094223466%26sysparm_stack=incident_list.do%3Fsysparm_query=active=true
```

  インスタンスの *glide.email.override.url* プロパティを設定して、会社のポータルページの URL を含める必要があります。このプロパティが存在しない場合は作成できます。

- 次に、企業ポータルはその URL を取得して、インスタンスへのリダイレクト URL をビルドする必要があります。その際、特定のレコードへのアクセスに必要なセグメントは保持され、URL の末尾に SSO 認証情報が追加されます。

```
https://<instance
 name>.service-now.com/nav_to.do?uri=incident.do?sys_id=009f8eda0a0a0b2b01ab4eb0942
23466%26sysparm_stack=incident_list.do%3Fsysparm_query=active=true&SM_USER=user_
name&DE_USER=lQjlVp7aRJtyPx5+2O/vgU24tbE=
```

SAML の電子署名サポートの追加

セキュリティアサーションマークアップ言語 (SAML) 2.0 update 1 で電子署名の次のプロパティを設定します。

マルチ SSO で電子署名が有効な場合、SAML プロパティは使用されません。電子署名プロパティが SAML2 Update1 プロパティ [saml2_update1_properties] テーブルに追加されます。

プロパティ	説明	デフォルト
eSignature 認証用の Assertion Consumer Index	エンドポイントを識別するインデックス番号	1
eSignature 認証用の Assertion Consumer URL	コンシューマーを識別する URL	https://yourinstance.service-now.com/consumer.do
eSignature 認証用の AuthnRequest URL	認証用の URL	なし

SAML 1.0 または SAML 2.0 (update 1 を含まない) で電子署名を使用している場合は、特別な設定手順「マルチプロバイダー SSO の電子署名」を参照してください。

> **注:** 電子署名を使用する生命科学関連のお客様の場合は、ユーザーのセルフロックアウト防止ビジネスルールを非アクティブ化してください。

既存の SAML 1.1 統合から SAML 2.0 への移行

SAML 1.1 統合から SAML 2.0 統合に移行するには、カスタマーサポートにお問い合わせください。

既存の SAML 2.0 統合の更新

既存の SAML 2.0 統合を更新します。

始める前に

必要なロール：admin

このタスクについて

SAML 2.0 Update 1 プラグインの要求カスタマーサービス & サポート に連絡して、*SAML 2.0* シングルサインオン - *Update 1: security enhancements* プラグインを要求します。このプラグインは、更新されたバージョンの *SAML2SingleSignon* インストレーションイグジット (ログインスクリプト)、*SAML2Logout* インストレーションイグジット (ログアウトスクリプト)、および *SAML2* スクリプトインクルード (スクリプトオブジェクト) に適用されます。

既存のインストレーションイグジットスクリプトからのカスタマイズを新しいスクリプトに結合します。更新により、統合のオリジナルのインストレーションイグジットスクリプトの非アクティブなコピーが保存されます。これらのコピーを使用して、ログインおよびログアウトスクリプトに加えたカスタマイズをすべて、これらのインストールの新しいバージョンに結合できます。

既存のインストレーションイグジットスクリプトのカスタマイズを新しいスクリプトに結合

オリジナルのインストレーションイグジットスクリプト名	オリジナルのスクリプトステータス	新規インストレーションイグジットのスクリプト名	新規スクリプトステータス
SAML2SingleSignon	非アクティブ	SAML2SingleSignon_update1	アクティブ
SAML2	非アクティブ	SAML2_update1	アクティブ
SAML2Logout	非アクティブ	SAML2Logout_update1	アクティブ

SAML 2.0 ログインおよびログアウトのインストレーションイグジットスクリプトには、次のようなパスで移動できます。

- **SAML 2** シングルサインオン > ログインスクリプト.
- **SAML 2** シングルサインオン > ログアウトスクリプト.
- システム定義 > インストレーションイグジット.

SAML 2.0 update 1 スクリプトインクルードには、次のようなパスで移動できます。

- **SAML 2** シングルサインオン > スクリプトオブジェクト.
- システム定義 > スクリプトインクルード.

更新をテストします。

手順

1. 値が *true* の `glide.authenticate.sso.saml2.debug` というシステムプロパティを追加します。
2. SAML 2.0 ログインを試行します。
3. システムログを確認します。
   SAML2 検証エラーの先頭の文字は *SAML2ValidationError* です。
4. 一般的なログインエラーを特定して修正します。
   詳細については、「マルチ SSO (SAML 2.0) のエラーと修正」を参照してください。

更新後の **SAML 2** 応答のサンプル

次のセクションでは、IdP が SAML 応答で指定する必要がある新しい要素と属性を示しています。

Issuer 要素を含む **SAML 2** 応答のサンプル

次の SAML 2 応答では *Issuer* 要素を使用します。

```
<samlp:Responsexmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
 Destination="https://demoi2.service-now.com/navpage.do"
 ID="s28da6774c88ae1eab292bf25fe625db81919d8e1e"
InResponseTo="SNC841720c227c81948cfd68cadcad235c6"
IssueInstant="2012-01-30T20:07:10Z"  Version="2.0"><saml:Issuer
xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion">http://idp.ssocircle.com</saml:Issuer>
  ...
  <saml:Assertion xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
    ID="s2f347f973c063836cf70ea38302d94976f9c5b851"
IssueInstant="2012-01-30T20:07:10Z"
Version="2.0"><saml:Issuer>http://idp.ssocircle.com</saml:Issuer>
     ...
  </saml:Assertion></samlp:Response>
```

SubjectConfirmation および SubjectConfirmationData 要素のあるサンプル SAML 2 応答

次に示す SAML 2 応答は、 *NotOnOrAfter* および *Recipient* 属性を備えた *SubjectConfirmation* および *SubjectConfirmationData* 要素を使用します。

```
<saml:SubjectConfirmationMethod="urn:oasis:names:tc:SAML:2.0:cm:bearer"><saml:Subject
ConfirmationData InResponseTo="SNC841720c227c81948cfd68cadcad235c6"
 NotOnOrAfter="2012-01-30T20:17:10Z"
 Recipient="https://demoi2.service-now.com/navpage.do"/></saml:SubjectConfirmation>
```

AudienceRestrictions と Audience 要素を含むサンプル SAML 2 応答

次に示す SAML 2 応答は、*NotBefore* および *NotOnOrAfter* 属性を備えた *AudienceRestrictions* および *Audience* 要素を使用します。

```
<saml:ConditionsNotBefore="2012-01-30T19:57:10Z"
 NotOnOrAfter="2012-01-30T20:17:10Z"><saml:AudienceRestriction><saml:Audience>http
s://
demoi2.service-now.com</saml:Audience></saml:AudienceRestriction></saml:Conditions>
```

SAML ユーザープロビジョニング

ユーザーが IdP に存在していてもインスタンスに存在しない場合は、SAML ユーザープロビジョニングによってインスタンスのユーザー [sys_user] テーブルにユーザーが自動的に作成されます。

マルチ SSO が有効になっている場合、SAML 2.0 Update 1 では SAML ユーザープロビジョニングがサポートされます。

SAML ユーザープロビジョニングの仕組み

SAML ユーザープロビジョニングが有効になっている場合、インスタンスに存在しない新しいユーザーが検出されると、インスタンスによって `u_imp_saml_user_<suffix>` という名前の一時テーブルにレコードが自動的に作成されます。`<suffix>` には自動的に生成されるテキスト識別子が入ります。インポートテーブルとユーザーテーブルの間のデータ関係を指定する変換マップも作成されます。システムで識別される各 IdP には、独自の変換マップがあります。変換マップは IdP ごとに 1 回作成されます。アドミニストレーターは必要に応じて変換マップを更新できます。

ユーザーがログインすると、IdP にアクセスしてログインします。

- SAML ユーザープロビジョニングを使用できるすべての IdP のリストが表示されます。SAML ユーザープロビジョニングを使用できる IdP が 1 つしかない場合は、それが自動的に使用されます。
- 上記のいずれの条件にも当てはまらない場合、自動リダイレクト IdP が使用されます。

SAML ユーザープロビジョニングの管理

IdP のユーザーでユーザーテーブルを更新するには、最初にフィールドマッピングを設定してから、マルチ SSO IdP 設定を使用してユーザープロビジョニングを有効にします。

始める前に
IdP マッピングを設定して、ユーザーテーブルの正しいフィールドにマッピングされている IdP のフィールドを特定します。

必要なロール：admin

手順

1. 移動先 すべて > マルチプロバイダー **SSO** > プロパティ.
2. この機能をアクティブ化するには、[デフォルトの **ID** プロバイダからユーザーテーブルへのユーザーの自動インポートを有効化] (*glide.authenticate.multisso.user.autoprovision*) を選択します。
3. **[Save** (保存)] をクリックします。
4. 移動先 マルチプロバイダー **SSO** > **ID** プロバイダー.
5. 使用する ID プロバイダーレコードを開きます。
6. ユーザーがまだ存在しない場合にユーザーテーブルにレコードを作成するには、[ユーザーの自動プロビジョニング **(Auto-provision Users)]** を選択します。
   このリリースにアップグレードした場合は、フォームを設定してこのフィールドを追加する必要があります。
7. ユーザーが IdP にログインし、IdP の情報がユーザーテーブルの情報よりも古い場合にユーザーレコードを更新できるようにするには、[各ログイン時のユーザーレコードの更新] を選択します。
   このリリースにアップグレードした場合は、フォームを設定してこのフィールドを追加する必要があります。
8. [ユーザープロビジョニングの変換マップ] をクリックして、自動的に作成されるマップを表示します。
9. 必要に応じてマップを変更します。

結果
不明ユーザーが最初にログインしようとすると、metadata.xml ファイルからインポートセットテーブルにフィールドが作成されます。

注: この最初のユーザーがログインするまで、IdP テーブルのフィールドはマップできません。

SAML 2.0 のトラブルシューティング

サポートに連絡する前に、Hi に基づくナレッジベースで入手可能なトラブルシューティングの解決策を試してみてください。

注: インスタンスは、外部サイトから提供されるソリューションをサポートしていません。

ナレッジベース記事：KB0540617「SAML Error Matrix (SAML エラーマトリクス)」 を参照してください。

その他のよくある問題

エラーまたは症状	解決策
エラーメッセージ：「関数ではありません (is not a function)」 この問題は、マルチノード環境で発生する可能性があります。プラグインがすべてのノードでアクティブ化されていない場合、次のようなエラーが表示されます。 org.mozilla.javascript.EcmaError：*[JavaPackage org.opensaml.saml2.core.impl.AuthnRequestBuilder]* は関数ではありません (is not a function)。	このエラーは、プラグインがアクティブではなく .jar ファイルをロードしなかったために発生します。したがって、コードが欠落しているように見えます。プラグインが存在しないノードを再起動するには、テクニカルサポートにお問い合わせください。
SAML が CMS ページにアクセスするユーザーを認証しません。	デフォルトでは、CMS ページは公開されているため、認証

その他のよくある問題 (続く)

エラーまたは症状	解決策
	は必要ありません。SAML で CMS ページを認証する場合は、*view_content.do* 公開ページを *active=true* から *active=false* に変更します。
SAML 認証後にユーザーを CMS ページにリダイレクトすることができません。	デフォルトで、SSO 統合は *URI* と呼ばれる URL パラメーターを使用して、IdP で認証した後のユーザーの誘導先を制御します。SSO は相対 URL を無視します。たとえば、SSO はユーザーを */ess* 相対 URL にリダイレクトできません。代わりに、ユーザーはディープリンク構文を使用する */nav_to.do?uri=/ess*などの URL に移動する必要があります。 ただし、この場合 ESS ポータルはメインナビゲーションコンテンツの IFrame 内に配置されます。つまり、サイトはページ全体を占有するのではなく、インスタンス内のページとしてロードされます。詳細については、CMS サイトとシングルサインオンを参照してください。 *view_content.do* を *active=false* に設定して CMS エントリページをプライベートにした場合、ディープリンクの動作には、インストレーションイグジットログインスクリプトのカスタマイズが必要です。URL の *URI* 部分を検索し、IdP での認証後にユーザーをリダイレクトするための相対 URL パスを含む *RelayState* URL パラメーターをビルドするスクリプトを作成します。
SAML が認証後にユーザーを適切なページにリダイレクトしません。	リレーステータスが IdP に渡され、認証中に返されるかどうかを確認します。これは、HTTP 要求ヘッダーと POST 情報を保存できるブラウザー (開発者ツールを備えた Chrome、または HTTPfox というアドオンを備えた Firefox など) を使用して実行できます。Internet Explorer の場合

その他のよくある問題 (続く)

エラーまたは症状	解決策
	は、Fiddler などのサードパーティアプリケーションを使用してください。その目的は、クライアント (ブラウザー) からインスタンスへ、またクライアントから IdP へ渡される要求を監視することです。

ログインアクティビティのイベントキューの監視

シングルサインオン統合はすべて、ログインアクティビティのイベントを作成します。

これらのイベントを使用してログインの失敗を監視し、対処すべきセキュリティ上の懸念があるかどうかを判断できます。

ログイン失敗のイベントキューの監視

イベント名	説明	レコード	パラメーター **1**	パラメーター **2**
external.authentication.succeeded	外部認証に成功し、ユーザーがインスタンス URL にアクセスしました。	セッション ID	正常にログインしたユーザーのユーザー ID	ユーザーがアクセスした URL (ディープリンクの場合あり)
external.authentication.failed	シングルサインオン要件がないか、指定されていません。		セッション ID	不足している認証要件
external.authentication.failed	ユーザー [sys_user] テーブルにユーザーが存在しません		ユーザー ID	文字列「ユーザーは存在しません」
external.authentication.failed	ユーザーはロックアウトされています。		ユーザー ID	文字列「ユーザーがロックアウトされました (User locked out.)」

イベントキューのログインイベント

SAML 2.0 統合は、ログインアクティビティのためのイベントを作成します。

これらのイベントを使用してログインの失敗を監視し、対処すべきセキュリティ上の懸念があるかどうかを判断できます。

ログインアクティビティイベント

イベント名	説明	イベントで使用される ID	イベント文字列
saml2.logout.validation.failed	IdP からのログアウト応答がログアウト要求の検証に失敗しました。このイベントは、<inResponseTo> 要素をセッション ID に対して検証します (<saml2p:LogoutRequest> 要素の ID 属性)。たとえば、発行されたログアウト要求のワークフローを参照してください。	セッション ID	文字列「SAML ログアウト応答の検証に失敗しました (SAML2 LogoutResponse validation failed)」
external.authentication.succeeded	外部認証に成功しました。		文字列「認証に成功しました (Authentication Successed)」
external.authentication.succeeded	外部認証に成功し、ユーザーがインスタンス URL にアクセスしました。	正常にログインしたユーザーのセッション ID とユーザー ID	ユーザーがアクセスした URL (ディープリンクの場合あり)
external.authentication.failed	シングルサインオン要件がないか、指定されていません。	セッション ID	不足している認証要件
external.authentication.failed	ユーザー [sys_user] テーブルにユーザーが存在しません。	ユーザー ID	文字列「ユーザーが存在しません」
external.authentication.failed	ユーザーはロックアウトされています。	ユーザー ID	文字列「ユーザーがロックアウトされました (User locked out)」

OAuth 受信と送信認証

OAuth ベースの認証は、認証プロトコルを使用してシステムで信頼を確立しようとするクライアントの ID を検証します。

OAuth 2.0 - オープン認証は認証の業界標準プロトコルであり、Web アプリケーション、デスクトップアプリケーション、およびモバイルデバイス向けの特定の認証フローを提供すると同時に、クライアント開発者の簡素化を実現します。

これは、Web サイトまたはアプリケーションがユーザーの代わりに他の Web アプリによってホストされているリソースにアクセスできるように設計された標準です。

リソースユーザーの認証情報を使用して保護されたリソースにアクセスする代わりに、クライアントはアクセストークンを取得します。ユーザーの承認を得てアクセストークンがサードパーティクライアントに発行され、クライアントはアクセストークンを使用して保護されたリソースにアクセスします。

チューリッヒから、次の拡張機能を使用して OAuth 統合を構成できます。

- Azure DevOps (ADO) などのサードパーティシステムのセキュリティ要件を満たすために、クライアントシークレットの長さを 2048 文字に増やします。
- JSON Web トークン (JWT) 署名検証の公開鍵を自動的に管理および更新するための JSON Web キーセット (JWKS) URL を指定します。
- 拡張セキュリティ (ES) アルゴリズムで署名された JWT 権限許可タイプを使用して、OAuth トークンを要求します。
- JWT トークンの一意の識別子を構成します。

受信

インスタンスにアクセスする外部クライアント用のエンドポイントを作成し ます。これにより、OAuth クライアントアプリケーションレコードが作成され、クライアントがインスタンスの制限付きリソースにアクセスするために必要なクライアント ID とクライアントシークレットが生成されます。詳細については、「OAuth 受信」を参照してください。

送信

インスタンスへのアクセスを承認するサードパーティ OAuth プロバイダーを使用します。別の OAuth プロバイダーに接続する場合は、OAuth プロファイルと OAuth スコープを指定します。詳細については、「OAuth 送信」を参照してください。

OAuth 2.0

OAuth 2.0 を使用すると、ユーザーは、各リソース要求でログイン認証情報を入力するのではなく、トークンを取得することで、外部クライアントを介してインスタンスリソースにアクセスできます。

OAuth 統合を管理するには、security_admin ロールが必要です。次のようなシナリオで OAuth 2.0 を設定します。

- **OAuth** 外部クライアントシナリオ (受信)：インスタンスは、インスタンスからデータをプルするためのサードパーティクライアント用のエンドポイントを提供します。
- **OAuth** プロバイダーシナリオ (送信)：インスタンスがサードパーティプロバイダーからデータをプルします。

注: トークンを初めて取得するには、ユーザー認証を行う必要があります。その後は、トークンの有効期限が切れるまでユーザーアカウントを使用して認証する必要はありません。

OAuth 2.0 は、シンプルなセキュリティフレームワークと高セキュリティフレームワークの両方でサポートされています。高セキュリティをお勧めします。高度なセキュリティが既に有効になっているバージョン、および高度なセキュリティをアクティブ化する方法について、情報を参照してください。

OAuth 2.0 実装の重要なコンセプト

コンセプト	説明
リソース所有者	保護されたリソースへのアクセスを許可できるエンティティ。リソース所有者が個人である場合は、エンドユーザーと呼ばれます。リソース所有者は常にユーザーアカウントです。
クライアント	リソース所有者の承認により、リソース所有者の代わりに保護されたリソースに対する要求を行うアプリケーション。
リソースサーバー	保護されたリソースをホストし、保護されたリソースへの要求を受け入れて応答できるサーバー。
認証サーバー	リソース所有者を正常に認証して認可を取得した後、クライアントにアクセストークンを発行するサーバー。
認証要求	保護されたリソースにアクセスするためにクライアントが必要とする権限。認証要求は常に、リソース所有者の代わりに動作するクライアントの ID と要求を許可する認証情報を含む HTTP POST メッセージです。
認証権限許可	リソースにアクセスするための、リソース所有者からの認可を表す認証情報。認証権限許可は、ユーザーのログイン認証情報またはリフレッシュトークンのいずれかです。
アクセストークン	保護されたリソースにアクセスするためにクライアントが使用する安全な文字列。インスタンスは、有効な認証権限許可を持つクライアントにアクセストークンを発行します。各アクセストークンには、特定のスコープ、有効期間、およびその他の属性が含まれます。 デフォルトで、インスタンスが OAuth プロバイダーであるシナリオでは、インスタンスは 30 分の有効期間でアクセストークンを発行します。サードパーティのトークンの場合は 30 日間です。
リフレッシュトークン	クライアントが追加のユーザー認証を要求せずに新しいアクセストークンを取得するために使用する認証情報。インスタンスは、初めてアクセストークンを持つことが許可されると、クライアントにリフレッシュトークンを発行します。 デフォルトで、インスタンスが OAuth プロバイダーであるシナリオでは、インスタンスは 100 日の有効期間でリフレッシュトークンを発行します。サードパーティのトークンの場合は 365 日間です。
自己署名証明書	ServiceNow AI Platform は自己署名証明書をサポートしていません。OAuth クライアントは証明書トラストストアを使用せず、自己署名証明書を組み込んだ OAuth エンドポイントへの接続を許可しません。
ユーザーエージェント	クライアントアプリケーション (多くの場合は Web サイト) へのアクセス権を委任するユーザー。アクセス権により、クライアントアプリケーションまたは Web サイトは、ユーザーがアクセス権を持つインスタンス内のデータへのアクセスを許可されます。ユーザーエージェントはシナリオで使用されます。

OAuth 権限許可タイプ

権限許可タイプは、クライアントがアクセストークンを取得する方法です。次の権限許可タイプがサポートされています。

- 認証コード：コンシューマーは最初に認証コードを取得し、それを使用してアクセストークンを取得します。OAuth プロファイルを指定 し、この権限許可タイプを指定することができます。認証コードを使用するプロセスは、*auth* コードフローまたは認証コードフローとも呼ばれます。
- リソース所有者のパスワード認証情報：リソースのコンシューマーは、アクセストークンを取得するためのユーザー認証情報をすでに持っています。このプロセスは、パスワードフローとも呼ばれます。
- クライアント認証情報：リソースのコンシューマーは、アプリケーションレジストリにすでに設定されているクライアント ID とクライアントシークレットを使用します。

認証情報のストレージ

OAuth クライアントシークレットは *password2* タイプフィールドとして格納され、KMF で暗号化されます。受信エンドポイント要求のチェックに使用されるユーザーパスワードは、ユーザーテーブルの *password* タイプフィールド (SHA 256) にハッシュ値として格納されます。この暗号化の詳細については、「KMF による Password2 暗号化」を参照してください。

OAuth の設定

OAuth を設定してアクティブ化し、OAuth システムプロパティを有効にし、外部クライアントアプリケーションがインスタンスにアクセスするための OAuth アプリケーションエンドポイントを作成し、OAuth パラメーターを設定します。

始める前に
必要なロール：admin

手順

1. OAuth プラグインがアクティブで、OAuth アクティブ化プロパティが true に設定されていることを確認します。
2. 次のいずれかの方法を使用して、OAuth アプリケーションレジストリを作成します。
   - インスタンスにアクセスする外部クライアント用のエンドポイントを作成し ます。これにより、**OAuth** クライアントアプリケーションレコードが作成され、クライアントがインスタンスの制限付きリソースにアクセスするために必要なクライアント ID とクライアントシークレットが生成されます。
   - インスタンスへのアクセスを承認するサードパーティ OAuth プロバイダーを使用します 。

     別の OAuth プロバイダーに接続する場合は、OAuth プロファイルを指定 し、OAuth スコープを指定 します。
3. OAuth トークンを要求する HTTP POST を作成するように、クライアントアプリケーションを設定します。
   アプリケーションは、返されたアクセストークンとリフレッシュトークンを使用するために、JSON 応答も解析できるようにする必要があります。

OAuth のアクティブ化

デフォルトでは、**OAuth 2.0 (com.snc.platform.security.oauth)** プラグインは新規インスタンスおよびアップグレードされたインスタンスでアクティブになっています。インスタンス上でプラグインがアクティブでない場合は、アクティブ化することができます。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > システムアプリケーション > 利用可能なすべてのアプリケーション > すべて.
2. フィルター基準と検索バーを使用してプラグインを検索します。

   名前または ID でプラグインを検索できます。プラグインが見つからない場合は、ServiceNow 担当者から要求する必要があります。
3. [インストール] を選択して、インストールプロセスを開始します。

   注: ドメインセパレーションと代理アドミンがインスタンスで有効になっている場合、管理ユーザーはグローバルドメインに含まれている必要があります。それ以外の場合、次のエラーが表示されます： 「別の操作が実行されているため、アプリケーションのインストールは利用できません： <プラグイン名> のプラグインの有効化 (Application installation is unavailable because another operation is running: Plugin Activation for <plugin name>)」

   インストールが完了するとメッセージが表示されます。プラグインとともにインストールされるコンポーネントの詳細については、「 アプリケーションとともにインストールされているコンポーネントの検索 」を参照してください。

OAuth プロパティの設定

登録済みアプリケーションに対して OAuth 2.0 トークンを生成するには、インスタンスに対して com.snc.platform.security.oauth.is.active プロパティがアクティブである必要があります。

始める前に
必要なロール：admin

手順

1. OAuth 2.0 を使用するには、ナビゲーターに 「sys_properties.list 」と入力し、 新規 を選択します。
   に移動して、システムプロパティリストを開くこともできます すべて > システムプロパティ > すべてのプロパティ > .
2. 次の設定をフォームに入力します。

   - 名前：com.snc.platform.security.oauth.is.active
   - タイプ：true | false
   - [値]：true

OAuth パスワードパラメーターの変更

このプロパティを使用して、サポートされているすべての権限許可タイプの入力として POST 本文パラメーターのみが受け入れられるようにします。

URI クエリーパラメーターを介して機密情報を送信すると、要求間でクライアント、サーバー、またはホストによる機密情報の漏洩につながる可能性があります。Madrid リリース以降、この新しいプロパティにより、サポートされるすべての権限許可タイプに対して、POST 本文パラメーターのみが入力として受け入れられることが保証されています。サポートされている権限許可タイプは次のとおりです。

- 認証コード
- パスワード

- クライアント認証情報
- リフレッシュトークン

OAuth パスワードパラメーターのプロパティ

プロパティ	説明
glide.oauth.allow.parameters.in.post.body.only	このプロパティは、OAuth 2.0 プラグインの一部として、zBoot にのみ true に設定されます。インスタンスにこの設定が必要な場合は、プロパティを作成して true に設定します。

OAuth 受信

OAuth 受信認証により、信頼できる外部アプリケーションが ServiceNow API に安全にアクセスでき、制御および許可された接続が確保されます。

システムで OAuth 統合を構成または管理するには、次のいずれかのロールが必要です。

- oauth_admin
- mi_admin
- admin（アドミン）

受信認証により、サードパーティシステムや他の ServiceNow® インスタンスなどの外部アプリケーションが ServiceNow API に安全に接続できます。受信認証により、信頼できるクライアントのみが制御された安全な方法で ServiceNow インスタンスにアクセスできることを確認します。ServiceNow は、それぞれが特定の統合シナリオ向けに設計されたいくつかの OAuth 2.0 権限許可タイプをサポートしています。次の情報を使用して、ユースケースに最適な権限許可タイプを選択してください。

認証コード権限許可

理想的な使用シナリオ	機能
ユーザーの同意を得て、ユーザーの代わりにユーザーデータにアクセスする必要があるアプリケーション。 例:ユーザーの代わりに動作する Web、モバイル、またはデスクトップアプリケーション。	ユーザーがクライアントアプリケーションからログインプロセスを開始すると、 ServiceNow ログインページにリダイレクトされます。ユーザーがログインして同意すると、クライアントアプリケーションは認証コードを受け取ります。クライアントアプリケーションは、アクセストークンの認証コードを ServiceNow インスタンスと交換します。 認証コード権限許可は、ユーザー向け統合で最も安全で広く使用されているワークフローです。機密クライアント (クライアントシークレットを使用) と、Proof Key for Code Exchange (PKCE) を使用するパブリッククライアントの両方をサポートします。

認証コード権限許可のワークフローと構成の詳細については、次を参照してください。 認証コードの許可

クライアント認証情報の権限許可

理想的な使用シナリオ	機能
バックエンドサービスや自動システム統合など、ユーザーの関与なしに ServiceNow API にアクセスする必要があるクライアントアプリケーション。	クライアントアプリケーションは、独自の認証情報 (クライアント ID とシークレット) を使用して、 ServiceNow インスタンスで直接認証します。認証されると、アプリケーションは ServiceNow API にアクセスするためのアクセストークンを受け取ります。

クライアント認証情報権限許可のワークフローと構成の詳細については、「 クライアント認証情報の権限許可」を参照してください。

サードパーティ **ID** トークンフロー

理想的な使用シナリオ	機能
ServiceNow が Azure AD や Okta などの外部 ID プロバイダーによって発行された ID トークンまたはアクセストークンを信頼するフェデレーション認証シナリオ。	クライアントアプリケーションは、信頼できるサードパーティの ID プロバイダーから ID またはアクセストークンを取得し、 ServiceNow インスタンスに API 要求を行うときに認証ヘッダーに含めます。 ServiceNow トークンを検証し、信頼されている場合は、アサートする ID に基づいてアクセス権を付与します。これにより、システム間でシームレスなシングルサインオン (SSO) とフェデレーション認証が可能になります。

サードパーティトークンのフローと構成の詳細については、「サードパーティトークンの付与」を参照してください。

JWT ベアラー助成金

理想的な使用シナリオ	機能
ユーザーの操作や共有シークレットの保存を必要とせずに、ユーザーの代わりに、またはユーザー自身として、 ServiceNow リソースへの安全なアクセスを必要とするクライアントアプリケーション。 クライアント アプリケーションは、それが表すユーザーやシステムなど、ID 関連の要求を含む署名付き JSON Web トークン (JWT) を作成します。次に、それを ServiceNow インスタンスに提示して、アクセストークンを要求します。	ユーザーの代理として行動する場合: トークンは、以前に認証されたユーザーを表します。これにより、ユーザーに認証情報や同意を再度求めることなく、安全でシームレスなアクセスが可能になります。署名済みトークンはユーザーの ID をアサートし、 ServiceNow リアルタイムのユーザー操作を必要とせずに要求を信頼できるようにします。 それ自体として行動する場合: トークンは、クライアントアプリケーションを直接識別して認証します。共有シークレットを使用する代わりに、クライアントは秘密鍵を使用してトークンに署名し、クライアント認証情報の付与に代わるより安全な代替手段にします。

JWT ベアラー権限許可のワークフローと構成の詳細については、「 JSON Web トークンベアラー権限許可」を参照してください。

リソース所有者のパスワード認証情報の付与

理想的な使用シナリオ	機能
アプリがユーザーの認証情報を直接収集する、制御された環境にある、信頼性の高い内部クライアントアプリケーション。	クライアントアプリケーションは、ユーザーのユーザー名とパスワードを収集し、それらを ServiceNow インスタンスにリダイレクトしてアクセストークンを取得します。ワークフローはリダイレクト画面と同意画面をバイパスしますが、ユーザー認証情報をクライアントアプリケーションに公開します。ServiceNow では、リソース所有者のパスワード認証情報の付与は、従来の環境または制御された環境でのみ実装することをお勧めします。

リソース所有者のパスワード認証情報付与のワークフローと構成の詳細については、「 リソース所有者のパスワード認証情報の付与」を参照してください。

暗黙的な権限許可

理想的な使用シナリオ	機能
従来のシングルページアプリケーション (SPA) またはブラウザーベースのアプリでは、クライアントシークレットを安全に保存できず、軽量のクライアント側認証フローが必要です。	ユーザーはブラウザーからログインします。クライアントアプリケーションは、ログイン後に中間認証コードのステップをバイパスして、URL で直接アクセストークンを受信します。 このフローはもともと、シークレットの安全な保存が不可能な、完全にブラウザで実行されるパブリッククライアント向けに設計されました。実装は簡素化されますが、ブラウザにトークンが公開されるため、傍受のリスクが高まります。セキュリティを強化するために、 ServiceNow は PKCE を使用して認証コード付与を実装することを推奨しています。

暗黙的な権限許可のワークフローと構成の詳細については、「 OAuth の暗黙的な権限許可」を参照してください。

OAuth スコープ

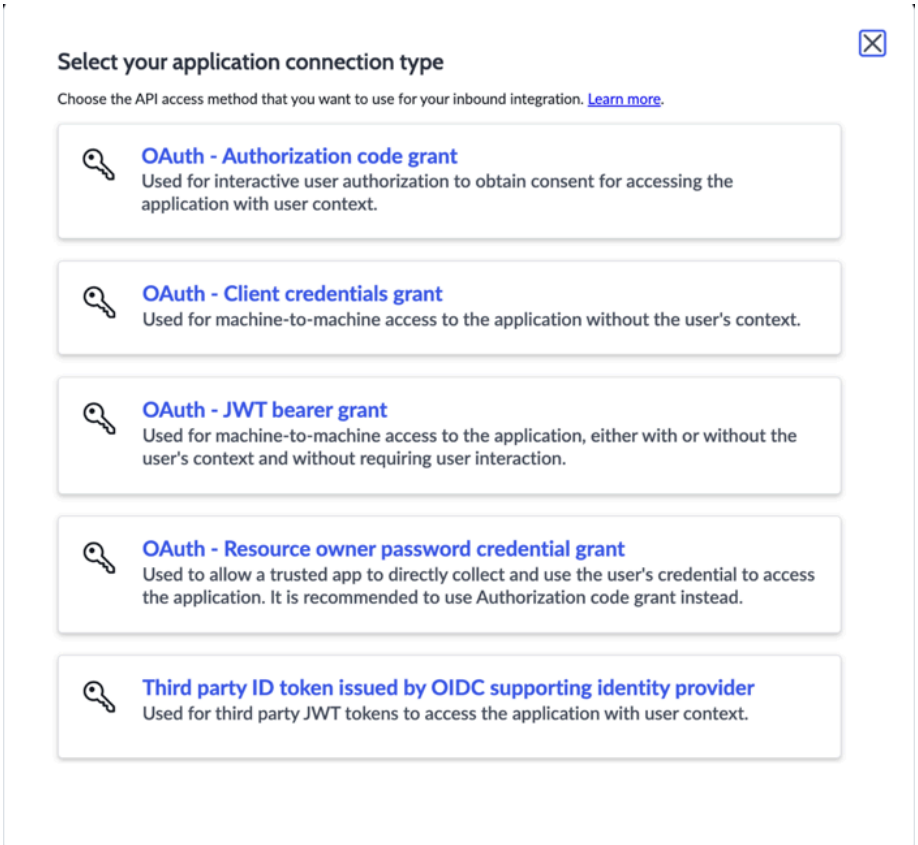
REST API の OAuth 認証スコープのサポートを詳しく調べることができます。OAuth スコープは、特定の REST API へのアクセスのみを提供します。詳細については、「REST API 認証スコープ」を参照してください。

新しいインバウンド統合エクスペリエンス

ServiceNow マシン ID コンソールの新しいインバウンド統合ワークフローは、インバウンド統合を管理するためのエクスペリエンスを強化します。

インバウンド統合の方法

インスタンスで、次に移動します: すべて > マシン **ID** コンソール > インバウンド統合 > 新しい統合. アプリケーション接続タイプ (権限許可タイプ) を選択します。



さまざまな権限許可タイプとその設定方法の詳細については、次のトピックを参照してください。

- 認証コードの許可
- クライアント認証情報の権限許可
- JSON Web トークンベアラー権限許可
- リソース所有者のパスワード認証情報の付与
- サードパーティトークンの付与

認証コードの許可

OAuth 認証コード権限許可は、ユーザーの同意を得てユーザーデータにアクセスする Web、モバイル、またはデスクトップアプリで広く使用されている安全で広く使用されているフローです。プライベートクライアント (クライアントシークレットを使用) とパブリッククライアント (PKCE を使用) の両方をサポートします。

認証コード権限許可フローでは、 ServiceNow は認証サーバー (ユーザー認証とトークン発行の処理) とリソース サーバー (API をホストする) の両方として機能します。SSO が有効になっている場合、 ServiceNow は認証のために構成済みの ID プロバイダー (IdP) にユーザーをリダイレクトします。IdP がユーザーを正常に認証すると、コントロールが ServiceNow に戻り、認証コードが発行されます。このプロセスにより、外部認証を使用しても、トークンを発行して API アクセスを管理するための権限 ServiceNow 維持されます。

注: 独自の ID プロバイダー (Azure AD や Okta など) を認証サーバーとして使用する場合は、 サードパーティトークンの付与 フローの使用を検討してください。

関連トピック

認証コード権限許可ワークフロー

OAuth 認証コード権限許可の構成

認証コード権限許可ワークフロー

OAuth 認証コード権限許可は、ユーザーの同意を得てユーザーデータにアクセスする Web、モバイル、またはデスクトップアプリで広く使用されている安全で広く使用されているフローです。プライベートクライアント (クライアントシークレットを使用) とパブリッククライアント (PKCE を使用) の両方をサポートします。

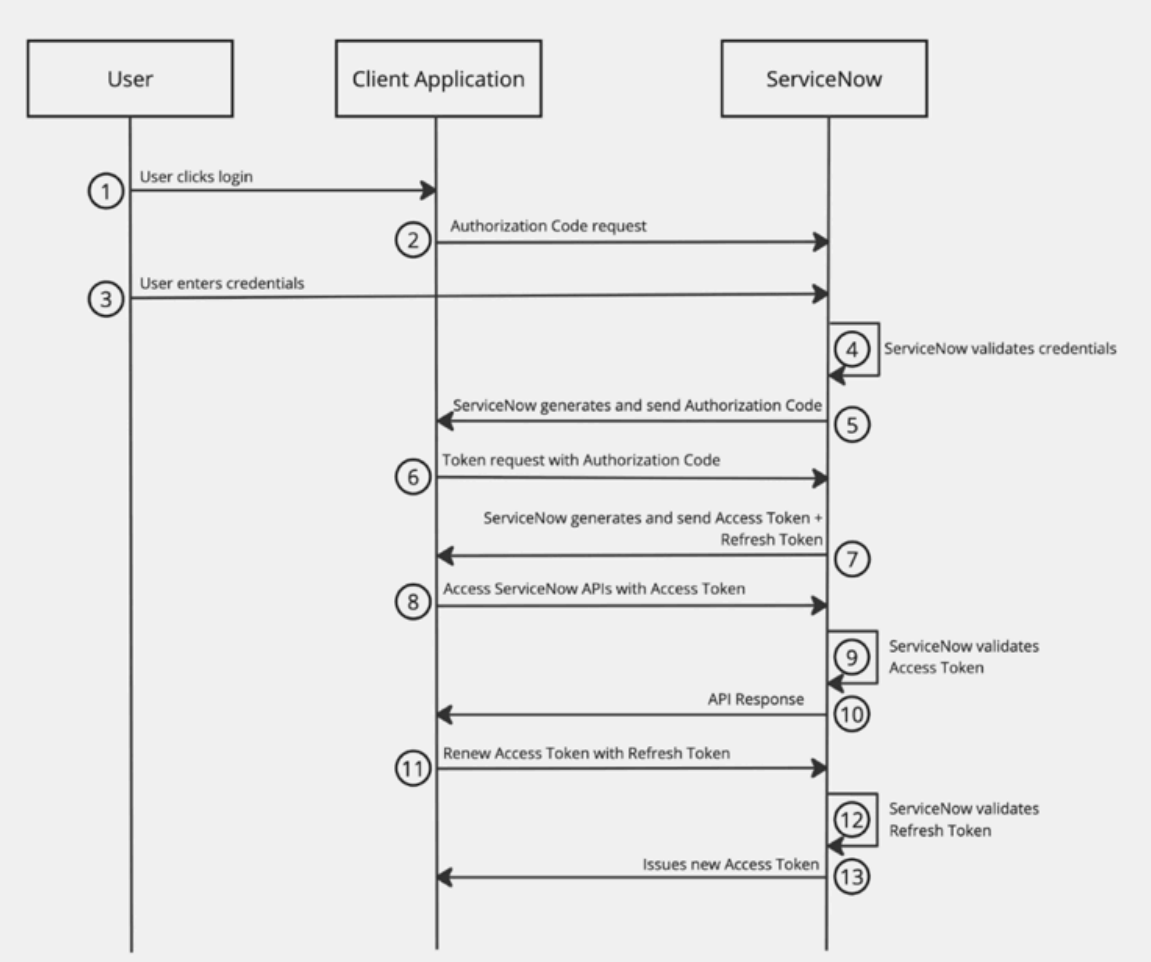
始める前に

必要なロール: oauth_admin、mi_admin、admin

このタスクについて

このトピックコレクションでは、クライアントアプリケーションが認証コード権限許可フローを使用して ServiceNow からトークンを取得し、そのトークンを使用して API 呼び出しを行う方法について説明します。プライベートクライアントはクライアントシークレットを使用し、パブリッククライアントは PKCE コードチャレンジを使用します。

認証ワークフロー



手順

1. クライアントアプリケーションからログインします。

   ユーザーは、クライアントアプリケーションインターフェイスからログインプロセスを開始します。

2. 認証要求を開始します。

クライアントは、ユーザーを ServiceNow 認証エンドポイントにリダイレクトします。認証要求を開始する方法は、クライアントのタイプ (パブリックまたはプライベート) によって異なります。

公開クライアント

パブリッククライアント (例:モバイルアプリケーションまたはシングルページアプリケーション) は、クライアントシークレットを安全に保存できません。したがって、セキュリティを強化するために コード交換の証明キー (PKCE) を使用する必要があります。

- 認証要求に PKCE コードチャレンジ を含め、 コードチャレンジメソッドを指定します。
- トークン要求中に、クライアントは コード検証ツール を送信して認証コードを検証する必要があります。

次のパラメーターを使用して、認証エンドポイントへの GET 要求を実行します。

```
Method: GET
Endpoint: https://<servicenow_base_url>/oauth_auth.do
```

認証要求パラメーター (パブリッククライアント - **PKCE)**

パラメーター	必須	説明
response_type	はい	認証コードフローを開始するには、 値を code に設定します。
client_id	はい	クライアントアプリケーションの一意の識別子。 形式:YOUR_CLIENT_ID
redirect_uri	はい	ServiceNow が認証コードを送信する URI。 例:https://yourapp.com/callback
code_challenge	はい	コード検証ツールの base64url でエンコードされた SHA-256 ハッシュ。これは、PKCE フローの一部として使用されます。
code_challenge_method	はい	コードチャレンジに使用する変換方法を指定します。S256に設定します。
スコープ	オプション	要求されたスコープのスペース区切りリスト。 例: incident_read incident_write。
state	はい	CSRF 攻撃を回避するために使用されるクライア

パラメーター	必須	説明
		ント生成値。値は変更されずにリダイレクト URI で返され、クライアントはそれを検証できます。

注: Madrid リリース以降、システムプロパティ glide.oauth.state.parameter.required により、OAuth 要求で state パラメーターの使用が義務付けられています。state プロパティは、新しいインスタンスではデフォルトで true に設定され、アップグレードされたインスタンスではオプションです。ステータスパラメーターがない場合、認証要求は失敗し、次のエラーが表示されます:要求にステータスパラメーターがありません。

プライベートクライアント

プライベートクライアント (例:サーバー側アプリケーション) は、クライアントシークレットを安全に保存でき、PKCE は必要ありません。

- 認証要求は、ユーザーを認証エンドポイントにリダイレクトすることによって開始されます。この手順では、クライアントシークレットまたは PKCE コードのチャレンジは必要ありません。
- トークン要求時に、クライアントはアクセストークンを取得するための認証コードとともにクライアントシークレットを含めます。

次のパラメーターを使用して、認証エンドポイントへの GET 要求を実行します。

```
Method: GET
Endpoint: https://<servicenow_base_url>/oauth_auth.do
```

認証要求パラメーター (プライベートクライアント/クライアントシークレット)

パラメーター	必須	説明
response_type	はい	認証コードフローを開始するには、 値を code に設定します。
client_id	はい	クライアントアプリケーションの一意の識別子。 形式:YOUR_CLIENT_ID
redirect_uri	はい	ServiceNow が認証コードを送信する URI。 例:https://yourapp.com/callback
スコープ	オプション	要求されたスコープのスペース区切りリスト。 例: incident_read incident_write。
state	はい	クロスサイトリクエストフォージェリ (CSRF) 攻撃を回避するために使用されるクライアント生成

パラメーター	必須	説明
		値。値は変更されずにリダイレクト URI で返され、クライアントはそれを検証できます。

3. ログインし、クライアントアプリケーションにアクセス同意を付与します。
   ServiceNow (SSO が有効になっている場合は IdP) にログインし、クライアントアプリケーションにアクセスの同意を付与します。
4. ServiceNow (SSO が有効な場合は IdP) が認証情報を検証し ServiceNow クライアントに認証コードを返します。

   認証が成功すると、ブラウザーは redirect_uriにリダイレクトされ、認証コードがクエリ文字列に含まれます。

   ```
   https://yourapp.com/callback?code=AUTH_CODE&state=xyz123
   ```

5. 認証要求を開始します。

   クライアントは、アクセストークンを取得するためにユーザーを ServiceNow 認証エンドポイントにリダイレクトします。認証要求を開始する方法は、クライアントのタイプ (パブリックまたはプライベート) によって異なります。

   公開クライアント

   パブリッククライアント (例:モバイルアプリケーションまたはシングルページアプリケーション) は、クライアントシークレットを安全に保存できません。したがって、セキュリティを強化するために コード交換の証明キー (PKCE) を使用する必要があります。

   - 認証要求に PKCE コードチャレンジ を含め、 コードチャレンジメソッドを指定します。
   - トークン要求中に、クライアントは コード検証ツール を送信して認証コードを検証する必要があります。

   クライアントは、次のパラメーターを使用してトークンエンドポイントに POST 要求を送信します。

   ```
   Method: POST
   Endpoint: https://<servicenow_base_url>/oauth_token.do
   Headers: Content-Type: application/x-www-form-urlencoded
   ```

   トークン要求パラメーター **(public client-PKCE)**

パラメーター	必須	説明
grant_type	はい	コードをトークンと交換するには、値を authorization_code に設定します。
コード	はい	認証エンドポイントから受信した認証コード。
redirect_uri	はい	最初の認証要求で使用される URI。

パラメーター	必須	説明
		例:https://yourapp.com/callback
client_id	はい	クライアントアプリケーションの一意の識別子。
code_verifier	はい	PKCE code_challengeの生成に使用された元の文字列。
state	はい	CSRF 攻撃を防止するために使用される、クライアント生成の値。値は変更されずにリダイレクト URI で返され、クライアントはそれを検証できます。

プライベートクライアント

プライベートクライアント (例:サーバー側アプリケーション) は、クライアントシークレットを安全に保存でき、PKCE は必要ありません。

- 認証要求は、ユーザーを認証エンドポイントにリダイレクトすることによって開始されます。この手順では、クライアントシークレットまたは PKCE コードのチャレンジは必要ありません。
- トークン要求時に、クライアントはアクセストークンを取得するための認証コードとともにクライアントシークレットを含めます。

次のパラメーターを使用して、認証エンドポイントへの POST 要求を実行します。

```
Method: POST
Endpoint: https://<servicenow_base_url>/oauth_token.do
Headers: Content-Type: application/x-www-form-urlencoded
```

トークン要求パラメーター (プライベートクライアント/クライアントシークレット)

パラメーター	必須	説明
grant_type	はい	コードをトークンと交換するには、値を authorization_code に設定します。
コード	はい	認証エンドポイントから受信した認証コード。
redirect_uri	はい	最初の認証要求で使用される URI。 例:https://yourapp.com/callback
client_id	はい	クライアントアプリケーションの一意の識別子。

パラメーター	必須	説明
client_secret	はい	トークンエンドポイントでの認証に使用されるクライアントのシークレット。
state	はい	CSRF 攻撃を防止するために使用される、クライアント生成の値。値は変更されずにリダイレクト URI で返され、クライアントはそれを検証できます。

6. アクセストークンを使用して ServiceNow API にアクセスします。

   例：

   アクセストークンを使用して、API への GET 要求を行います。Authorization ヘッダーにアクセストークンを含めます。

```
Method: GET
End Point: https://<servicenow_base_url>/api/now/incident
Authorization: Bearer YOUR_ACCESS_TOKEN
```

7. アクセストークンが期限切れになっている場合は更新します。
   次のパラメーターを使用して、アクセストークンを更新する POST 要求を行います (プライベートクライアントのみ)。

```
Method: POST
Endpoint: https://<servicenow_base_url>/oauth_token.do
Headers: Content-Type: application/x-www-form-urlencoded
```

リフレッシュトークン要求パラメーター (プライベートクライアント)

パラメーター	必須	説明
grant_type	はい	新しいアクセストークンを要求するには、値を refresh_token に設定します。
refresh_token	はい	トークンエンドポイントによって以前に発行されたリフレッシュトークン。
client_id	はい	クライアントアプリケーションの一意の識別子。
client_secret	はい	トークンエンドポイントでの認証に使用されるクライアントシークレット。

OAuth 認証コード権限許可の構成

OAuth 認証コード権限許可を設定して、安全でインタラクティブなユーザー認証を有効にし、アプリケーションがユーザーの代わりにリソースにアクセスできるようにします。OAuth 認証コードの権限許可は、ユーザー ID と権限に基づいて API アクセスが許可されていることを確認します。

始める前に
必要なロール: oauth_admin、mi_admin、admin

手順

1. 移動先 マシン **ID** コンソール > インバウンド統合 > 新しい統合 > > **OAuth** 認証コード権限許可. [新規レコード] ページが表示されます。
2. [詳細] フォームのテキストフィールドを適切な情報で更新します。

[詳細] フォーム

フィールド	説明
OAuth エンティティの名前	OAuth エンティティの名前。
リダイレクト **URL**	認証後に認証コードを送信する URL。
クライアント **ID**	アプリケーションを識別するためにアサインされた一意の ID。
クライアントシークレット	アプリケーションと認証サーバーのみが識別できる秘密キー。アプリケーションはこのキーを使用して、アクセストークンを認証して取得します。

アプリケーションが認証情報を安全に保存できず、承認時に ID を証明するために秘密キーを必要としない場合は、[これはパブリッククライアント です] チェックボックスをオンにします。クライアントシークレット情報は、パブリッククライアントに対して処理されます。

3. [詳細オプション (オプション)] フォームのテキストフィールドを適切な情報で更新します。

詳細オプションフォーム

フィールド	説明
アクセストークンの有効期間	OAuth アクセストークンが期限切れになるまで有効な期間 (秒)。 注: デフォルト値は 1800 秒です。
リフレッシュトークンの有効期間	OAuth リフレッシュトークンが期限切れになるまで有効である期間 (秒)。 注: デフォルト値は 8,640,000 秒です。
ログイン **URL**	認証サーバーで認証するための HTTP リダイレクトエンドポイント。
ロゴ **URL**	認証および認可プロセス中のアプリケーションを表す画像の Web アドレス。これは認証サーバーの同意画面に表示され、要求元のアプリケーションを認識するのに役立ちます。

トークン制限を適用すると、OAuth アクセストークンの使用方法に制限が適用され、トークンが特定の条件下でのみ有効であることを検証することでセキュリティが強化されます。[トークン制限の適用] チェックボックスをオンにして、OAuth アクセストークンを API アクセスポリシーで

自動翻訳

定義された特定の API に制限します。[トークン制限の適用] がオフになっている場合、トークンは他の REST API で使用できます。

4. [認証スコープ (オプション)] フォームのテキストフィールドを適切な情報で更新します。認証スコープは、アプリケーションがリソースに対して持つアクセスのレベルを定義します。アクセスする特定の REST API の認証スコープを選択します。

認証スコープフォーム

フィールド	説明
認証スコープ	アプリケーションがリソースに対して持つアクセスのレベル。認証スコープは、アクセストークンが API またはデータに対して実行できるアクションを制限します。
制限認証	認証を制限する API の名前。

5. [新しい認証スコープの作成] を選択して、新しい認証スコープを追加します。
6. **[Save** (保存)] を選択します。
   新しい OAuth 認証コード権限許可が作成されます。
7. 検索項目 すべて > インバウンド統合 > アプリケーションレジストリ をクリックして、新しく作成された OAuth 認証コード権限許可を表示します。

クライアント認証情報の権限許可

ユーザーの操作なしで ServiceNow® API にアクセスするバックエンドサービスまたは自動統合には、OAuth クライアント認証情報の権限許可タイプを使用します。クライアント アプリケーションは、クライアント ID とシークレットを使用して直接認証し、ユーザーではなくアプリケーション自体を表すアクセストークンを受け取ります。

関連トピック

- クライアント認証情報権限許可ワークフロー
- OAuth クライアント認証情報の権限許可の構成

クライアント認証情報権限許可ワークフロー

クライアント認証情報ワークフローを使用してクライアントアプリケーションを認証します。クライアント認証情報権限許可ワークフローは、バックエンドサービスまたはシステム統合によって、ユーザーの関与なしに ServiceNow® API にアクセスするために使用されます。

始める前に

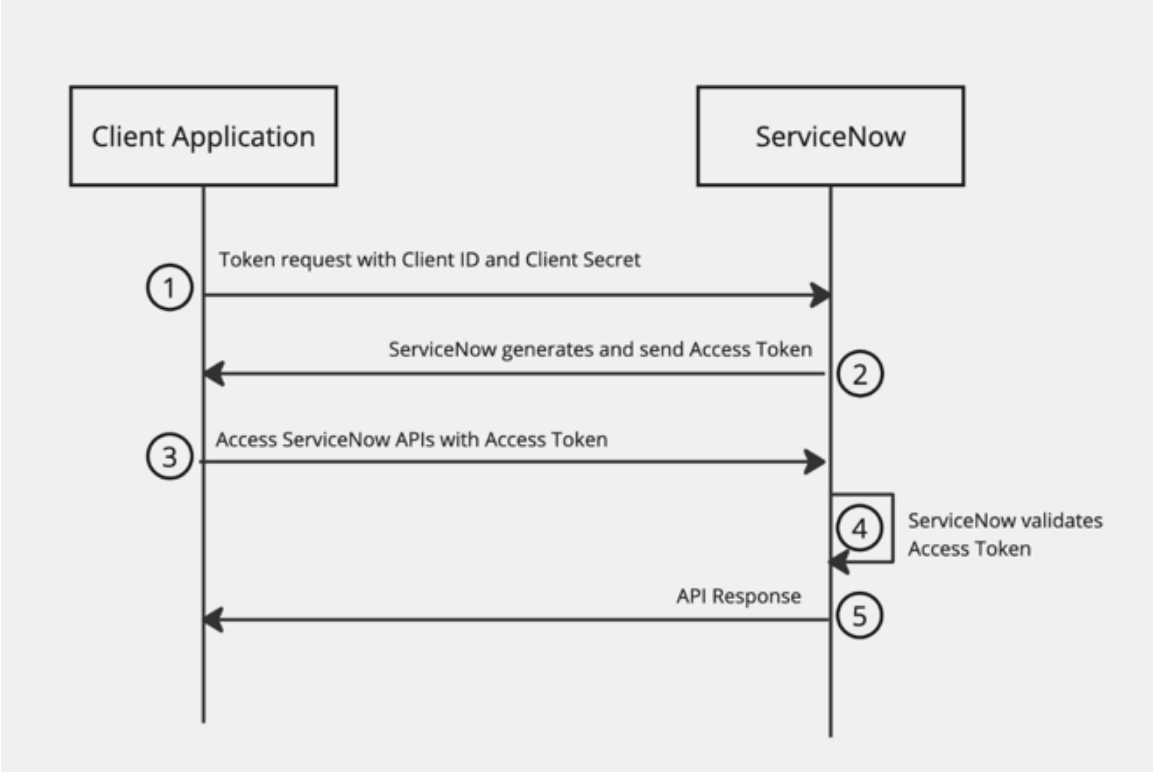
必要なロール: oauth_admin、mi_admin、admin

このタスクについて

このワークフローでは、クライアントアプリケーション (バックエンドサービスまたはシステム統合) が、ユーザーの操作なしでクライアント認証情報を使用して ServiceNow で直接認証する方法について説明します。アプリケーションは、クライアント ID とクライアントシークレットを使用してアクセストークンを要求し、トークンを発行する前に ServiceNow 検証します。その後、クライアントはこのトークンを使用して ServiceNow API にアクセスします。 ServiceNow 、適切な応答を返す前に、各要求を検証します。

自動翻訳

クライアント認証情報権限許可ワークフロー



手順

1. クライアントアプリケーションは、 次のパラメーターを使用して ServiceNow エンドポイントに対してトークン要求を行います。

```
Method: POST
Endpoint: https://<servicenow_base_url>/oauth_token.do
```

トークン要求パラメーター

パラメーター	必須	説明
grant_type	はい	OAuth 2.0 権限許可タイプ。 例: client_credentials
client_id	はい	クライアントアプリケーションの一意の識別子。 例:YOUR_CLIENT_ID
client_secret	はい	クライアント ID に関連付けられたシークレット。 例:YOUR_CLIENT_SECRET
スコープ	オプション	アクセストークンに対して要求された権限。 例: incident_read incident_write

2. ServiceNow は認証情報を検証し、アクセストークンを返します。
3. アクセストークンを使用して API 要求を行います。

   各 API 要求の 認証 ヘッダーにアクセストークンを含めます。

```
Method: POST
Endpoint: https://<servicenow_base_url>/api/now/incident
Authorization: Bearer YOUR_ACCESS_TOKEN
```

4. ServiceNow トークンを検証し、適切な API 応答を返します。

> **注:** クライアント認証情報権限許可ワークフローは、信頼できるサーバー側アプリケーションでのみ使用してください。client_secretを安全にメンテナンスしてください。ブラウザーやモバイルアプリなどのクライアント側環境で client_secret を使用しないようにします。

OAuth クライアント認証情報の権限許可の構成

ユーザーの操作なしで安全なマシン間認証を行うための OAuth クライアント認証情報権限許可を構成します。クライアント認証情報を使用してアプリケーションを認証し、スコープ対象の権限で制御された API アクセスを付与します。

始める前に

必要なロール: oauth_admin、mi_admin、admin

手順

1. 移動先 マシン **ID** コンソール > > インバウンド統合 > > 新しい統合 > **OAuth** クライアント認証情報の権限許可.
   OAuth クライアント認証情報の設定ページが表示されます。
2. [詳細] フォームのテキストフィールドを適切な情報で更新します。

   [詳細] フォーム

フィールド	説明
Name (名前)	OAuth エンティティの名前。
OAuth アプリケーションユーザー	アプリケーションが OAuth を介して認証および認可するサービスアカウントのユーザー名。
クライアント **ID**	アプリケーションを識別するためにアサインされた一意の ID。
クライアントシークレット	アプリケーションと認証サーバーのみが識別できる秘密キー。アプリケーションはこのキーを使用して、アクセストークンを認証して取得します。

   [**Active** (有効)] チェックボックスをオンにします。
3. [詳細オプション (オプション)] フォームのテキストフィールドを適切な情報で更新します。
4. [認証スコープ (オプション)] フォームのテキストフィールドを適切な情報で更新します。
   認証スコープは、アプリケーションがリソースに対して持つアクセスのレベルを定義します。アクセスする特定の REST API の認証スコープを選択します。

認証スコープフォーム

フィールド	説明
認証スコープ	アプリケーションがリソースに対して持つアクセスのレベル。認証スコープは、アクセストークンが API またはデータに対して実行できるアクションを制限します。
制限認証	認証を制限する API の名前。

a. [新しい認証スコープの作成] を選択して、新しい認証スコープを追加します。

5. **[Save** (保存)] を選択します。
   新しい OAuth クライアント認証情報の権限許可が作成されます。
6. 検索項目 すべて > インバウンド統合 > アプリケーションレジストリ をクリックして、新しく作成されたクライアント認証情報の権限許可を表示します。

サードパーティトークンの付与

サードパーティのトークン権限許可により、 ServiceNow® Azure AD や Okta などの信頼できる外部 ID プロバイダーから ID トークンを受け入れることができます。サードパーティのトークン権限許可は、安全なトークンベースのアクセスを提供します。この方法は、フェデレーション認証シナリオで安全なアクセスとシングルサインオン (SSO) をサポートします。

クライアントアプリケーションは、Azure AD や Okta® などの信頼できる外部 ID プロバイダーに ID またはアクセストークン®を要求し、ServiceNow® への API 要求の Authorization ヘッダーに含めます。ServiceNow トークンを検証し、信頼されている場合は、アサーションされた ID に基づいてアクセス権を付与します。

サードパーティの ID プロバイダー (IdP) のアカウントを使用して、次の目的で ServiceNow® API にアクセスできます。

- ユーザーアカウントのサードパーティトークンのワークフロー
- サービスアカウントのサードパーティトークンワークフロー

ユーザーアカウントのサードパーティトークンのワークフロー

このワークフローは、トークンフェデレーションの概念に基づいています。これにより、クライアントアプリケーションは IdP から直接トークンを取得し、それを使用して ServiceNow API にアクセスできるようになります。

始める前に

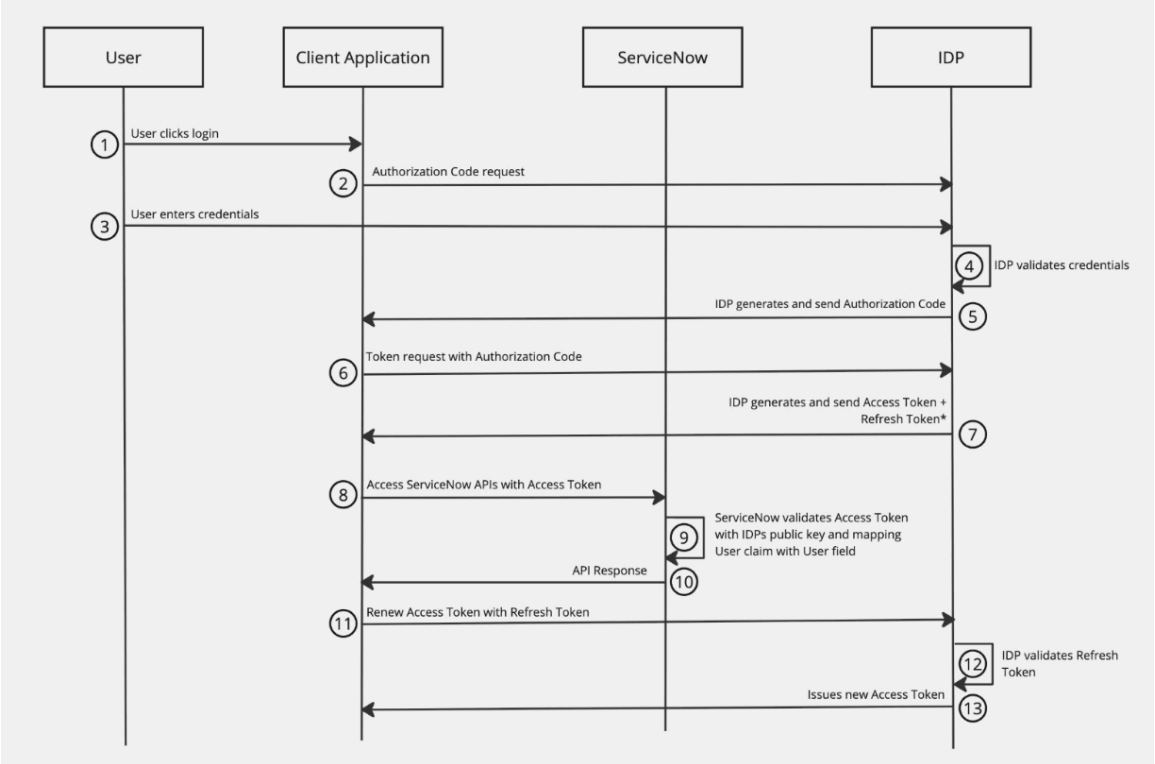
必要なロール: oauth_admin、mi_admin、admin

このタスクについて

サードパーティクライアントアプリケーションは、ID プロバイダー (IdP) に直接トークンを要求します。クライアントと IdP 間の認証方法は柔軟で、特定の要件を満たすように構成できます。認証に成功すると、IdP は ID トークンまたはアクセストークンを発行し、オプションでリフレッシュトークンを発行します。これらのトークンはクライアントアプリケーションに直接送信され、クライアントアプリケーションはそれらを使用して ServiceNow API にアクセスします。

注: ServiceNow は、セットアップ中に構成された公開鍵を使用してトークンを検証し、要求された API へのアクセスを許可します。トークンが JSON Web トークン (JWT) 形式であることを確認します。

ユーザーアカウントのワークフロー



注: この図は説明のためのものです。クライアントアプリケーションと ID プロバイダー間の認証コード権限許可フローが表示されます。ワークフローは柔軟です。要件に基づいて別のフローを使用できます。

手順

1. サードパーティクライアントアプリケーションを構成します。

   ID プロバイダー (IdP) から直接トークンを要求するようにサードパーティクライアントアプリケーションを設定します。セキュリティと統合の要件に最も適した認証方法を選択します。

2. ServiceNow で OAuth クライアントを作成します。

   ID プロバイダー (IdP) からの受信トークンの検証を有効にするために必要な詳細を入力します。構成方法の詳細については、次を参照してください。 サードパーティ ID トークンの構成

サービスアカウントのサードパーティトークンワークフロー

ユーザーアカウントのサードパーティトークンワークフローを作成する手順を実行します。さらに、サービスアカウントに対応する sys_user アカウントを ServiceNow に作成します。通常、ServiceNow に既存のレコードを持つ人間のユーザーとは異なり、サービスアカウントはアプリケーション固有であり、プラットフォームで有効な ID を確立するには手動で作成する必要があります。

始める前に

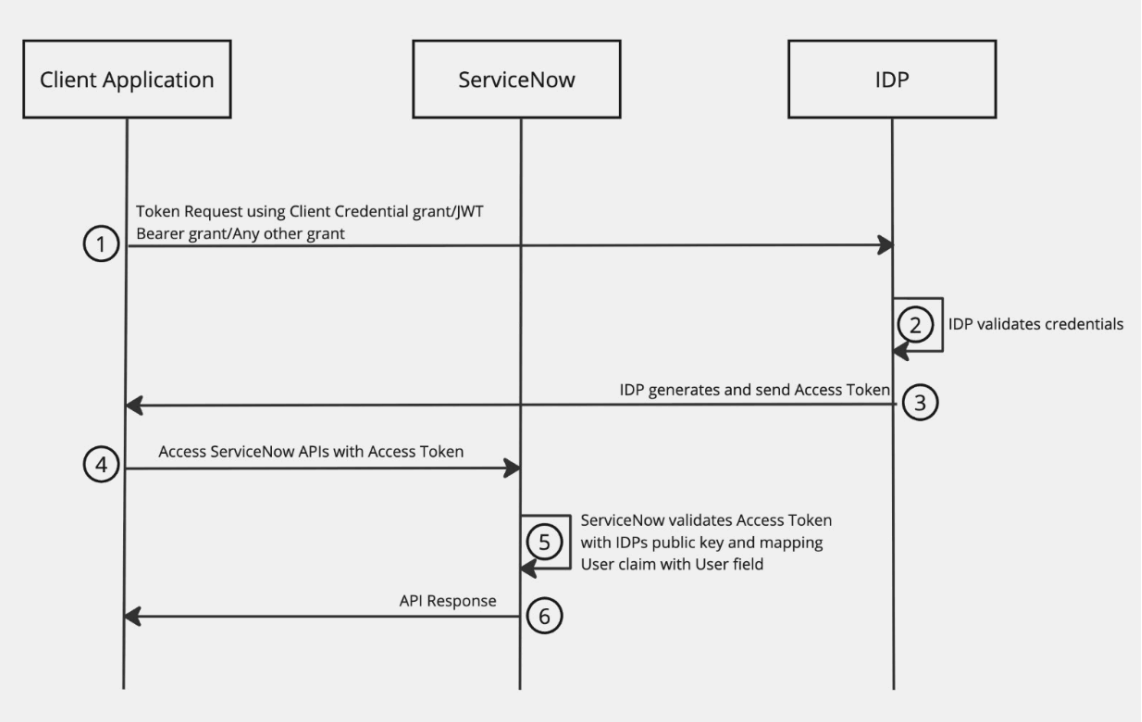
必要なロール: oauth_admin、mi_admin、admin

このタスクについて

サードパーティアプリケーションが外部 ID プロバイダー (IdP) からのトークンを使用して認証する場合、 ServiceNow ID をマッピングしてアクセス制御を適用するための対応するユーザーレコードが必要です。

IdP によって発行されたトークンの初期セットアップ中に構成された要求の値は、そのセットアップ中に指定されたユーザーフィールドにマッピングされます。このアカウントは、ServiceNow のサービス ID を表します。このアカウントを API アクセスのみに制限し、適切なロールとグループを追加して必要な権限を割り当てることができます。

サービスアカウントワークフロー



手順

1. ユーザーアカウントの ID トークンフローセットアップに記載されている手順を実行します。
2. サービスアカウント ID を表す sys-user アカウントを ServiceNow に作成します。トークン要求値が、ユーザーレコード内のマッピングされたユーザーフィールド (user_name やメールなど) の値と一致することを確認します。例:user_name、メール。
   a. [Web サービスへのアクセスのみ] オプションを選択して、アカウントを API アクセスに制限します。
   b. 必要なロールとグループをアサインして、適切な権限を付与します。

   ServiceNowプラットフォームは、構成された要求をsys_userレコードの指定されたユーザーフィールドにマッピングします。そのユーザーにアサインされたロールとグループに基づいてアクセス権が適用されます。
3. 次のエンドポイントに対して、認証ヘッダーを使用して GET 要求を行います。

```
Method: GET
Endpoint: https:// <servicenow_base_url> /api/now/incident
Authorization: Bearer YOUR_ACCESS_TOKEN
```

サードパーティ ID トークンの構成

サードパーティ ID トークンを構成し、外部 IdP を介してユーザー ID を検証することで安全な認証を有効にします。サードパーティ ID トークンは、保存される認証情報を減らすことでセキュリティを向上させ、シームレスな認証を確認し、OpenID Connect (OIDC) などの業界標準との相互運用性をサポートします。

始める前に
必要なロール: oauth_admin、mi_admin、admin

手順

1. 移動先 マシン **ID** コンソール > > インバウンド統合 > 新しい統合 > サードパーティ **ID** トークン.
2. [詳細] フォームのテキストフィールドを適切な情報で更新します。

[詳細] フォーム

フィールド	説明
Name (名前)	認証時にリソース所有者 (ユーザー) によって指定された名前。
クライアント **ID**	アプリケーションを識別するためにアサインされた一意の ID。
クライアントシークレット	アプリケーションと認証サーバーのみが識別できる秘密キー。アプリケーションはこのキーを使用して、アクセストークンを認証して取得します。

トークン制限を適用すると、OAuth アクセストークンの使用方法に制限が適用され、トークンが特定の条件下でのみ有効であることを検証することでセキュリティが強化されます。[トークン制限の適用] チェックボックスをオンにして、OAuth アクセストークンを API アクセスポリシーで定義された特定の API に制限します。[トークン制限の適用] がオフになっている場合、トークンは他の REST API で使用できます。

3. [詳細オプション (オプション)] フォームのテキストフィールドを適切な情報で更新します。

詳細オプションフォーム

フィールド	説明
アクセストークンの有効期間	OAuth アクセストークンが期限切れになるまで有効な期間 (秒)。 注: デフォルト値は 1800 秒です。
リフレッシュトークンの有効期間	OAuth リフレッシュトークンが期限切れになるまで有効な期間 (秒)。 注: デフォルト値は 8,640,000 秒です。

4. [認証スコープ (オプション)] フォームのテキストフィールドを適切な情報で更新します。認証スコープは、アプリケーションがリソースに対して持つアクセスのレベルを定義します。アクセスする特定の REST API の認証スコープを選択します。

認証スコープフォーム

フィールド	説明
認証スコープ	アプリケーションがリソースに対して持つアクセスのレベル。認証スコープは、アクセストー

自動翻訳

フィールド	説明
	クンが API またはデータに対して実行できるアクションを制限します。
制限認証	認証を制限する API の名前。

a. [新しい認証スコープの作成] を選択して、新しい認証スコープを追加します。

5. **[Save** (保存)] を選択します。
   新しいサードパーティ ID トークンが作成されます。
6. 検索項目 すべて > インバウンド統合 > アプリケーションレジストリ をクリックして、新しく作成されたサードパーティ ID トークンを表示します。

JSON Web トークンベアラー権限許可

このフローは、クライアントアプリケーションが、それ自体として、またはユーザーの代わりに、ServiceNow リソースへの安全な無人アクセスを必要とする場合に使用します。

クライアント アプリケーションは、それが表すユーザーやシステムなどの ID 関連の要求を含む署名済み JWT を生成します。ServiceNowインスタンスに送信して、アクセストークンを要求します。

JWT 構造

JWT は、クライアントの秘密鍵を使用して署名する必要があります。次の標準要求を含める必要があります。

- iss – 発行者 (クライアント ID)
- sub:件名 (ユーザーまたはシステム ID)
- aud – 対象者 (ServiceNow トークンエンドポイント)
- exp – 有効期限
- iat – 発行日

注: ServiceNow は公開鍵 (OAuth JWT プロファイルにアップロード) を使用して署名を検証し、サブ要求をユーザーレコードにマッピングします。

関連トピック

JSON Web トークンの付与ワークフロー

OAuth JSON Web トークンベアラー権限許可の構成

JSON Web トークンの付与ワークフロー

このフローは、クライアントアプリケーションが、それ自体として、またはユーザーの代わりに、ServiceNow リソースへの安全な無人アクセスを必要とする場合に使用します。

始める前に

必要なロール: oauth_admin、mi_admin、admin

このタスクについて

クライアント アプリケーションは、それが表すユーザーやシステムなどの ID 関連の要求を持つ署名済み JWT を生成します。クライアントアプリケーションは、JWT を ServiceNow インスタンスに送信してアクセストークンを要求します。

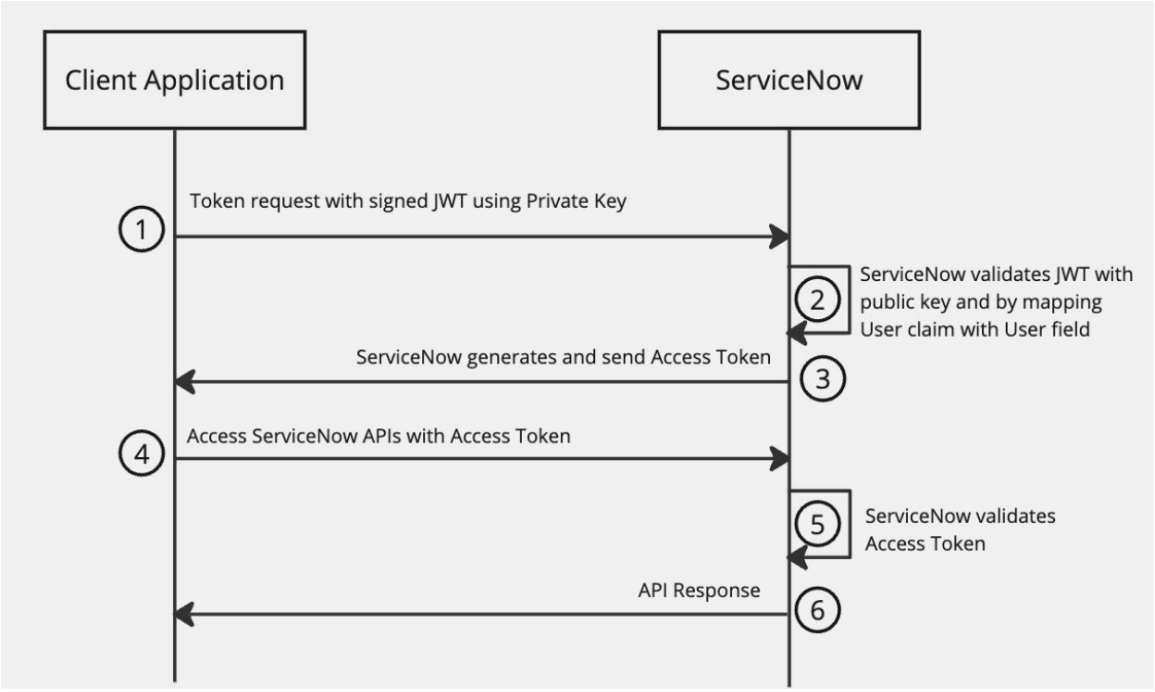
- ユーザーの代理として行動する場合:

  トークンは、以前に認証されたユーザーを表します。これにより、ユーザーに認証情報や同意を求めることなく、安全でシームレスなアクセスが可能になります。ServiceNow は、署名されたトークンからユーザーの ID を検証することで要求を信頼し、リアルタイムのユーザー操作の必要性を排除します。

- それ自体として行動する場合:

  トークンは、クライアントアプリケーションを識別して認証します。共有シークレットを使用する代わりに、アプリケーションは秘密鍵を使用してトークンに署名します。これにより、クライアント認証情報の付与に代わる、より安全な代替手段が提供されます。

JWT 権限許可ワークフロー



手順

1. クライアントアプリケーションは、秘密鍵で署名された JWT を使用して、トークン要求を ServiceNow に送信します。
2. ServiceNow 対応する公開鍵を使用して JWT を検証します。
   トークン内の `サブ` (サブジェクト) クレームを `sys_user` レコードにマッピングします。
3. ServiceNow JWT を検証し、アクセストークンを発行します。
4. クライアントは、 ServiceNowへの API 要求にアクセストークンを含めます。
5. ServiceNow はアクセストークンを検証し、適切な API 応答を返します。

OAuth JSON Web トークンベアラー権限許可の構成

OAuth JSON Web トークン (JWT) ベアラー権限許可を構成すると、ユーザーの操作なしでトークンベースの認証が保護されます。署名付き JWT でセキュリティを強化し、繰り返しログイン試行をなくすことで認証のオーバーヘッドを削減します。

始める前に
必要なロール: `oauth_admin`、`mi_admin`、`admin`

JSON Web トークン (JWT) でサポートされているアルゴリズム:
RS256、RS384、RS512、ES256、ES384、ES512。

手順

1. 移動先 マシン **ID** コンソール > > インバウンド統合 > > 新しい統合 > **JWT** ベアラー権限許可.
2. [詳細] フォームのテキストフィールドを適切な情報で更新します。

[詳細] フォーム

フィールド	説明
Name (名前)	JWT OAuth アクセスが必要なアプリケーションを識別する一意の名前
クライアント **ID**	自動生成された一意のアプリケーション ID。システムは、このフィールドの値を使用して公開鍵または共有鍵を取得し、JWT を検証します。このフィールドの値は、JWT の発行者要求および対象者要求の値と一致する必要があります。
クライアントシークレット	インスタンスとクライアントアプリケーションまたは Web サイトの両方が、相互通信を許可するために使用する共有シークレット文字列。インスタンスでクライアントシークレットを自動生成するには、このフィールドを空白のままにします。既存のクライアントシークレットを表示するには、ロックアイコンを選択します。
ユーザーフィールド	JWT の件名要求の値を照合するためにシステムが使用するユーザー (sys_user) テーブル内のフィールド。 例： サブジェクト要求の値が user.name@example.com のトークンを追加する場合は、[ユーザーフィールド] を [メール] に設定します。このフィールドは、メールフィールドで user.name@example.com の値を検索し、一致するユーザーレコードを受信要求で使用するようにシステムに指示します。
JTI 検証を有効化	トークン交換のたびに新しいトークンを要求する場合に選択します。 デフォルト：オン
JWKS URL	JSON Web キー設定 URL。これは、JSON 形式の公開鍵のコレクションです。ID プロバイダーは、クライアントアプリケーションとサービスがキーを取得して JSON Web トークン (JWT) の署名を検証できるように、既知の URL で JWKS を公開します。
JTI 請求	各トークンの一意の識別子。ServiceNow はこの要求を使用して、トークンが再利用されていないことを確認することで、トークンの再生を検出して防止します。

フィールド	説明
コメント	関連するコメントを追加します。
JWT 検証者マップ	ID プロバイダー、検証方法 (JWKS URL や証明書など)、および ServiceNow ユーザー フィールドへの JWT 要求のマッピングを指定します。 プラスアイコンをクリックして、マップを追加または編集します。**[JWT** 検証者マップ] ページで次の詳細を入力します。 ○ 名前:JWT 検証者マップ構成の一意の名前。 ○ アプリケーション:検証者マップが使用されるアプリケーション。 ○ kid (キー ID):JWT 署名の検証に使用されるキーの識別子。 ○ システム証明書:トークンの検証に使用される ServiceNow 証明書レコード ○ 共有キー:トークンの検証に使用される対称キー。
Active (アクティブ)	チェックボックスをオンにすると、OAuth アプリケーションがアクティブになります。

3. [詳細オプション (オプション)] フォームのテキストフィールドを適切な情報で更新します。

詳細オプション (オプション) フォーム

フィールド	説明
トークン制限の適用	認証プロファイルを有効にするように設定された API でのみトークンの使用を有効にする場合に選択します。API アクセスポリシーを使用してアクセス許可を設定できます。 デフォルト：未選択。
JWKS キャッシュ有効期間	ServiceNow が ID プロバイダーからの JSON Web キーセット (JWKS) をキャッシュする期間 (分)。
アクセストークンの有効期間	アクセストークンが期限切れになるまで有効な期間 (秒)。
クロックスキュー	トークンの生成と検証に関連するサーバーまたはデバイスのシステム クロックにわずかな違いがあると、時間的制約のある要求を検証するときに問題が発生する可能性があります。上記の時間を調整します。デフォルト値:0 秒。

4. [認証スコープ (オプション)] フォームのテキストフィールドを適切な情報で更新します。

注: 認証スコープを選択すると、関連するすべての API が [認証の制限] テキスト ボックスに自動的に入力されます。

認証スコープフォーム

フィールド	説明
認証スコープ	アプリケーションのアクセスレベル。認証スコープは、アクセストークンが API またはデータに対して実行できるアクションを制限します。
制限認証	認証を制限する API の名前。

a. [別の行を追加] を選択して、認証スコープを追加します。

b. [新しい認証スコープの作成] を選択して、新しい認証スコープを追加します。
[スコープ] フィールド のテキストボックスに認証スコープの名前を入力して、新しく作成した認証スコープを選択します。新しい認証スコープに関連付ける必要がある API を手動で追加および編集できます。

> **注:** [認証スコープ] メニューから API を追加または編集すると、同じ認証スコープに関連付けられているすべての OAuth エンティティに影響します。

5. **[Save** (保存)] を選択します。
新しい OAuth JSON Web トークンベアラー権限許可が作成されます。
6. 検索項目 すべて > インバウンド統合 > アプリケーションレジストリ をクリックして、新しく作成された JWT ベアラー権限許可を表示します。

リソース所有者のパスワード認証情報の付与

OAuth リソース所有者パスワード認証情報 (ROPC) 権限許可を設定すると、アプリケーションは認証情報を直接使用してアクセストークンを取得してユーザーを認証できます。

セキュリティに関する考慮事項

ROPC フローは、ユーザー認証情報をクライアントアプリケーションに直接公開するため、最新の代替手段よりも本質的に安全性が低くなります。クライアントが完全に信頼され、厳密に制御され、安全に管理されているシナリオでのみ使用する必要があります。

絶対に必要な場合を除き、最新のアプリケーションでこの権限許可を使用しないでください。安全なユーザーベースのアクセスのために、PKCE を使用した認証コードフローを使用することを強くお勧めします。これにより、認証情報がクライアントから保護され、安全なリダイレクトとトークン処理のプラクティスが活用されます。

関連トピック

- リソース所有者のパスワード認証情報権限許可ワークフロー
- OAuth リソース所有者のパスワード認証情報権限許可の構成

リソース所有者のパスワード認証情報権限許可ワークフロー

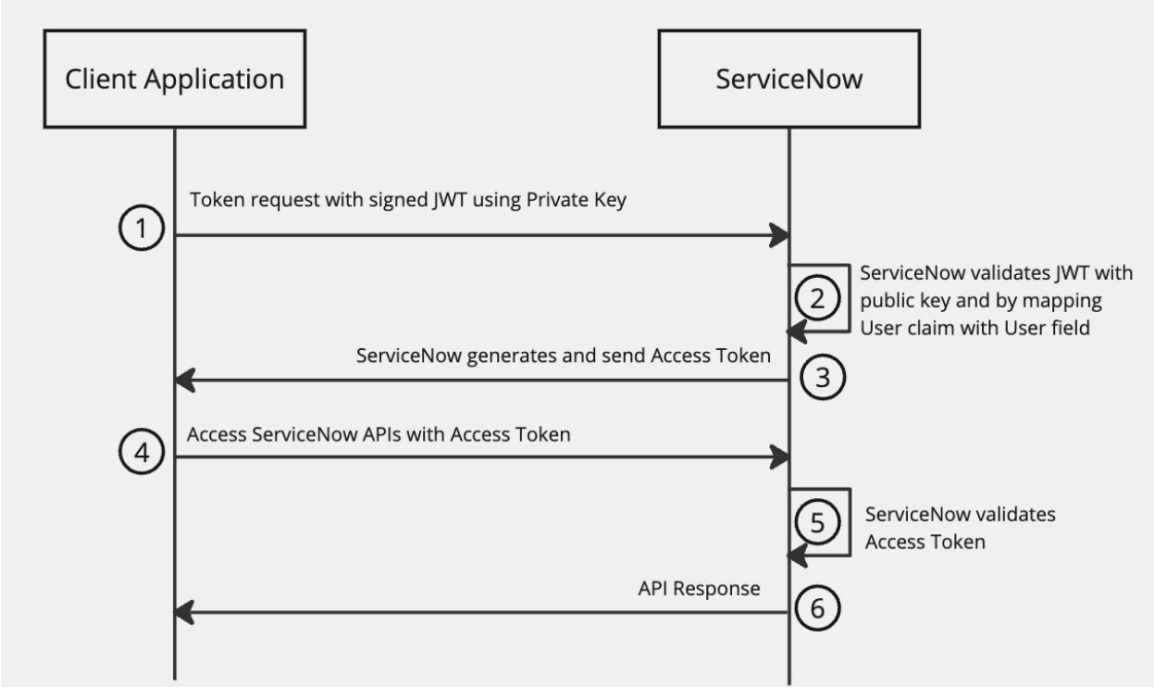
このフローは、安全な代替手段が実現不可能な従来の環境または高度に制御された環境で使用されます。クライアント アプリは、ユーザーの資格情報を直接収集して ServiceNow に送信し、アクセストークンを取得するため、信頼できる内部使用にのみ適しています。

始める前に
必要なロール: oauth_admin、mi_admin、admin

このタスクについて
この手順では、信頼できるクライアントアプリケーションがユーザー認証情報を直接処理してアクセストークンを取得し、それを使用して ServiceNow API にアクセスする方法について概説します。

ユーザーはアプリを通じてログインし、自分の認証情報とユーザーの認証情報の両方をServiceNowに送信します。 ServiceNow 認証情報を検証し、アプリが API を呼び出すために使用するアクセストークンを発行します。

リソース所有者のパスワード認証情報権限許可ワークフロー



手順

1. ユーザーがクライアントアプリケーションにログインします。
2. クライアントアプリケーションは、次のパラメーターを使用してトークン要求を送信します。
   - クライアント ID とクライアントシークレット。
   - ユーザーのユーザー名とパスワード。

例

```
Method: POST
Endpoint: https://<servicenow_base_url>/oauth_token.do
Headers: Content-Type: application/x-www-form-urlencoded
```

トークン要求パラメーター

パラメーター	必須	説明
grant_type	はい	OAuth 権限許可タイプを指定します。
client_id	はい	クライアントアプリケーションの一意の識別子。 形式:YOUR_CLIENT_ID

パラメーター	必須	説明
client_secret	はい	クライアントアプリケーションの秘密キー。 形式:YOUR_CLIENT_SECRET
ユーザー名	はい	ユーザーの ServiceNow ユーザー名。
パスワード	はい	ユーザーの ServiceNow パスワード。
スコープ	オプション	要求されるアクセスのレベルを定義します。 例： ○ incident_read ○ incident_write

3. ServiceNow はクライアントとユーザーの両方の認証情報を検証し、アクセストークンを返します。
4. クライアントはアクセストークンを使用して ServiceNow API を呼び出します。

   例

```
Method: GET
Endpoint: https://<servicenow_base_url/api/now/incident
Authorization: Bearer YOUR_ACCESS_TOKEN
```

5. ServiceNow はアクセストークンを検証し、API 応答を返します。

OAuth リソース所有者のパスワード認証情報権限許可の構成

OAuth リソース所有者パスワード認証情報 (ROPC) 権限許可を設定すると、アプリケーションは認証情報を直接使用してアクセストークンを取得してユーザーを認証できます。この方法は、ブラウザベースのフローを使用せずに認証を必要とする信頼できるアプリケーションやレガシーシステムに最適であり、安全なトークン検証と制御された API アクセスを可能にします。

始める前に
必要なロール: oauth_admin、mi_admin、admin

手順

1. 移動先 マシン **ID** コンソール > > インバウンド統合 > > 新しい統合 > **OAuth** リソース所有者のパスワード認証情報の付与.
2. [詳細] フォームのテキストフィールドを適切な情報で更新します。

   [詳細] フォーム

フィールド	説明
Name (名前)	認証時にリソース所有者 (ユーザー) によって指定された名前。

フィールド	説明
クライアント **ID**	アプリケーションを識別するためにアサインされた一意の ID。
クライアントシークレット	アプリケーションと認証サーバーのみが識別できる秘密キー。アプリケーションはこのキーを使用して、アクセストークンを認証して取得します。
Active (アクティブ)	チェックボックスをオンにすると、OAuth アプリケーションがアクティブになります。

3. [詳細オプション (オプション)] フォームのテキストフィールドを適切な情報で更新します。

   トークン制限を適用すると、OAuth アクセストークンの使用方法に制限が適用され、トークンが特定の条件下でのみ有効であることを検証することでセキュリティが強化されます。[トークン制限の適用] チェックボックスをオンにして、OAuth アクセストークンを API アクセスポリシーで定義された特定の API に制限します。[トークン制限の適用] がオフになっている場合、トークンは他の REST API で使用できます。

   [詳細] フォーム

フィールド	説明
アクセストークンの有効期間	アクセストークンが期限切れになるまで有効な期間 (秒)。
リフレッシュトークンの有効期間	リフレッシュトークンが発行された後、リフレッシュトークンが有効なままになる期間 (秒単位) は、[有効期間] フィールドで指定されます。

4. [認証スコープ (オプション)] フォームのテキストフィールドを適切な情報で更新します。認証スコープは、アプリケーションがリソースに対して持つアクセスのレベルを定義します。アクセスする特定の REST API の認証スコープを選択します。

   注: 認証スコープを選択すると、関連するすべての API が [認証の制限] テキスト ボックスに自動的に入力されます。

   認証スコープフォーム

フィールド	説明
認証スコープ	アプリケーションのアクセスレベル。認証スコープは、アクセストークンが API またはデータに対して実行できるアクションを制限します。
制限認証	認証を制限する API の名前。

   注: [認証スコープ] メニューから API を追加または編集すると、同じ認証スコープに関連付けられているすべての OAuth エンティティに影響します。

   a. [新しい認証スコープの作成] を選択して、新しい認証スコープを追加します。

5. [別の行を追加] を選択して、関連する API を使用して別の認証スコープを作成します。
6. **[Save** (保存)] を選択します。

新しい OAuth リソース所有者のパスワード認証情報権限許可が作成されます。

7. 検索項目 すべて > インバウンド統合 > アプリケーションレジストリ して、新しく作成された OAuth リソース所有者のパスワード認証情報権限許可を表示します。

古いインバウンド統合エクスペリエンス

古い経験 - インバウンド統合。

注:

次の権限許可タイプに応じて、OAuth 受信を構成できます。

- OAuth 認証コード権限許可フロー

  注: 認証コードフローの場合、ユーザーはローカルログイン、SSO、または MFA による認証を完了し、同意する必要があります。

- パスワード権限許可
- JWT ベアラー権限許可フロー
- ID トークンフロー
- OAuth の暗黙的な権限許可
- クライアント認証情報

Zurich リリースからの次の拡張機能を含む OAuth 統合を構成します。

- サードパーティシステムのセキュリティ要件を満たすために、クライアントシークレットの長さを最大 4096 文字に増やします。
- JSON Web トークン (JWT) 署名検証の公開鍵を自動的に管理および更新するための JSON Web キーセット (JWKS) URL を指定します。
- 受信 JSON Web トークン (JWT) に対して、楕円曲線デジタル署名アルゴリズム (ES) 署名アルゴリズム (ES256、ES384、ES512 など) で署名された JWT 権限許可タイプを使用して OAuth トークンを要求します。
- 受信 OpenID Connect (OIDC) フローと JWT ベアラーフローの両方で、JWT ID (JTI) 要求名をカスタマイズします。

OAuth 認証コード権限許可フロー

認証コード権限許可フローでは、リソースを信頼する OAuth サーバーで直接認証することで、ユーザーはリソースへのアクセスを許可されます。これは、ユーザー名/パスワード認証情報を使用した認証とは対照的です。

この OAuth 認証コードフローの実装により、REST を介してリソースにアクセスできるようになります。認証コードフレームワークは、ユーザーにユーザー名/パスワードの入力を要求するのではなく、ユーザーが設定した承認済み URL を介してアクセストークンを取得します。ユーザー名/パスワードは、リソースへのアクセスを要求しているクライアントに公開されることはありません。

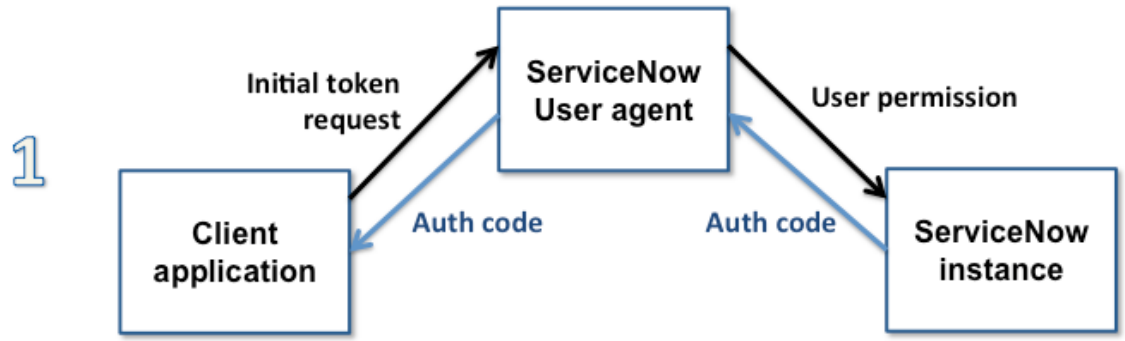
認証サーバーとしての **ServiceNow** インスタンス

OAuth サーバーは通常、サードパーティの認証サーバーです。認証コードフローのトークンを発行する認証サーバーとして ServiceNow インスタンスを指定することもできます。

制限付きリソースを所有するユーザーは、アクセスを許可する必要があります。またユーザーは、発行されたアクセストークンをいつでも取り消してアクセスを終了できます。

認証コード権限許可フロープロセス

認証コード権限許可フロープロセスは、次の 3 段階のステップで構成されています。



ステップ 1 では、クライアントアプリケーションまたは Web サイトが、ユーザーエージェントを介してインスタンスへの GET 要求の形式で REST API 呼び出しを開始します。通常、REST 呼び出しは、エンドユーザーがクライアントアプリケーションまたは Web サイトのボタンまたはリンクをクリックしてアクセストークンを要求すると開始されます。クライアントアプリケーションでは、エンドユーザーは認証 URL、トークン URL、クライアント ID、およびクライアントシークレットも指定する必要があります。これらのアイテムの説明については、「サードパーティ OAuth プロバイダーの使用」に記載されているフィールドの説明を参照してください。クライアントが権限許可タイプを要求した場合、エンドユーザーは認証コードを選択する必要があります。

クライアントアプリケーションからインスタンスへの GET 要求の例：

```
https://myinstance.service-now.com/oauth_auth.do?response_type=code&redirect_uri={the_redirect_url}&client_id={the_client_identifier}
```

注: **response_type** は、標準の OAuth コード権限許可フローを使用するための **code** である必要があります。

エンドユーザーは、インスタンスの制限付きリソースへのアクセスを手動で許可する必要があります。ServiceNow 実装では、エンドユーザーはインスタンスにログインする必要があります。インスタンスは、[許可] および [拒否] ボタンがある UI ページでエンドユーザーにプロンプトを表示します。

クライアントアプリケーションが実際にトークンフォームを要求しているアイテムは、作成した OAuth プロバイダーアプリケーションレジストリレコードで、認証エンドポイントとも呼ばれます (「サードパーティ OAuth プロバイダーの使用」を参照してください)。認証コードが認証エンドポイントからクライアントに送信されます。この場合、クライアントには直接移動せず、認証エンドポイントフォームで指定するリダイレクト **URL** に移動します。この URL はコールバック URL とも呼ばれます。この URL は、クライアントアプリケーションまたは Web サイトから取得できます。

認証コードを提供する、インスタンスからクライアントアプリケーションへの応答の例：

```
https/http://{callbackURL}?code={the actual auth code}
```

自動翻訳

クライアントアプリケーションは認証コードを取得したので、クライアントはそのコードを使用してアクセストークンを要求します。認証コードは、ユーザーがステップ 1 で同意したことを証明します。

クライアントアプリケーションから ServiceNow インスタンスへの POST 要求の例 (認証コードを提供し、アクセストークンを要求) :

```
https://myinstance.service-now.com/oauth_token.do?grant_type=authorization_code&code={the auth
 code}&redirect_uri={the_same_redirect_url}&client_id={the_same_client_identifier}&client_secret={client_secret_value}
```

インスタンスのエンドポイントは、アクセストークンとリフレッシュトークンを返します。リフレッシュトークンを使用して、追加のアクセストークンを要求できます。

インスタンスでは、トークンの取り消しなどのトークンの管理を実行できます。「OAuth トークンの管理」を参照してください。

クライアントアプリケーションは、アクセストークンを使用して REST API を認証します。クライアントアプリケーションを認証した後、REST API は要求されたデータを JSON ペイロードで返します。

インシデント [incident] テーブルのデータの JSON ペイロードの GET 要求の例 :

```
https://myinstance.service-now.com/api/now/table/incident?access_token={the_token}
```

注: システムは、暗黙的な権限許可コードフローとも呼ばれる OAuth 暗黙的な権限許可もサポートしています。

統合サポート

認証コードフローは、インスタンスについて次の統合をサポートしています。

- マルチ SSO
- SAML 2.0 Update 1
- マルチファクター認証

モバイルインターフェイスもサポートされています。

認証コードフローを使用した **OAuth** エンドポイントへのアクセス許可

ServiceNow インスタンスの保護されたリソースを所有するエンドユーザーは、インスタンスがアクセストークンを提供できるようにする前に、リソースへのアクセスを許可する必要があります。

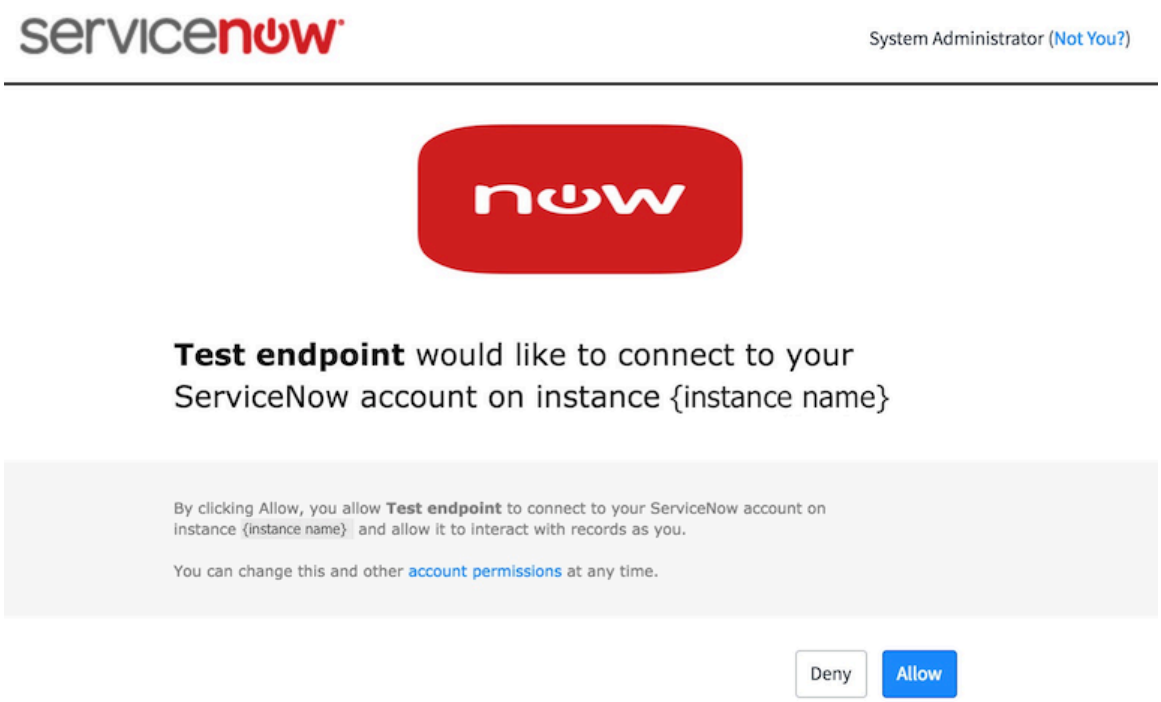
始める前に

必要なロール : なし。保護されたリソースを保持するインスタンスにすでにログインしている必要があります。あるいは、ServiceNowアドミニストレーターが設定済みの認証方法 (マルチファクター認証や SAML など) を使用してログインすることもできます。

手順

1. インスタンスの保護されたリソースへのアクセスを要求しているクライアントアプリケーションのリンクまたはボタンをクリックします。
   これにより、トークン要求が開始されます。あるインスタンスから別のインスタンスへの REST 呼び出しを行う場合、このリンクは REST メッセージフォームの **[OAuth** トークンの取得] になります。
2. ログインしていない場合は、今すぐログインしてください。
   右上隅で指定されているユーザーと同じユーザーでない場合は、[別のユーザーである場合] をクリックしてログインします。
3. アカウントの権限をクリックして、発行済みのアクセストークンのリストを開きます。
   これは、 セルフサービス > 自分の接続済み アプリトークンリスト。
4. [許可] をクリックしてアクセスを許可し、インスタンスに認証コード (認証コードフローを使用する場合) またはアクセストークン (暗黙的な権限許可タイプを使用する場合) を発行させます。

   [拒否] をクリックすると、認証は許可されませんが、インスタンスからログアウトされることはありません。



   アクセスを確認するメッセージが表示されます。インスタンスの REST メッセージフォームからアクセスを要求すると、フォームの上部に [OAuth リフレッシュトークンが利用可能になり、{date} に期限切れになります (OAuth Refresh token is available and will expire at {date})] というメッセージが表示されます。

認証コードフローステータスパラメーターの要件

glide.oauth.state.parameter.required システムプロパティを使用すると、認証コードフローの OAuth 要求でステータスパラメーターを要求できます。

ステータスパラメーター

必要なロール：なし。

Madrid リリース以降、システムプロパティ *`glide.oauth.state.parameter.required`* によって OAuth 要求のステータスパラメーターが追加されます。zboot インスタンスの場合、このプロパ

ティは true です。アップグレードされたインスタンスの場合、このプロパティは存在しないため、ステータスパラメーターは有効になりません。ステータスパラメーターは文字列値であり、特殊文字を含めることはできません。ステータスパラメーターを空または空白文字にすることはできません。

ステータスパラメーターの検証

インスタンスにアクセスするクライアント用のエンドポイントを作成し ます。oauth_auth.do の認証コードフローを開始します。例：

```
http://myinstance.service-now.com/oauth_auth.do?grant_type=authorization_code&client_id=e9dba45b380d1300e676ccc91cef468f&response_type=code
```

要求でステータスパラメーターを指定しないと、エラーが発生し、認証コードは返されません。(要求にステータスパラメーターがありません。(Missing State parameter in request.))

ステータスパラメーターを要求に追加します。

```
http://myinstance.service-now.com/oauth_auth.do?grant_type=authorization_code&client_id=e9dba45b380d1300e676ccc91cef468f&response_type=code&state=123
```

ステータスパラメーターを追加すると、ログイン画面にリダイレクトされ、通常の認証コードフローが認証コードを返します。

> **注:** 応答 URL には、要求で渡されたステータスパラメーターが含まれています。この例では、追加されたパラメーターは `state=123` です。

認証コードフローが oauth_initiator.do から開始される場合：

```
http://myinstance.service-now.com/oauth_initiator.do?oauth_requestor_context=sys_rest_message&oauth_requestor=eab8341fec0d1300964f214a2c2fcf67&oauth_provider_profile=dfa8f01fec0d1300964f214a2c2fcf51&response_type=code
```

ステータスパラメーターは、oauth_auth.do によってリダイレクトされると自動的に追加されます。

```
http://myinstance.service-now.com/oauth_auth.do?response_type=code&state=-790938844&redirect_uri=http://10.11.95.5:16001/oauth_redirect.do&client_id=e9dba45b380d1300e676ccc91cef468f
```

認証コードフローの例：認証サーバーとしての **ServiceNow** インスタンス

インスタンスを認証サーバーとして使用し、認証コードフローを使用してクライアントにトークンを発行するすることもできます。

始める前に

必要なロール：なし。

この例では、2 つのインスタンスを使用します。1 つは認証サーバーとして、もう 1 つはクライアントとして使用します。1 つのインスタンスが REST 呼び出しを使用して、別のインスタンスからトークンを要求します。

両方のインスタンスで OAuth のアクティブ化 を実行する必要があります。

手順

1. 認証サーバーインスタンス (Istanbul 以降のリリースを実行中) で、 システム **OAuth** > アプリケーションレジストリー をクリックし、[新規] をクリックします。
2. [外部クライアント用の **OAuth API** エンドポイントを作成 **(Create an OAuth API endpoint for external clients)]** をクリックします。

3. 「インスタンスにアクセスするクライアント用のエンドポイントを作成し ます。」の説明に従って、OAuth アプリケーションレコードのフォームフィールドに入力します。
これらの手順を完了すると、認証サーバーが設定されます。次の手順に従って、クライアントサーバーを設定します。
4. クライアントインスタンスで、 システム **OAuth** > アプリケーションレジストリー をクリックし、[新規] をクリックします。
5. [サードパーティ **OAuth** プロバイダーに接続 **(Connect to a third party OAuth Provider)]** をクリックします。
6. 「」の説明に従って、OAuth アプリケーションレコードのフォームフィールドに入力します。
次のフィールド値に注意：
   - 名前：OAuth アクセスが必要なアプリケーションを識別する一意の名前。
   - クライアント **ID**：認証サーバー用に作成したアプリケーションレジストリレコードのクライアント ID。
   - クライアントシークレット：[読み取り専用] 自動生成された一意のアプリケーション ID。インスタンスは、アクセストークンを要求するときにクライアント ID を使用します。
   - デフォルトの権限許可タイプ：[認証コード] を選択します。
   - 認証 **URL**：認証サーバーであるインスタンスの URL。URL の末尾に `oauth_auth.do` を追加することを忘れないでください。
   - ロゴ **URL**：アプリケーションロゴとして使用するイメージを含む URL。ユーザーがインスタンス上で制限されているリソースへのアクセスをクライアントアプリケーションに許可する要求を受け取ると、承認ページにロゴが表示されます。
   - トークン **URL**：認証サーバーであるインスタンスの URL。URL の末尾に `oauth_token.do` を追加することを忘れないでください。
   - リダイレクト **URL**：このインスタンスの URL：クライアントサーバーインスタンス。URL の末尾に `oauth_redirect.do` を追加することを忘れないでください。
7. 認証コード権限許可タイプでレコードのプロファイルを作成します。
クライアントサーバーがセットアップされます。これで、送信 REST メッセージを作成し、OAuth トークンを取得できるようになりました。

インスタンスにアクセスするクライアント用のエンドポイントを作成し ます。

外部クライアントアプリケーションが ServiceNow インスタンスにアクセスするための OAuth アプリケーションエンドポイントを作成します。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > システム **OAuth** > アプリケーションレジストリー をクリックし、[新規] をクリックします。
2. インターセプターページで、[外部クライアント用の **OAuth API** エンドポイントを作成] をクリックし、フォームに入力します。

フィールド	説明
名前	OAuth アクセスが必要なアプリケーションを識別する一意の名前。

フィールド	説明
クライアント ID	[読み取り専用] 自動生成された一意のアプリケーション ID。インスタンスは、アクセストークンを要求するときにクライアント ID を使用します。
クライアントシークレット	[必須] インスタンスとクライアントアプリケーションまたは Web サイトの両方が、相互通信を許可するために使用する共有シークレット文字列。インスタンスは、アクセストークンを要求するときにクライアントシークレットを使用します。インスタンスでクライアントシークレットを自動生成するには、このフィールドを空白のままにします。既存のクライアントシークレットを表示するには、ロックアイコンをクリックします。
リダイレクト URL	認証サーバーがリダイレクトするコールバック URL。リソースへのアクセスを要求しているクライアントの完全な URL を、末尾に /oauth_redirect.doを付けて入力します。たとえば、http://token_consumer:port/oauth_redirect.do のようになります。考えられるすべてのトークンコンシューマーに対して必要な数の URL を入力します。インスタンスは、受信要求の URL をいずれかのリダイレクト URL と照合します。一致するものがない場合、インスタンスは最初のリダイレクト URL を使用します。
ロゴ URL	アプリケーションロゴとして使用するイメージを含む URL。ユーザーがインスタンス上で制限されているリソースへのアクセスをクライアントアプリケーションに許可する要求を受け取ると、承認ページにロゴが表示されます。
アクティブ	チェックボックスをオンにすると、アプリケーションレジストリがアクティブになります。
リフレッシュトークンの有効期間	リフレッシュトークンが有効である秒数。インスタンスは、リフレッシュトークンを要求するときに有効期間の値を使用します。デフォルトでは、リフレッシュトークンは 100 日 (8640000 秒) で期限切れになります。
トークン制限の適用	認証プロファイルを許可するように設定された API でのみトークンの使用を許可する場合に選択します。API アクセスポリシーを使用してアクセス許可を設定できます。詳細については、「REST API アクセスポリシーの作成」を参照してください。 デフォルト：未選択。
モバイルクライアント	モバイルアプリまたは Web のエンティティを表します。この情報は、モバイルまたは Web でログイン情報を分析するために使用されます。
アクセストークン有効期間	アクセストークンが有効である秒数。インスタンスは、アクセストークンを要求するときに有効期間の値を使用します。デフォルトでは、アクセストークンは 30 分 (1800 秒) で期限切れになります。

フィールド	説明
コメント	アプリケーションに関連付ける追加情報。
クライアントタイプ	クライアントのタイプに基づいて、クライアントタイプを選択します。オプション： ○ **iFrame** 埋め込み ○ ユーザーとして連携 ○ サービスとして連携 詳細については、「OAuth および SSO レコードのクライアントタイプの構成」を参照してください。

3. [送信] をクリックします。

結果
システムにより、OAuth クライアントタイプのレコードがアプリケーションレジストリ [oauth_entity] テーブルに作成されます。インスタンスが実際にトークンと認証コードを発行すると、それらはテーブルに格納されます。詳細については、「OAuth トークンの管理」を参照してください。

OAuth API 応答パラメーター

OAuth 2.0 API は、次のパラメーターを name:value ペアとして含む JSON 応答を生成します。

アクセストークン応答パラメーター

応答パラメーター	説明
scope	アクセストークンによって付与されるアクセスの範囲。スコープは常に **useraccount** です。これは、アクセストークンがトークンを承認したユーザーアカウントと同じ権限を持つことを意味します。たとえば、Abel Tuter がログイン認証情報を提供してアプリケーションを承認した場合、その結果生じるアクセストークンは Abel Tuter と同じアクセス権限をトークンベアラーに付与します。
token_type	OAuth RFC で定義されている要求によって発行されたトークンのタイプ。トークンタイプは常にベアラー です。これは、アクセストークンを持っているすべてのユーザーが、暗号化キーを提供しなくても保護されたリソースにアクセスできることを意味します。OAuth 2.0 でのベアラートークンの使用方法の詳細については、RFC6750 を参照してください。
expires_in	アクセストークンの有効期間 (秒単位)
refresh_token	リフレッシュトークンの文字列値
access_token	アクセストークンの文字列値アクセストークンの有効期限内に行われたアクセス要求は、常に現在のアクセストークンを返します。
format	[オプション] 応答の出力形式。この値は常に JSON です。

注: OAuth プロバイダーが応答の本文を「Content-Type」でなく「content-type」として送信すると、OAuth HTTP クライアントで応答が正しく解析されない可能性があります。この問題を修正するには、次のパラメーターを使用してシステムプロパティを作成します。

フィールド	値
名前	glide.oauth.inhouse.httpclient.enabled
タイプ	true | false
値	false

システムプロパティの作成の詳細については、「システムプロパティを追加する」を参照してください。

次の例は、アクセストークン要求によって返される JSON 文字列を示しています。(読みやすくするためにスペースを追加しました)。

```
{"scope":"useraccount","token_type":"Bearer","expires_in":1800,
"refresh_token":"w599voG89897rGVDmdp12WA681r9E5948c1CJTPi8g4HGc4NWaz62k6k1K0FMxHW40H8yOO3Hoe",
"access_token":"F0jh9korTyzd9kaZqZ0SzjKZuS3ut0i4P46Lc52m2JYHiLIcqzFAumpyxshU9mMQ13gJHtxD2fy"}
```

OAuth API 要求パラメーター

アクセストークン要求が使用する OAuth API 要求パラメーターについて説明します。

注: OAuth API のコンテンツタイプは *application/x-www-form-urlencoded* である必要があります。コンテンツタイプが *application/json* の場合、詳細不明のエラーが発生します。

アクセストークン要求パラメーター

要求パラメーター	説明
grant_type	[必須] アクセストークンの要求を許可する認証情報のタイプ。このパラメーターは、次のいずれかの値でなければなりません。 • **password**：アクセストークン要求を許可するためのユーザー認証情報のセット。ユーザー名とパスワードのパラメーターでユーザー認証情報を指定します。 • **refresh_token**：既存のリフレッシュトークンによってアクセストークン要求が許可されます。refresh_token パラメーターでリフレッシュトークンを指定します。
client_id	[必須] アクセストークンを要求するクライアントアプリケーションに対して自動的に生成される一意の ID
client_secret	[必須] インスタンスと OAuth アプリケーションが相互の通信を許可するために使用する共有シークレット文字列
ユーザー名	アクセストークン要求を許可するユーザーアカウント名。このパラメーターは、**password** の **grant_type** を使用したアクセストークン要求に必要です。

自動翻訳

アクセストークン要求パラメーター (続く)

要求パラメーター	説明
パスワード	アクセストークン要求を許可するユーザーアカウントのパスワード。このパラメーターは、**password** の **grant_type** を使用したアクセストークン要求に必要です。
refresh_token	アクセストークン要求を許可する既存のリフレッシュトークン。このパラメーターは、**grant_type** の **refresh_token** を使用したアクセストークン要求に必要です。

ユーザー認証情報を使用した要求

インスタンスでは、最初にクライアントを許可するとき、または新しいリフレッシュトークンの作成を許可するときに、ユーザーのログイン認証情報を提供することをクライアントに要求します。このタイプの要求は、常に次の 2 つのトークンを返します。

- アクセストークン
- リフレッシュトークン

インスタンスは、ユーザーがアクティブであり、現在ロックアウトされておらず、インタラクティブセッションがあることを確認します。これらの条件のいずれかが false の場合、インスタンスはアクセストークンを生成しません。アクセストークンの有効期限内に行われたアクセス要求は、常に現在のアクセストークンを返します。

注: このタイプの認証権限許可は、送信中にユーザー認証情報を保護するために TLS 暗号化に依存しています。

次の例は、一連のユーザー認証情報を使用してアクセストークンを要求する方法を示しています (読みやすくするためにスペースが追加されています)。

```
$ curl -d"grant_type=password&client_id=be3aeb583ace210011c15b24a43e25d8
&client_secret=client_password
&username=admin&password=admin"
https://instancename.service-now.com/oauth_token.do
```

リフレッシュトークンを使用した要求

インスタンスは、既存のリフレッシュトークンを使用して新しいアクセストークンを作成できます。このタイプの要求は、アクセストークンのみを返します。インスタンスは、新しいアクセストークンを生成する前に、リフレッシュトークンが期限切れになっていないことを確認します。リフレッシュトークンの有効期限内に行われたアクセス要求は、常に現在のリフレッシュトークンを返します。一般に、リフレッシュトークンを送信する方が、ユーザー認証情報を送信するよりも安全です。次の例は、既存のリフレッシュトークンを使用してアクセストークンを要求する方法を示しています (読みやすくするためにスペースが追加されています)。

```
$ curl -d"grant_type=refresh_token&client_id=be3aeb583ace210011c15b24a43e25d8
&client_secret=client_password
&refresh_token=w599voG89897rGVDmdp12WA681r9E5948c1CJTPi8g4HGc4NWaz62k6k1K0FM
xHW40H8yOO3Hoe"
https://instancename.service-now.com/oauth_token.do
```

外部クライアント向けの **OAuth JWT API** エンドポイントの作成 (マシン間の統合)

OAuth JWT ベアラートークンは、エンドユーザーに手動ログインを要求したり、パスワードを共有したりしなくても、受信 JWT 権限許可タイプを使用して、クライアント Web アプリケーションがシームレスにインスタンスの認証を行えるようにします。

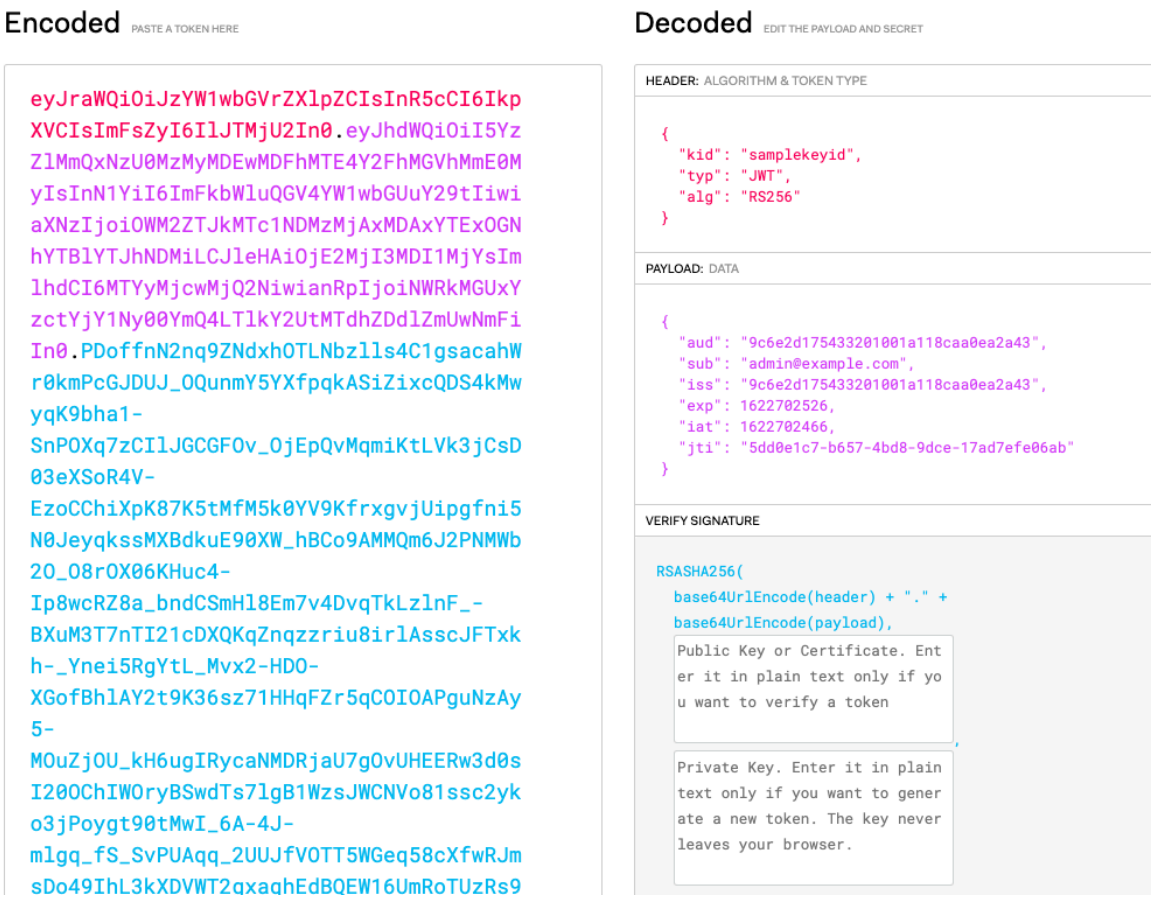
開始する前に

JSON Web トークン (JWT) でサポートされているアルゴリズム:
RS256、RS384、RS512、ES256、ES384、ES512。

クライアント側で次の要求を含む JWT を生成します。

- `aud`：クライアント ID の値と一致する必要があります。
- `sub`：トークンを関連付けるユーザーのメールなど、ユーザー識別子である必要があります。
- `iss`：クライアント ID の値と一致させることをお勧めします。`aud` と `iss` が一致しない場合は、要求検証に iss 値を追加します。
- `exp`：任意の有効期限

デコードされた **JSON Web** トークンの例



ServiceNow での構成

このタスクについて

JWT 権限許可タイプを使用すると、要求にパスワードが含まれないため、Web サービス間のセキュリティを強化できます。たとえば、外部アプリケーションを開発し、トークンを使用して ServiceNow インスタンスへの受信要求を認証できます。

必要なロール：admin

JSON Web トークンの詳細については、「https://jwt.io/」を参照してください。

手順

1. クライアントアプリの公開鍵を **sys_certificate** テーブルに追加します。
2. ServiceNow インスタンスの構成を設定して、受信 JWT を確認します。

   a. 移動先 システム **OAuth** > アプリケーションレジストリー.

   b. [外部クライアント向けの **OAuth JWT API** エンドポイントの作成] を選択します。

   c. トークンに関する情報をフォームに入力します。

[OAuth JWT] テーブル

フィールド	説明
名前	JWT OAuth アクセスが必要なアプリケーションを識別する一意の名前
クライアント ID	自動生成された一意のアプリケーション ID。システムは、このフィールドの値を使用して公開鍵または共有鍵を取得し、JWT を検証します。このフィールドの値は、JWT の発行者要求および対象者要求の値と一致する必要があります。
クライアントシークレット	インスタンスとクライアントアプリケーションまたは Web サイトの両方が、相互通信を許可するために使用する共有シークレット文字列。インスタンスでクライアントシークレットを自動生成するには、このフィールドを空白のままにします。既存のクライアントシークレットを表示するには、ロックアイコンを選択します。 注: [公開クライアント] を選択した場合は、[クライアントシークレット] を省略できます。
ユーザーフィールド	JWT の件名要求の値を照合するためにシステムが使用するユーザー (sys_user) テーブル内のフィールド。たとえば、件名要求の値が user.name@example.com であるトークンを追加する場合は、ユーザーフィールドを [メール] に設定します。このフィールドは、メールフィールドで user.name@example.com の値を検索し、一致するユーザーレコードを受信要求で使用するようにシステムに指示します。
JTI 検証を有効化	トークン交換のたびに新しいトークンを要求する場合に選択します。 デフォルト：オン
アプリケーション	読み込み専用のアプリケーションスコープこのフィールドは自動入力されます。
アクセス可能	Cross-scope Access ポリシー。詳細については、「アプリケーションのアクセス設定」を参照してください。
アクセストークン有効期間	トークンが有効である時間 単位：秒

フィールド	説明
クロックスキュー	JWT の `exp` および `nbf` 要求を検証する際のサーバークロックとクライアントクロック間の許容時間差 単位：秒 デフォルト：300
トークン制限の適用	認証プロファイルを有効にするように設定された API でのみトークンの使用を有効にする場合に選択します。API アクセスポリシーを使用してアクセス許可を設定できます。詳細については、「REST API アクセスポリシーの作成」を参照してください。 デフォルト：未選択。
コメント	アプリケーションに関連付ける追加情報。
公開クライアント	JWT クライアントが公開の場合は、このフィールドをフォームに追加します。選択した場合、クライアントシークレットを含める必要はありません。 デフォルト：未選択。
クライアントタイプ	クライアントのタイプに基づいて、クライアントタイプを選択します。オプション： ▪ **iFrame** 埋め込み ▪ ユーザーとして連携 ▪ サービスとして連携 詳細については、「OAuth および SSO レコードのクライアントタイプの構成」を参照してください。

d. フォームを保存します。

e. [JWT 検証者マップ] 関連リストにレコードを追加して、JWT 署名を検証します。

JWT 検証者マップテーブル

フィールド	説明
名前	JWT マッピングレコードの名前。
Kid	JWT からのキー ID。
共有キー	指定されたキー ID の共有キー。
アプリケーション	読み込み専用のアプリケーションスコープ
システム外部審査済み書	X.509 証明書 (sys_certificate) テーブルの証明書レコード。ステップ 1 でアップロードした証明書。

f. JWT に関連付けられているカスタム要求をすべて [OAuth JWT 要求検証] 関連リストに追加します。

次に示す必須要求には、レコードを追加する必要はありません。

- iss
- aud
- sub
- exp

注:

- aud と iss が一致しない場合は、要求検証に iss 値を追加します。
- 証明書の場合は、複数のキーに関連付けられた複数の検証者マップを追加できます。

[OAuth JWT 要求検証] テーブル

フィールド	説明
自分の外部クライアント (My external client)	OAuth JWT レコードが自動入力されます。
要求値タイプ	請求値のデータタイプ
要求名	追加する要求の名前
要求値	要求の値
アプリケーション	読み込み専用のアプリケーションスコープ

3. JWT トークンを含む cURL 要求を送信して、インスタンスからアクセストークンを取得します。

Example

アクセストークンを要求する cURL コマンドのサンプルを次に示します。

```
$ curl -d"grant_type= urn:ietf:params:oauth:grant-type:jwt-bearer
&client_id=be3aeb583ace210011c15b24a43e25d8
&client_secret=client_password
&assertion=
 eyJraWQiOiJzYW1wbGVrZXIpZCIsInR5cCI6IkpXVCIsImFsZyI6IlJTMjU2In0.eyJhdWQiOiI5YzZl
MmQxNzU0MzMyMDEwMDFhMTE4Y2FhMGVhMmE0MyIsInN1YiI6ImFkbWluQGV4YW1wbG
UuY29tIiwiaXNzIjoiOWM2ZTJkMTc1NDMzMjAxMDAxYTExOGNhYTBlYTJhNDMiLCJleHAiOjE
2MjI3MDI1MjYsImlhdCI6MTYyMjcwMjQ2NiwianRpIjoiNWRkMGUxYzctYjY1Ny00YmQ4LTlkY
2UtMTdhZDdlZmUwNmFiIn0.PDoffnN2nq9ZNdxhOTLNbzlls4C1gsacahWr0kmPcGJDUJ_OQu
nmY5YXfpqkASiZixcQDS4kMwyqK9bha1-SnPOXq7zCIIJGCGFOv_OjEpQvMqmiKtLVk3jCsD03
eXSoR4V-EzoCChiXpK87K5tMfM5k0YV9KfrxgvjUipgfni5N0JeyqkssMXBdkuE90XW_hBCo9AM
MQm6J2PNMWb2O_O8rOX06KHuc4-Ip8wcRZ8a_bndCSmHI8Em7v4DvqTkLzlnF_-BXuM3T7n
TI21cDXQKqZnqzzriu8irlAsscJFTxkh-_Ynei5RgYtL_Mvx2-HDO-XGofBhlAY2t9K36sz71HHqFZr
5qCOIOAPguNzAy5-MOuZjOU_kH6ugIRycaNMDRjaU7gOvUHEERw3d0sI20OChIWOryBSwdT
s7lgB1WzsJWCNVo81ssc2yko3jPoygt90tMwI_6A-4J-mlgq_fS_SvPUAqq_2UUJfVOTT5WGeq58
cXfwRJmsDo49IhL3kXDVWT2gxaqhEdBQEW16UmRoTUzRs9A9sOm18y3skmOVtnEOm-MIJM
FQZ754UMzbiH0ZsMmk1ivCGIjex5J0_IDjKElWF5RHGz3YShCoa4JKDZsqYMvIk1SvzyQXjuFqP
dS2vzg2m1eKGUwr3m6uNs_HflcDystwVdMZ7nLIBG4"
https://instancename.service-now.com/oauth_token.do
```

JWT クライアントが Mobile SDK などの公開クライアントである場合は、要求から client_id パラメーターと client_secret パラメーターを省略できます。client_id および client_secret を省略するアクセストークンを要求する cURL コマンドのサンプルを次に示します。

```
$ curl -d"grant_type= urn:ietf:params:oauth:grant-type:jwt-bearer
&assertion=
 eyJraWQiOiJzYW1wbGVrZXIpZCIsInR5cCI6IkpXVCIsImFsZyI6IlJTMjU2In0.eyJhdWQiOiI5YzZl
MmQxNzU0MzMyMDEwMDFhMTE4Y2FhMGVhMmE0MyIsInN1YiI6ImFkbWluQGV4YW1wbG
UuY29tIiwiaXNzIjoiOWM2ZTJkMTc1NDMzMjAxMDAxYTExOGNhYTBlYTJhNDMiLCJleHAiOjE
2MjI3MDI1MjYsImlhdCI6MTYyMjcwMjQ2NiwianRpIjoiNWRkMGUxYzctYjY1Ny00YmQ4LTlkY
2UtMTdhZDdlZmUwNmFiIn0.PDoffnN2nq9ZNdxhOTLNbzlls4C1gsacahWr0kmPcGJDUJ_OQu
nmY5YXfpqkASiZixcQDS4kMwyqK9bha1-SnPOXq7zCIIJGCGFOv_OjEpQvMqmiKtLVk3jCsD03
eXSoR4V-EzoCChiXpK87K5tMfM5k0YV9KfrxgvjUipgfni5N0JeyqkssMXBdkuE90XW_hBCo9AM
MQm6J2PNMWb2O_O8rOX06KHuc4-Ip8wcRZ8a_bndCSmHI8Em7v4DvqTkLzlnF_-BXuM3T7n
TI21cDXQKqZnqzzriu8irlAsscJFTxkh-_Ynei5RgYtL_Mvx2-HDO-XGofBhlAY2t9K36sz71HHqFZr
5qCOIOAPguNzAy5-MOuZjOU_kH6uglRycaNMDRjaU7gOvUHEERw3d0sI20OChIWOryBSwdT
s7lgB1WzsJWCNVo81ssc2yko3jPoygt90tMwI_6A-4J-mlgq_fS_SvPUAqq_2UUJfVOTT5WGeq58
cXfwRJmsDo49IhL3kXDVWT2gxaqhEdBQEW16UmRoTUzRs9A9sOm18y3skmOVtnEOm-MIJM
FQZ754UMzbiH0ZsMmk1ivCGIjex5J0_IDjKElWF5RHGz3YShCoa4JKDZsqYMvIk1SvzyQXjuFqP
dS2vzg2m1eKGUwr3m6uNs_HflcDystwVdMZ7nLIBG4"
https://instancename.service-now.com/oauth_token.do
```

インスタンスは、その応答でアクセストークンを返します。

```
{
   "access_token":
 "KynMY2H0uwWkRc8g8YLXjnQxWbH5_wbnSiLsnaOoKw61GZkkV0ytZP74uF7hJyjfsWfaaFijq
Qzq2kcABNJxNA",
   "scope": "useraccount",
   "token_type": "Bearer",
   "expires_in": 1799
}
```

注: 受信 JWT 権限許可タイプには、リフレッシュトークンは含まれません。

4. アクセストークンを使用してリソースにアクセスするための REST API 呼び出しを行います。

Example

トークンを使用してインシデントテーブルにアクセスする cURL コマンドは次のとおりです。

```
$ curl -H "Authorization: Bearer
 KynMY2H0uwWkRc8g8YLXjnQxWbH5_wbnSiLsnaOoKw61GZkkV0ytZP74uF7hJyjfsWfaaFijq
Qzq2kcABNJxN"
https://instancename.service-now.com/api/now/v1/table/incident
```

結果
システムが REST 呼び出しでアクセストークンを取得し、要求されたリソースにアクセスできるようにします。

サードパーティトークンを受け入れるための **OAuth OIDC** プロバイダーの構成

受信 API 呼び出しまたはシングルサインオン (マルチプロバイダー SSO) を使用して OAuth OIDC プロバイダーを構成し、サードパーティ OIDC プロバイダーによって生成された ID トークンを受け入れるように設定できます。

始める前に
必要なロール：admin

このタスクについて
ServiceNow AI Platform は、受信 API 呼び出しに加えて、外部シングルサインオン (SSO) 実装を通じて OIDC をサポートします。OIDC プロバイダー構成の例については、「Azure AD の設

定」を参照してください。OIDC プロバイダー構成の SSO 固有の例については、「「Create an OpenID Connect (OIDC) configuration for Single Sign-On (SSO) (シングルサインオン (SSO) 用の OpenID Connect (OIDC) 構成の作成)」を参照してください。

手順

1. 移動先 すべて > システム **OAuth** > アプリケーションレジストリー.
   - [新規] を選択し、**[ID** トークンを検証するために **OIDC** プロバイダーを構成します] を選択して、フォームに記入します。
   - 既存のテンプレートを OIDC プロバイダー (ADFS、Auth0、Azure AD、Google、Okta) に選択して、フォームに入力します。

     注: OIDC プロバイダテンプレートは、デモデータを OAuth 2.0 プラグインを使用してロードした後に利用できます。

フィールド	説明
名前	OAuth OIDC エンティティを識別する一意の名前。
クライアント ID	サードパーティの OAuth OIDC サーバーに登録されているアプリケーションのクライアント ID。この値は、JWT トークンの aud 要求の値と同じにする必要があります。
OAuth API スクリプト	外部 OAuth プロバイダーへの要求および応答のカスタマイズに使用されるスクリプト
OAuth OIDC プロバイダ設定	OIDC プロバイダー (ADFS、Auth0、Azure AD、Google、Okta) を使用して、JWT トークンを検証できます。OIDC プロバイダー構成のレコードを選択して、[ユーザー要求] と [ユーザーフィールド] が適切に設定されていることを検証します。 また、以下のフィールドに入力してください。 ○ **JTI** 要求検証を有効化：有効にすると、プロバイダーによって送信された JTI も ServiceNow JWT トークン検証で検証されます。 ○ **OIDC** メタデータ **URL**：OIDC プロバイダーの既知の構成の詳細。 注: 検証がチェックされない場合、jti が JWT トークンに存在するかどうかにかかわらず、検証できなくなります。トークン内の要求名は jti である必要があります。
クロックスキュー	制約が有効と見なされる秒数。デフォルトは 300 です。
トークン制限の適用	認証プロファイルを有効にするように設定された API でのみトークンの使用を有効にする場合に選択します。API アクセスポリシーを使用してアクセス許可を設定できます。詳細については、「REST API アクセスポリシーの作成」を参照してください。 デフォルト：未選択。
アクティブ	チェックボックスをオンにすると、OAuth アプリケーションがアクティブになります。

フィールド	説明
クライアントタイプ	クライアントのタイプに基づいて、クライアントタイプを選択します。オプション： ○ **iFrame** 埋め込み ○ ユーザーとして連携 ○ サービスとして連携 詳細については、「OAuth および SSO レコードのクライアントタイプの構成」を参照してください。

2. [送信] を選択します。
   レコードは、アプリケーションレジストリー [oauth_entity] テーブルに保存されます。
   インスタンスがトークンと認証コードを発行すると、アプリケーションレジストリ [oauth_entity] テーブルに、外部 **OIDC** プロバイダータイプのレコードが作成されます。詳細については、を参照してください。
3. オプション: レコードの OAuth エンティティプロファイルの関連リストに移動し、スコープのない新規 OAuth プロバイダーのシステム生成デフォルトプロファイルを検証します。
   名前、権限許可タイプ、OAuth スコープなどの OAuth プロバイダープロファイルを変更または追加できます。
4. オプション: レコードの OAuth エンティティスコープの関連リストに移動し、この OAuth プロバイダーで利用可能なすべての OAuth スコープを定義します。
   定義されたスコープは、プロファイルを作成または更新するときに選択できます。定義された各 OAuth スコープには、読み取りスコープや書き込みスコープなど、プロバイダーの仕様からの取得が必要な名前とスコープが含まれています。各スコープは個別に定義する必要があります。
5. オプション: ユーザープロビジョニングレコードの関連リストに移動して、自動ユーザープロビジョニングを有効にします。

オプション	説明
ユーザーを自動プロビジョニング	ユーザーの強制認証を有効にするオプション。
データソースをプロビジョニング	OIDC トークンを ServiceNow ユーザーに変換するために使用するデータソース。ルックアップリストを使用して、事前定義されたデータソーステンプレートを選択し、レコードを開いて変換テーブルのマッピングを構成します。変換マッピングを構成する際に、ソースフィールドは *JWT token*から取得され、ターゲットフィールドは *sys_user* テーブルから取得されます。
プロビジョニングされたユーザーに適用されるユーザーロール	新しくプロビジョニングされた ServiceNow ユーザーに適用されるユーザーロール。

Example: 以下に **REST API** 呼び出しを呼び出す **cURL** 要求の例を示します。
REST API 呼び出しを呼び出します。

次の手順を実行します。

- OpenID Connect プロバイダーにアプリを登録します。
- OAuth OIDC エンティティを構成します。
- OIDC プロバイダーを構成します。

OIDC プロバイダー

OIDC プロバイダー	OIDC プロバイダーの名前。
OIDC メタデータ URL	OIDC メタデータ URL (既知の構成 URL) を指定します。この情報を使用して、 jwks エンドポイントからトークンを検証するための公開鍵をフェッチします。
ユーザー要求	ユーザーテーブルに対して検証される要求。
ユーザーフィールド	ユーザーレコードを識別するユーザー要求。
JTI 要求検証を有効化	有効にすると、ServiceNow JWT トークン検証により、プロバイダーから送信された JTI も検証されます。 **注:** 検証がチェックされない場合、jti が JWT トークンに存在するかどうかにかかわらず、検証できなくなります。トークン内の要求名は jti である必要があります。この情報は、リプレイ攻撃を防ぐために使用されます。

- JWT トークンを取得します。
- REST API 呼び出しを呼び出します。
  - テーブル API またはスクリプト Web サービスにアクセスするための認証ヘッダーの ID トークン。

```
curl -X GET --header "Accept:application/json"
 https://<instance_name>.service-now.com/api/now/table/incident/897b04f2dbd4a300a135364e9d961952 -k
--header "Authorization: Bearer
 eyJraWQiOiJjNTZtZTlXU0xPVnY3UFMwcTg4Qzl1b0lzNjFQYTdmUG4yZFVFOW9RNUg4IiwiYWxnIjoiUlMyNTYifQ.eyJzdWIiOiIwMHVnZDg1ODVkczI1WXpUSjBoNyIsIm5hbWUiOiJpbXJhbiBhbGkiLCJsb2NhbGUiOiJlbi1VUyIsImVtYWlsIjoiaW1yb241NDNAZ21haWwuY29tIiwidmVyIjoxLCJpc3MiOiJodHRwczovL2Rldi05MzQxMjEub2t0YXByZXZpZXcuY29tIiwiYXVkIjoiMG9hZ2Q4bzk3a2lCT3dwd0IwaDciLCJpYXQiOjE1Mzc5MzMzMjYsImV4cCI6MTUzNzkzNjkyNiwianRpIjoiSUQueThVdXpWNUg2bm16SzRsOTI1RFVrQnJoR1o1MmJzVVpGVHRVTEphQjg3ayIsImFtciI6WyJwd2QiXSwiaWRwIjoiMDBvZ2Q4NTgycEFqZDZTemcwaDciLCJub25jZSI6InNub3ciLCJwcmVmZXJyZWRfdXNlcm5hbWUiO
iJpbXJvbjU0M0BnbWFpbC5jb20iLCJnaXZlbl9uYW1lIjoiaW1yYW4iLCJmYW1pbHlfbmFtZSI6ImFsaSIsInpvbmVpbmZvIjoiQW1lcmljYS9Mb3NfQW5nZWxlcyIsInVwZGF0ZWRfYXQiOjE1Mzc5MzAxOTcsImVtYWlsX3ZlcmlmaWVkIjp0cnVlLCJhdXRoX3RpbWUiOjE1Mzc5Mjk2NjF9.OG87SYxWFgHGlhBYby2H79diRm9rlYZTeEkIINRUatwg-p4739htB8xEY-5_t6yU_6k5w10pdgtt5M5QFZRPXVbQZNoGtY-Bxn0BjaimcFgoWfhY_0ldnGTkzN2RYyIHvrf9-yhxg347zvczmLrgMMa_VwG4rxrtE6rUXaIpIeIK5b-Deq8ADz8UTUTKpF_5RWk4X-oh5xK6BLniFHk4ShOZq2v_mjproXwKk5euJKrVrar2lQ4adZCOSTRuTf3ThMO5WDh0sel-82LngXtLzRJJ51IqxAsXns0kJHLLqLtH1hXNRKfwT1ScQoE_OfWm4t0KryI2j4wSMEanFtLXIw"
```

  - ユーザーが認証されると、有効なアプリケーション/json 応答が返されます。そうでない場合は、「ユーザーが認証されていません」という エラーメッセージが返されます。

```
User Not Authenticated
{"error":{"message":"User Not Authenticated","detail":"Required to provide Auth
 information"},"status":"failure"}
```

OAuth および SSO レコードのクライアントタイプの構成

OAuth および SSO レコード関連構成の [クライアントタイプ] フィールドを構成します。

Web UI (インタラクティブログイン)、Iframe 埋め込み、埋め込み、統合などのさまざまなログインタイプのセッションを確立する場合は、OIDC (OAuth エンティティ)、SAML、およびダイジェストレコードにクライアントタイプを構成することで、さまざまなログインに使用できます。

クライアントタイプの選択肢は次のとおりです。

- **iFrame** 埋め込み：サードパーティの Web サイトの Iframe に置かれたインタラクティブ ServiceNow インスタンスに使用できます。たとえば、機密テーブル (sys_user テーブル - ユーザーの電話番号) がある場合、アドミンは ACL を構成するときに、sec 属性 (Iframe 埋め込み) を false に設定し、ユーザーがサードパーティの Iframe 埋め込みセッションのデータ (テーブル情報) にアクセスできないようにすることができます。
- ユーザーとして連携：Slack、Teams などのデスクトップアプリにインストールされている仮想エージェントチャットボットに使用できます。
- サービスとして連携：マシン間の統合 (サービス間の通信) に使用できます。

注:

- OIDC (OAuth エンティティ) の場合：すべての選択肢が利用可能です。
- SAML およびダイジェストの場合：Iframe 埋め込みのみ。フォームを編集し、[クライアントタイプ] フィールドをレコードに追加して、**[iFrame** 埋め込み] クライアントタイプを選択する必要があります。
- [クライアントタイプ] に [なし] を選択した場合、セッションの分類はなくなります。

OIDC (OAuth エンティティ)、SAML、およびダイジェスト用に作成されたすべてのレコードにクライアントタイプフィールドを使用することをお勧めします。これにより、構成が同じでありながらクライアントタイプで区別される各ログイン方法をより適切に制御できます。

フィールドの構成後、対応する構成 (OAuth または SSO) からユーザーがログインするたびに、認証が成功すると、構成されたクライアントタイプに基づいてセッションが考慮され、それに応じてセッションタイムアウトが適用されます。

現在のセッションの場合、対応するセキュリティ属性が含まれています。または、それを利用して、選択したクライアントタイプ内のテーブル固有の情報にユーザーがアクセスできないようにすることができます。詳細については、「OOB (Out-of-Box) セキュリティ属性」を参照してください。

クライアントタイプのセッションタイムアウト

さまざまなクライアントタイプのセッションタイムアウトに関連するシステムプロパティは次のとおりです。

- glide.session_timeout.iframe_embedded
- glide.session_timeout.integration_as_a_user
- glide.session_timeout.integration_as_a_service

OAuth の暗黙的な権限許可

ServiceNow インスタンスは、アクセストークンの暗黙的な権限許可をサポートしています。

暗黙的な権限許可タイプは、*implicit grant code flow*とも呼ばれ、ユーザーエージェント (通常は Web ブラウザーまたはモバイルデバイス) を介してクライアントアプリケーションにアクセストークンを直接付与できます。リフレッシュトークンは付与されません。エンドユーザーは、標準と同様に、インスタンス上の保護されたリソースへのアクセスを許可する必要があります。

OAuth の暗黙的な権限許可フロープロセス

標準の認証コードフロープロセスと同様に、クライアントアプリケーションはインスタンスで制限付きリソースの使用を要求し、エンドユーザーはそれを承認します。要求はインスタンスに送信される URL の形式です。URL には次のようなパラメーターを含める必要があります。

- client_id=<必要なクライアント ID>。これは、クライアントアプリケーションがアクセスしようとしている保護リソースを識別するために必須です。
- response_type=token。これはアクセストークンを直接要求するために必須です (認証コードの要求とは対照的)。暗黙的な権限許可では、値は token になります。標準の認証コードフローの例では、応答タイプは code です。
- redirect_uri=<a URL>：トークンが送信される場所。

認証サーバーは、ユーザーエージェントを通じて、認証コードではなくアクセストークンをクライアントアプリケーションに送信します。

インシデント [incident] テーブルのデータの JSON ペイロードを取得する GET 要求の例を示します。

```
https://myinstance.servicenow.com/oauth_auth.do?response_type=token&redirect_uri={the_redirect_url}&client_id={the_client_identifier}
```

ユーザーがアクセスを許可すると、トークンがリダイレクト (コールバック) URL に含まれます。

```
https/http://{callbackURL}?access_token={the_token}
```

クライアント認証情報

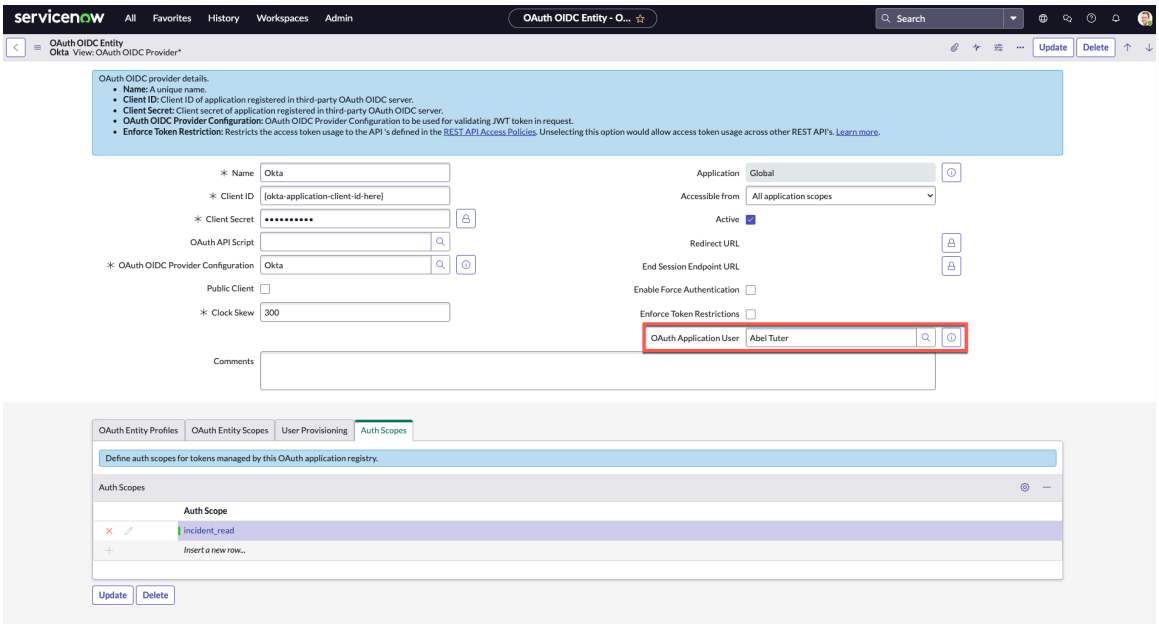
サードパーティの OAuth クライアントから ServiceNow® プラットフォームへのインバウンド統合には、OAuth クライアント認証情報の権限許可タイプを使用します。

アドミニストレーターは、クライアント認証情報 (CC) の権限許可タイプを使用して、サードパーティの OAuth クライアントから ServiceNow プラットフォームへの統合を行えます。

受信クライアント認証情報の権限許可タイプは、システムプロパティを介して制御できる機能です。デフォルトでは、このシステムプロパティは false です。

クライアント認証情報の権限許可タイプを使用するには、次の手順を実行する必要があります。

- glide.oauth.inbound.client.credential.grant_type.enabled システムプロパティを作成します。
- OAuth エンティティフォームに **[OAuth** アプリケーションユーザー]フィールドを追加します。



クライアント認証情報システムプロパティの作成

OAuth インバウンド統合にクライアント認証情報の権限許可タイプを使用するには、glide.oauth.inbound.client.credential.grant_type.enabled システムプロパティを作成します。

始める前に

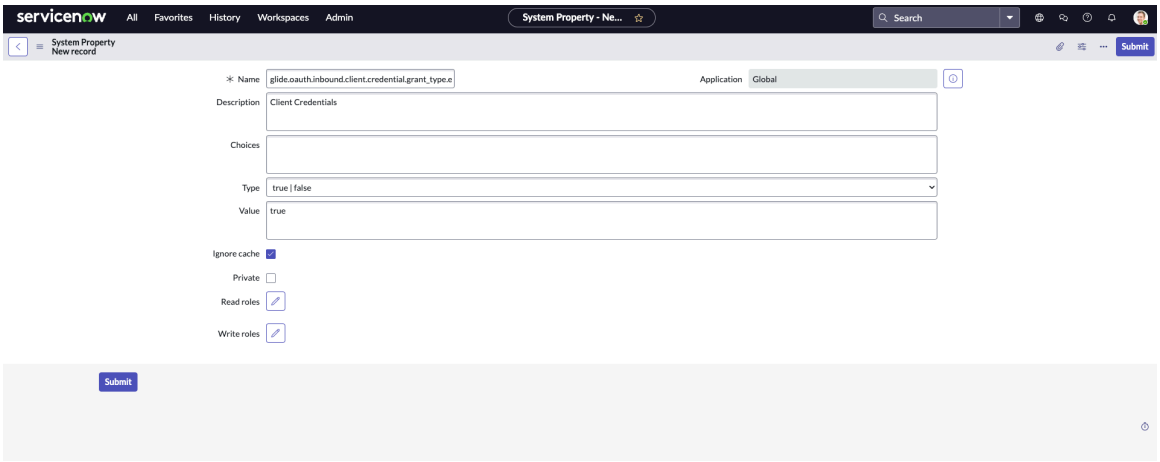
必要なロール：admin

必要なプラグイン：OAuth 2.0

手順

1. ナビゲーションフィルターで、「sys_properties.list」と入力します。
   システム プロパティ [sys_properties] テーブル内のプロパティの全リストが表示されます。
2. [新規] を選択します。
3. フォームの次のフィールドに入力します。

フィールド	説明
名前	作成しているプロパティの名前。ここでは glide.oauth.inbound.client.credential.grant_type.enabled です。
説明	プロパティの機能を説明する簡単な説明文を入力します。
タイプ	リストから適切なデータ型を選択します。ここでは、tTrue/false です。
値	プロパティの目標値を設定します。true にすると、OAuth インバウンド統合のクライアント認証情報の権限許可タイプが有効になります。



注: 選択肢、キャッシュを無視、プライベート、読み込みロール、書き込みロールなどのフォームの他のフィールドは、要件に従って設定できます。

4. [送信] を選択します。

注: [キャッシュを無視] チェックボックスをオンにすると、パラメーターが変更されたときにサーバーのキャッシュがフラッシュされます。

次に、OAuth クライアント (外部クライアント用の OAuth API エンドポイント) を作成し、OAuth クライアントレコードに [OAuth アプリケーションユーザー] フィールドを追加する必要があります。

OAuth アプリケーションユーザーを追加

OAuth インバウンド統合にクライアント認証情報の権限許可タイプを使用するには、OAuth エンティティフォームに [OAuth アプリケーションユーザー] フィールドを追加します。

始める前に
必要なロール：admin

必要なプラグイン：OAuth 2.0

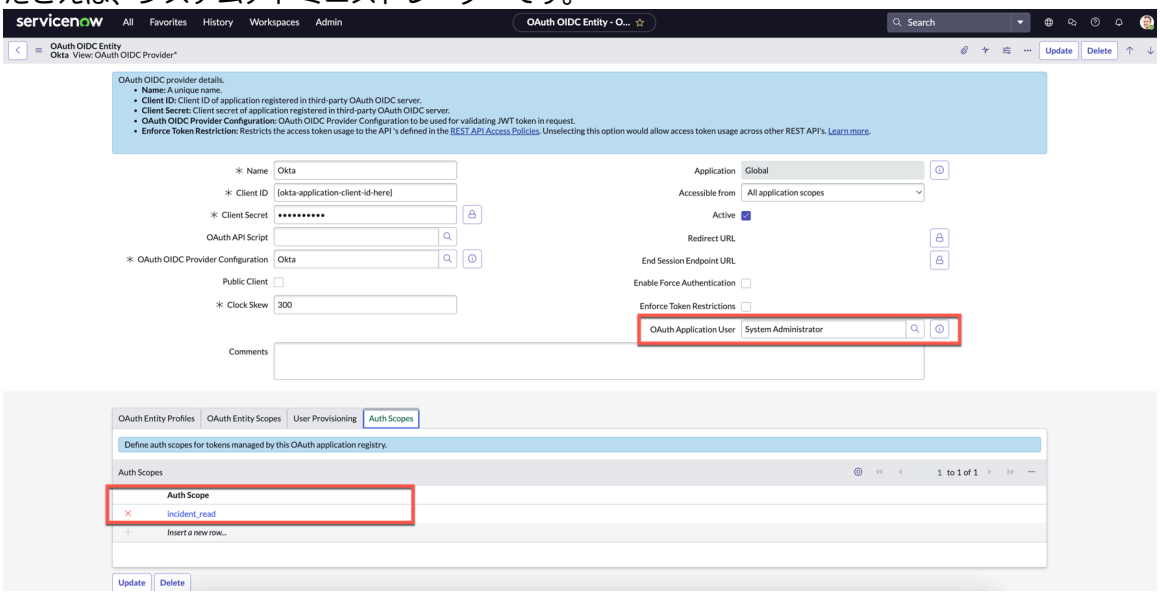
OAuth クライアントを作成する必要があります。詳細については、「インスタンスにアクセスするクライアント用のエンドポイントを作成し ます。」を参照してください。

手順

1. 作成された OAuth クライアントレコードを開きます。
2. ページヘッダーの [その他のオプション] アイコンを選択します。
3. 選択 構成 > フォームデザイン.
4. [フォームデザイン] ページで、フィールドのリストから **[OAuth** アプリケーションユーザー] を追加します。
5. フォームを保存または更新します。
6. **[OAuth** アプリケーションユーザー] にユーザーを選択します。

自動翻訳

たとえば、システムアドミニストレーターです。



注: サードパーティのクライアントに提供されるアクセスを制御するには、クライアント認証情報の権限許可タイプで REST API 認証スコープを使用する必要があります。

7. フォームを保存または更新します。

権限許可タイプがクライアント認証情報、クライアント **ID**、およびシークレットである認証要求はすべて、ServiceNow の関連付けられた OAuth アプリケーションユーザーに渡されます。

注: OAuth クライアントレコードで OAuth アプリケーションユーザーが選択されていない場合、またはクライアント認証情報プロパティが false に設定されている場合、認証要求は渡されません。

OAuth トークンの管理

制限付きリソースへのアクセスを提供する OAuth トークンを開きます。

始める前に

必要なロール：任意のユーザーまたは admin

このタスクについて

インスタンスとサードパーティの OAuth プロバイダーによって発行された OAuth トークンは、[oauth_credential] テーブルに格納されます。

このテーブルの重要な列には次のようなものがあります。

- トークン： ServiceNow インスタンスによって発行されたトークンの値。
- タイプ：トークンがアクセストークンかリフレッシュトークンかを判断します。
- 有効期限：アクセストークンまたはリフレッシュトークンの有効期限が切れる日付/時刻。
- 受信したトークン：サードパーティの OAuth プロバイダーによって発行されたトークンの値。この値は暗号化された形式です。

トークンの有効期限と有効性は次のとおりです。

- アクセストークン：デフォルトで、インスタンスが OAuth プロバイダーであるシナリオでは、インスタンスは 30 分の有効期間でアクセストークンを発行します。
- リフレッシュトークン：デフォルトで、インスタンスが OAuth プロバイダーであるシナリオでは、インスタンスは 100 日の有効期間でリフレッシュトークンを発行します。

手順

1. 次のいずれかのメニューオプションに移動します。
   - セルフサービス > 自分の接続済みアプリ をクリックして、インスタンスのリソースへのアクセスを許可したときにインスタンスが作成したトークンを表示します。
   - システム **OAuth** > トークンを管理 をクリックして、すべてのトークンを表示します。このモジュールへは、アドミンのみがアクセスできます。
2. [名前] をクリックしてトークンを開きます。
3. [アクセスを取り消す] をクリックして、制限付きリソースへのアクセスを防止します。

   アクセスが許可されているスコープや有効期限など、トークンに関するその他の情報を表示することもできます。

   スケジュールページ (sys.trigger.list) から [期限切れの **OAuth** 認証情報を消去 **(Clean Expired OAuth Credentials)]** レコードを選択し、以下を設定できます。

   - **com.snc.platform.security.oauth.is.active**:デフォルトでは、値は true に設定されています。
   - **com.snc.platform.security.oauth.hours.expired.credential.is.keeppt**：要件に基づいて値を設定し、期限切れの oauth 認証情報をシステムに保持する時間数を決定します。
   - **com.snc.platform.security.oauth.day.old.credential.is.keeppt**:要件に基づいて値を設定し、期限切れの OAuth 認証情報をシステムに保持する日数を決定します。

OAuth トークンの取り消し

セキュリティ上の理由から、OAuth アクセス権を取り消したり、トークンをリフレッシュしたりすることができます。

始める前に
必要なロール：admin

このタスクについて
トークンの取り消しは、インスタンスが OAuth リソースサーバーとして機能する状況に関連しています。URL を使用してトークンを取り消すことができます。

手順

oauth_revoke_token.do を使用してインスタンスにアクセスし、アクセストークンまたはリフレッシュトークンを追加します。

例：https://[Your_ServiceNow_Instance]:[port]/oauth_revoke_token.do?token=[access or refresh token] (角かっこ [ ] は削除)

結果
このエンドポイントアクセスには認証は必要ありません。この要求のトークンは期限切れとしてマークされています。

OAuth 送信

OAuth 送信を使用すると、サードパーティプロバイダーからインスタンスにデータをプルできます。

OAuth 統合を管理するには、security_admin ロールが必要です。

次の権限許可タイプに対して送信 OAuth 2.0 を設定できます。

- サードパーティプロバイダーに接続 **(Connect to third-party provider)**：クライアント ID とシークレットを使用して OAuth プロバイダーに送信します。詳細については、「サードパーティ OAuth プロバイダーへの接続」を参照してください。
- **JWT** ベアラー：認証サーバーは、ID とセキュリティ情報をセキュリティドメイン間で共有できるようにする JWT トークンを検証します。詳細については、「JWT ベアラー権限許可タイプで OAuth プロバイダーを設定」を参照してください。
- **SAML2** ベアラー：SAML2 アサーションを生成し、プロバイダーとアクセストークンのアサーションを交換します。

  注: SuccessFactors への送信要求では、**[SAML2** ベアラー] をデフォルトの権限許可タイプとして使用します。 **SAML2** ベアラーの設定方法の詳細については、 「 v4.x.x の設定 」の例を参照してください。

- 認証コード：アクセストークンを取得するためにクライアントに付与され、リソースへのアクセス権を取得するために使用されるコード。このオプションを選択する場合は、認証 URL (認証サーバーの URL) が必要です。
- リソース所有者のパスワード認証情報：リソースへのアクセス権の取得を試行するユーザーのユーザー名とパスワード。
- クライアント認証情報：どちらもアクセストークンを取得するために使用されるクライアント ID とクライアントシークレット。このメソッドはリフレッシュトークンを提供しません。
- **MID** サーバー:MID サーバーは、単一の ServiceNow® インスタンスと外部のアプリケーション、データソース、サービスの間で行われる通信やデータ移動を支援します。MID サーバーを介したアウトバウンド統合要求には、認証コード、リソース所有者のパスワード認証情報、SAML ベアラー、および JWT ベアラー OAuth 権限許可タイプの OAuth を使用します。個人認証は、MID サーバーを介してもサポートされています。

OAuth プロバイダーシナリオ (送信) を構成できます。インスタンスがサードパーティプロバイダーからデータをプルします。

注: トークンを初めて取得するには、ユーザー認証を行う必要があります。その後は、トークンの有効期限が切れるまでユーザーアカウントを使用して認証する必要はありません。

サードパーティ **OAuth** プロバイダーへの接続

クライアント ID とシークレットを OAuth プロバイダーに送信する方法を設定します。

始める前に

必要なロール：admin

手順

1. 移動先 すべて > システム **OAuth** > アプリケーションレジストリー をクリックし、[新規] をクリックします。
2. インターセプターページで、[サードパーティ **OAuth** プロバイダーに接続します] をクリックし、フォームに入力します。

フィールド	説明
名前	サードパーティ OAuth 接続のための一意の名前

フィールド	説明
クライアント ID	サードパーティの OAuth サーバーに登録されているアプリケーションのクライアント ID
クライアントシークレット	サードパーティの OAuth サーバーに登録されているアプリケーションのクライアントシークレット
OAuth API スクリプト	外部 OAuth プロバイダーへの要求および応答のカスタマイズに使用されるスクリプト
ロゴ URL	OAuth アプリケーションのロゴの URL
デフォルトの権限許可タイプ	トークンを確立するために使用するデフォルトの権限許可タイプ次の選択肢があります。 ◦ 認証コード：アクセストークンを取得するためにクライアントに付与され、リソースへのアクセス権を取得するために使用されるコード。このオプションを選択する場合は、認証 URL (認証サーバーの URL) が必要です。 ◦ リソース所有者のパスワード認証情報：リソースへのアクセス権の取得を試行するユーザーのユーザー名とパスワード。 ◦ クライアント認証情報：どちらもアクセストークンを取得するために使用されるクライアント ID とクライアントシークレット。このメソッドはリフレッシュトークンを提供しません。 ◦ **JWT** ベアラー：認証サーバーは、ID とセキュリティ情報をセキュリティドメイン間で共有できるようにする JWT トークンを検証します。 ◦ **SAML2** ベアラー：SAML2 アサーションを生成し、プロバイダーとアクセストークンのアサーションを交換します。 **注:** SuccessFactors への送信要求では、**[SAML2** ベアラー] をデフォルトの権限許可タイプとして使用します。
リフレッシュトークンの有効期間	リフレッシュトークンが有効になる時間 (秒) デフォルトの時間は 86,400,000 秒です。
公開クライアント	パブリッククライアントが認証に PKCE を要求できるようにします。 **注:** PKCE が有効になっている場合、*`Default Grant type`*として使用できるのは認証コードのみです。
コードの課題メソッド	OAuth PCKE ワークフローで使用されるコード課題メソッド次の選択肢があります。 ◦ S256 [デフォルト] ◦ プレーン ◦ なし
コメント	OAuth アプリに関するコメントを追加します。
アプリケーション	このレコードを含むアプリケーションとスコープです。

フィールド	説明
アクセス可能	すべてのアプリケーションスコープから、またはこのスコープからのみアプリにアクセスできるようにします。
アクティブ	チェックボックスをオンにすると、アプリがアクティブになります。
認証 URL	OAuth 認証コードのエンドポイント
トークン URL	OAuth サーバートークンエンドポイント
トークン失効 URL	OAuth サーバートークン失効エンドポイント
リダイレクト URL	OAuth コールバックエンドポイント空白のままにすると、インスタンスによってエントリが自動生成されます。
相互認証を有効にします	トークンの要求と取り消しに相互認証を使用するチェックボックス。この機能では、相互認証プロファイルを指定する必要があります。
認証情報の送信	OAuth クライアントが要求にクライアント認証情報を入力します。 ○ 要求本文内 (フォーム URL エンコード) ○ 基本認証ヘッダー ○ 秘密キー JWT として
クライアントタイプ	クライアントのタイプに基づいて、クライアントタイプを選択します。オプション： ○ **iFrame** 埋め込み ○ ユーザーとして連携 ○ サービスとして連携 詳細については、「OAuth および SSO レコードのクライアントタイプの構成」を参照してください。

3. オプション: レコードの OAuth エンティティプロファイルの関連リストに移動し、スコープのない新規 OAuth プロバイダーのシステム生成デフォルトプロファイルを検証します。
名前、権限許可タイプ、OAuth スコープなどの OAuth プロバイダープロファイルを変更または追加できます。
4. オプション: レコードの OAuth エンティティスコープの関連リストに移動し、この OAuth プロバイダーで利用可能なすべての OAuth スコープを定義します。
プロファイルを作成または更新するときにスコープを選択できます。各 OAuth スコープには、読み取りスコープや書き込みスコープなど、プロバイダーの仕様からの取得が必要な名前とスコープが含まれています。各スコープは個別に定義する必要があります。

JWT ベアラー

JSON Web トークン (JWT) を使用すると、ServiceNow と外部 API プロバイダーの間でユーザーの介入を必要とせずにサーバー間 API のやり取りを設定する機能が有効化されます。

JSON Web トークン (JWT) ベアラー権限許可は、アクセストークンを発行する前に、認証サーバー側の JWT 権限許可ハンドラーによって評価および検証される要求値を含む JSON 文字列です。

JWT ベアラー権限許可タイプを使用すると、送信 REST メッセージの OAuth 2.0 JWT ベアラー権限許可フローを構成できます。

JWT ベアラー権限許可タイプで OAuth プロバイダーを設定

JSON Web トークン (JWT) を使用すると、ServiceNow と外部 API プロバイダーの間でユーザーの介入を必要とせずにサーバー間 API のやり取りを設定する機能が有効化されます。このサポートにより、統合ハブ や JWT を使用する他の自動化タスクが、API およびサービスとさまざまなプロバイダーとの統合を構成できるようになります。

始める前に
必要なロール：admin

このタスクについて
次のタスクは、OAuth 2.0 クライアント認証および認証権限許可に JWT を使用するように ServiceNow を設定する方法を示しています。ServiceNow は OAuth クライアントで、Box や Docusign などの OAuth プロバイダーを構成できます。

手順

1. Java KeyStore 証明書のアップロード
   JWT クライアント認証を有効にするために使用する JKS 証明書をインスタンスに添付します。
2. JWT 署名キーの構成
   Java KeyStore (JKS) 証明書にアサインする JWT 署名キーを作成します。
3. JWT 署名キーを使用した JWT プロバイダーの作成
   JWT プロバイダーを ServiceNow インスタンスに追加します。
4. サードパーティ OAuth プロバイダーへの接続
   ServiceNow アプリケーションレジストリのデフォルトの権限許可タイプとして JWT ベアラーを使用して、サードパーティ OAuth プロバイダーを作成します。
5. OAuth プロファイルの指定
   OAuth プロバイダーの OAuth エンティティプロファイルを開き、JWT プロバイダーをアサインします。

Java KeyStore 証明書のアップロード

JWT クライアント認証を有効にするために使用する Java KeyStore (JKS) 証明書をインスタンスに添付します。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > マルチプロバイダー **SSO** > **x509** 証明書.
2. 必要に応じてフォームに入力します。

オプション	説明
名前	証明書の一意の名前
期限切れ時に通知	証明書の有効期限が切れたときに通知するユーザーを指定します。
有効期限切れ前に警告	証明書の有効期限が切れる前に、証明書マネージャーにメール通知を送信します。
アクティブ	トークン要求に使用する証明書を有効にします。
タイプ	アップロードする証明書のタイプ

オプション	説明
数日中に期限切れ	証明書の有効期限が切れるまでの日数
キーストアパスワード	証明書に関連付けられたパスワード
簡単な説明	

3. [送信] をクリックします。

JWT 署名キーの構成

Java KeyStore (JKS) 証明書にアサインする JSON Web トークン (JWT) 署名キーを作成します。

始める前に
必要なロール：admin

注: JWT キーの一部として **X.509** 証明書 **SHA-1** サムプリント **int (x5t)** をヘッダーに追加したい場合は、フォームを設定し、**X.509** 証明書 **SHA-1** サムプリント **int (x5t)** フィールドを追加する必要があります。

手順

1. 移動先 すべて > システム **OAuth** > **JWT** キー.
2. 必要に応じてフォームに入力します。

オプション	説明
名前	JWT キー署名構成の一意の名前。
署名キーストア	JWT に署名するときに指定されたキーストア。
キー ID	Key ID (kid) は、複数のキーを使ってトークンに署名する際に、どのキーを使用するかを識別するのに役立ちます。 注: このフィールドを設定した場合、キー ID 要求は JWT に含まれます。このフィールドを設定しない場合、JWT にキー ID 要求はありません。
署名アルゴリズム	JWT キーでの署名に使用するアルゴリズム。使用可能なアルゴリズムは RSA 256 のみです。
署名キーパスワード	署名キーに関連付けられたパスワード。
アクティブ	JWT キーエイリアスが JWT プロバイダーからアクティブに参照されることを指定します。

3. [送信] をクリックします。

JWT 署名キーを使用した JWT プロバイダーの作成

JSON Web トークン (JWT) プロバイダーを ServiceNow インスタンスに追加します。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > システム **OAuth** > **JWT** プロバイダー.
2. フォームに入力し、[送信] をクリックします。

オプション	説明
名前	JWT プロバイダー構成の一意の名前
有効期限の間隔 (秒)	JWT プロバイダーによって生成されたトークンの有効期間 (秒)
署名構成	適用する ServiceNow JWT 署名キー構成

JSON Web トークン (JWT) の生成

ServiceNow AI Platform で二者間で安全にクレームを表すための JSON Web トークン (JWT) を作成します。

GlideJWT API は、JWT を生成するスコープ付きのスクリプト可能な API です。JWT を生成する前に必要な引数は 3 つあります。

- JWT プロバイダーの Sys_id
- JSON シリアル化ヘッダー
- JSON シリアル化ペイロード

JWT プロバイダーを構成するときに使用できる JWT API スクリプトは、JWTTokenInternal と JWTTokenRestricted の 2 つです。JWTTokenRestricted スクリプトを使用すると、アドミニストレーターは JWT を生成できるユーザーを設定できます。JWTTokenInternal スクリプトは読み取り専用で、ログインしたユーザーのみが JWT を生成できます。

JWT を生成するには：

- 共有キー (HMAC) または署名キーストア (RSA) を使用して JWT キーを作成します
- JWT キーを参照して JWT プロバイダーを署名構成に関連付けます

API を使用してトークンを作成できます。

JWT プロバイダーを構成するときには、標準およびカスタム要求を使用できます。generateJWT API 署名の一部として、動的なヘッダーとペイロードの要求を渡すことができます。

API をテストするサンプルスクリプト：

```
var jwtAPI = new sn_auth.GlideJWTAPI();
var headerJSON = {  "kid": "a1234"  };
var header = JSON.stringify(headerJSON);

var payloadJSON = { "jti": "testjti", "iss": "testiss", "sub": "testsub" };
var payload = JSON.stringify(payloadJSON);

var jwtProviderSysId = "7a40dde2d5303300964fb7c8f3c14ab5";
var jwt = jwtAPI.generateJWT(jwtProviderSysId, header, payload);

gs.info("JWT:" + jwt);
```

OAuth クライアント API

OAuth クライアント API は、OAuth トークンを要求するメソッドや取り消すためのメソッドを提供します。

OAuth クライアントは次のクラスを提供します。

- GlideOAuthClient ：リフレッシュトークンとアクセストークンの要求と取り消しを行うメソッド。
- GlideOAuthClientRequest ：クライアント要求を処理するためのメソッド。
- GlideOAuthClientResponse ：クライアント応答を処理するためのメソッド。
- GlideOAuthToken ：アクセストークンと、アクセストークンに関する情報を取得するためのメソッド。

OAuthUtil スクリプトインクルードをカスタマイズして、要求パラメーターをインターセプトし、外部 OAuth プロバイダーからの応答を解析することもできます。

スコープ付きスクリプトで OAuth クラスを使用する場合は、`sn_auth` 名前空間識別子を使用します。

デフォルトプロファイルサポート用の OAuth パラメーター

デフォルトのプロファイル機能には、*`setParameter()`* API と同時に使用して、OAuth リクエスター、リクエストのコンテキスト、プロバイダー プロファイルを指定できる一連のパラメーターが必要です。

OAuth プロバイダーのシナリオでは、デフォルトで使用する OAuth プロファイルを OAuth プロバイダーに指示する 3 つのパラメーターを設定する必要があります。これらの 3 つのパラメーターが設定されると、アクセストークンがインスタンスデータベースに保存されます。パラメーターは *`GlideOAuthClientRequest`* と共に使用します。

デフォルトプロファイルサポート用の OAuth パラメーター

パラメーター	説明
oauth_requestor	オブジェクトの sys_id。これはユーザーレコードまたは電子メールアカウントです。
oauth_requestor_context	OAuth リクエスターのコンテキストを提供する記述子。oauth_requestor オブジェクトが保存されたテーブルの名前を使用することをお勧めします。
oauth_provider_profile	デフォルトの OAuth プロファイルレコードの sys_id (「OAuth プロファイルの指定 」を参照)。

値は OAuth プロファイルレコードで設定されているため、パラメーターを使用して権限許可タイプとスコープを設定する必要はありません。パラメーターを使用しない場合は、GlideOAuthClientRequest API メソッド *`setScope`* および *`setGrantType`* を使用できます。追加情報については、「setScope 」と「setGrantType 」を参照してください。

OAuth 2.0 クライアント認証での秘密鍵 JWT サポート

OAuth 2.0 クライアント認証での JWT サポートを行います。

秘密鍵 JWT クライアント認証は、トークンエンドポイントの使用時にクライアントが認証サーバーに対する認証に使用できる認証方法です。

この認証メカニズムでは、公開鍵を登録し、その鍵を使用して JWT に署名したクライアントのみが認証できます。

JWT には、必須の要求値を含める必要があり、オプションの要求値を含めることができます。private_key_jwt 認証用の JWT に必要な要求値の詳細については、OpenID Connect コア ドキュメントの「クライアント認証 (Client Authentication)」セクションを参照してください。

注: 認証トークンは、client_assertion パラメーター値として送信する必要があります。**client_assertion_type** パラメーターの値は、urn:ietf:params:oauth:client-assertion-type:jwt-bearer である必要があります。

JWT トークンを使用した OAuth 2.0 クライアント認証に必要なプラグイン：

- **OAuth 2.0 (com.snc.platform.security.oauth)**：このプラグインは新規インスタンスおよびアップグレードされたインスタンスでアクティブです。インスタンス上でプラグインがアクティブでない場合は、アクティブ化することができます。
- 統合 - マルチプロバイダーシングルサインオンインストーラー **(com.snc.integration.sso.msoi.installer)**：OIDC ベースのシングルサインオンのユースケースの場合。

次の目的で、秘密鍵 JWT を使用した OAuth 2.0 クライアント認証を使用できます。

- OIDC ベースのシングルサインオン
- 送信 OAuth 統合

OIDC ベースの SSO 用の秘密鍵 JWT の設定

OIDC ベースの SSO 統合用に、秘密鍵 JWT を設定します。

始める前に
必要なロール：admin

OIDC ベースの SSO に秘密鍵 JWT を選択する前に、次のタスクを実行する必要があります。

- Java KeyStore 証明書のアップロード：JWT クライアント認証を有効にするために使用する JKS 証明書をインスタンスに添付します。
- JWT 署名キーの構成：Java KeyStore (JKS) 証明書にアサインする JWT 署名キーを作成します。

  注: JWT キーの一部として **X.509** 証明書 **SHA-1** サムプリント **int (x5t)** をヘッダーに追加したい場合は、フォームを設定し、**X.509** 証明書 **SHA-1** サムプリント **int (x5t)** フィールドを追加する必要があります。

- JWT 署名キーを使用した JWT プロバイダーの作成：JWT プロバイダーを ServiceNow インスタンスに追加します。

OIDC ベースの ID プロバイダーの JWT キーを含めるには、以下を行う必要があります。

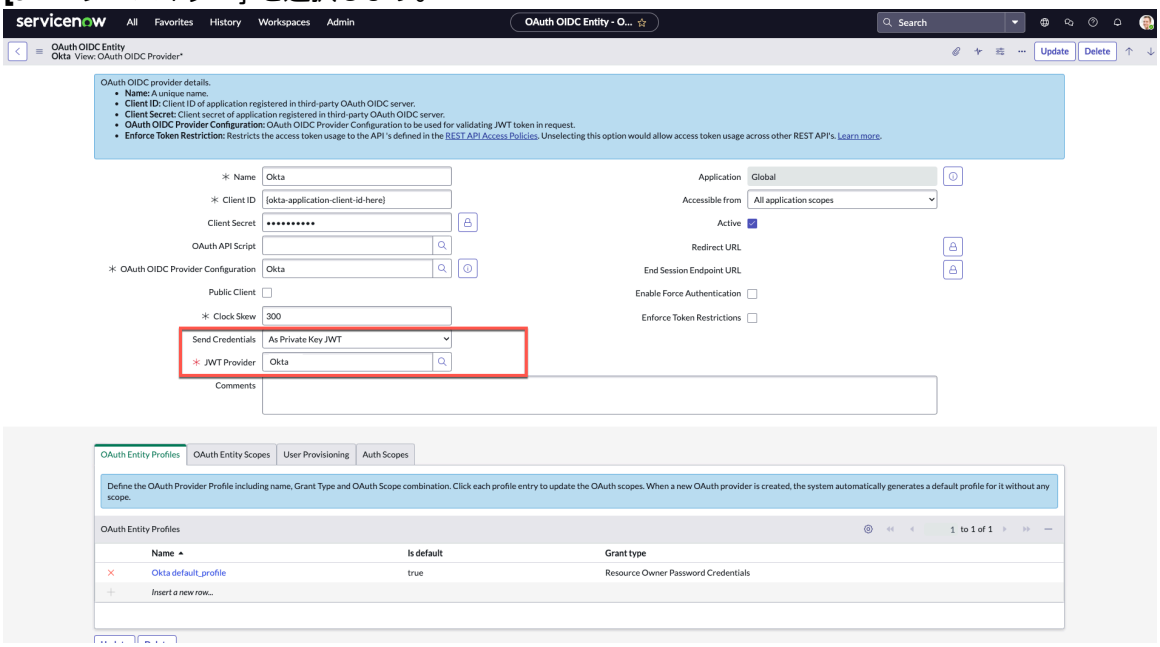
- 統合 - マルチプロバイダーシングルサインオンインストーラー (com.snc.integration.sso.msoi.installer) プラグインをインストールします。
- マルチプロバイダー **SSO** のプロパティを有効にします。詳細については、「複数プロバイダー SSO のプロパティ、テーブル、およびスクリプト」を参照してください。
- OIDC ID プロバイダーを作成します。詳細については、「Single Sign-on (SSO) 用の OpenID Connect (OIDC) 構成の作成」を参照してください。

手順

1. 移動先 すべて > システム **OAuth** > アプリケーションレジストリー.
2. 作成した OIDC ID プロバイダーを選択します。
3. フォームの上部にある 構成 > フォームデザイン.

> 注: OIDC ベースの ID プロバイダー認証リクエスト用に秘密鍵 JWT を使用するには、フォームに [認証情報の送信] フィールドと **[JWT** プロバイダー] フィールドを追加する必要があります。

4. [認証情報の送信] に [秘密キー **JWT** として] を選択します。
5. **[JWT** プロバイダー] を選択します。



ユーザーが認証されると、認証ページに Okta 経由でログインするオプションが表示されます。

送信 **OAuth** の秘密鍵 **JWT** の設定

送信 OAuth 統合用の秘密鍵 JWT を設定します。

始める前に

必要なロール：admin

送信 OAuth 統合用の秘密鍵 JWT を設定する前に、次のタスクを実行する必要があります。

- Java KeyStore 証明書のアップロード：JWT クライアント認証を有効にするために使用する JKS 証明書をインスタンスに添付します。
- JWT 署名キーの構成：Java KeyStore (JKS) 証明書にアサインする JWT 署名キーを作成します。

> 注: JWT キーの一部として **X.509** 証明書 **SHA-1** サムプリント **int (x5t)** をヘッダーに追加したい場合は、フォームを設定し、**X.509** 証明書 **SHA-1** サムプリント **int (x5t)** フィールドを追加する必要があります。

- JWT 署名キーを使用した JWT プロバイダーの作成：JWT プロバイダーを ServiceNow インスタンスに追加します。

手順

1. 移動先 すべて > システム **OAuth** > アプリケーションレジストリー をクリックし、[新規] をクリックします。
2. インターセプターページで、[サードパーティ **OAuth** プロバイダーに接続します] をクリックし、フォームに入力します。

   注: 送信 OAuth 認証リクエスト用に秘密鍵 JWT を使用するには、フォームに [認証情報の送信] フィールドと **[JWT** プロバイダー] フィールドを追加する必要があります。

フィールド	説明
名前	サードパーティ OAuth 接続のための一意の名前
クライアント ID	サードパーティの OAuth サーバーに登録されているアプリケーションのクライアント ID
クライアントシークレット	サードパーティの OAuth サーバーに登録されているアプリケーションのクライアントシークレット
OAuth API スクリプト	社外 OAuth プロバイダーへの要求および応答のカスタマイズに使用されるスクリプト
ロゴ URL	OAuth アプリケーションのロゴの URL
デフォルトの権限許可タイプ	選択：クライアント認証情報：どちらもアクセストークンを取得するために使用されるクライアント ID とクライアントシークレット。このメソッドはリフレッシュトークンを提供しません。
リフレッシュトークンの有効期間	リフレッシュトークンが有効になる時間 (秒) デフォルトの時間は 8,640,000 秒です。
公開クライアント	パブリッククライアントが認証に PKCE を要求できるようにします。 注: PKCE が有効になっている場合、*Default Grant type*として使用できるのは認証コードのみです。
コメント	OAuth アプリに関するコメントを追加します。
アプリケーション	このレコードを含むアプリケーションとスコープです。
アクセス可能	すべてのアプリケーションスコープから、またはこのスコープからのみアプリにアクセスできるようにします。
アクティブ	チェックボックスをオンにすると、アプリがアクティブになります。
認証 URL	OAuth 認証コードのエンドポイント
トークン URL	OAuth サーバートークンエンドポイント
トークン失効 URL	OAuth サーバートークン失効エンドポイント
リダイレクト URL	OAuth コールバックエンドポイント空白のままにすると、インスタンスによってエントリが自動生成されます。

フィールド	説明
相互認証を有効にします	トークンの要求と取り消しに相互認証を使用するチェックボックス。この機能では、相互認証プロファイルを指定する必要があります。
認証情報の送信	選択：秘密キー **JWT** として
JWT プロバイダー	JWT プロバイダーの詳細。ルックアップを使用して JWT プロバイダーを選択できます。

システムにより、秘密 JWT キー認証に使用できる、タイプが OAuth プロバイダーのレコードがアプリケーションレジストリ [oauth_entity] テーブルに作成されます。

送信 **REST** メッセージの作成

送信 REST メッセージを作成して、インスタンスを認証サーバーとして認可します。

始める前に
必要なロール：admin

手順

1. 移動先 システム **Web** サービス > アウトバウンド > **REST** メッセージ をクリックし、[新規] をクリックします。
2. 「」の説明に従って、OAuth アプリケーションレコードのフォームフィールドに入力します。
   次のフィールド値に注意：
   - エンドポイント：認証サーバーであるインスタンスの URL
   - 認証タイプ：**OAuth 2.0**
   - **OAuth** プロファイル：クライアントサーバー用に作成した OAuth プロファイル
3. REST メッセージレコードで、**[OAuth** トークンを取得] をクリックします。
4. トークンを提供するインスタンスで認証します。方法はシングルサインオン統合に応じて変わります。
   使用できるものは次のとおりです。
   - インスタンスへの認証に使用するユーザー名とパスワード。
   - 有効になっている場合は IdP のユーザー名とパスワード。[外部ログインを使用] をクリックして IdP ログイン画面にアクセスします。

     注: IdP ログインページに自動的にリダイレクトするには、glide.authenticate.external システムプロパティを設定する必要があります。

   - MFA が有効になっている場合は、コード。
5. [許可] または [拒否] をクリックして認証を完了し、トークンを発行します。
   この後のプロセスの概要は「OAuth 認証コード権限許可フロー」で説明されています。

個人認証

個人認証を使用すると、 Microsoft OneDrive や Google ドライブ などの OAuth ベースの統合に安全に接続して管理できます。

OAuth 2.0 認証情報

ServiceNow (oauth_2_0_credentials) の OAuth 2.0 認証情報モジュールを使用すると、外部 OAuth 2.0 準拠システムとの接続に使用されるアクセストークンを構成および管理できます。各ユーザーが外部システムに対して独自の ID で認証する必要がある場合は、パーソナル統合タイプを使用します。

認証情報フォームを使用して、 integration_type = Personal で OAuth 2.0 認証情報を構成します。詳細については、「OAuth 2.0 認証情報」を参照してください。

個人認証は、次の OAuth 2.0 権限許可タイプでのみサポートされています。

- 認証コード
- リソース所有者のパスワード認証情報 (ROPC)

クライアント認証情報や JWT ベアラー権限許可などの権限許可タイプは、 integration_type = Personal ではサポートされていません。

注: 認証コードおよび ROPC 権限許可タイプについては、MID サーバーを介した個人認証もサポートされています。詳細については、「 MID を介した OAuth トークンのフェッチ 」を参照してください。

個人認証ダッシュボード

個人の認証情報を使用して、サードパーティ統合に接続します。簡素化された統合インターフェイスを介して、個人認証を表示、認証、取り消し、および更新します。詳細については、「個人認証ダッシュボードの使用 」を参照してください。

注: 個人認証ダッシュボードにアクセスできるのは、ロール sn_personal_auth.personal_auth_user にアサインされたユーザーのみです

個人認証の構成

フローデザイナーの REST ステップで個人 OAuth 認証を設定できます。

始める前に

必要なロール：admin

統合ハブスターターパックインストーラー (com.glide.hub.integrations) 以降のバージョンがインストールされていることを確認します。

このタスクについて

このタスクでは、 ServiceNow® フローデザイナーで REST ステップの個人用 OAuth 認証を構成する方法について説明します。これにより、セッションユーザーの認証情報を使用して REST 呼び出しを実行できるため、安全でパーソナライズされた API アクセスが確保されます。

ユーザーの個人用 OAuth トークンが存在するかどうかを確認する方法については、「 個人の OAuth トークンを取得 (GlideOAuthClient を使用)」を参照してください。

認証情報ページにアクセスできないユーザーの初期トークンを生成する方法については、「 個人認証イニシエーター URL の生成」を参照してください。

手順

1. [アプリケーションレジストリ] に移動し、外部エンドポイントに接続するための OAuth アプリケーションレジストリを作成します。
2. [接続および資格情報エイリアス] に移動し、接続エイリアスを作成します。

このエイリアスは REST ステップで使用されます。詳細については、「接続情報および認証情報エイリアスの作成」を参照してください。

3. **[HTTP(S)** 接続] に移動し、前のステップで作成した接続レコードの外部エンドポイントの詳細を更新します。
4. **OAuth** 認証情報を作成します。
   a. **[OAuth 2.0** 認証情報] に移動します。
   b. 新しい OAuth 認証情報レコードを作成し、ステップ 1 で作成した OAuth プロファイルにリンクします。
5. 認証情報フォームに **[IntegrationType**] フィールドを追加します。
6. ステップ 4 で作成した認証情報の **[IntegrationType**] フィールドを **[Personal]** に更新します。
7. 個人アクセストークンを生成します。
   a. ログインしているユーザーとして、認証情報レコードを開きます。
   b. [**OAuth** トークンの取得] を選択して、個人用トークンを作成します。
   c. トークンを表示および管理するには、[トークンを管理] を選択します。

   注: エンドユーザーがトークンを生成できるように、アプリケーションに UI アクションを追加します。認証情報フォームを直接開くことができるのはアドミンのみです。

8. [アクション] に移動して、ユースケースのアクションを作成します。
9. アクションに REST ステップを追加します。
   ステップ 2 で作成した接続エイリアスを選択します。

   注: REST 送信コールを使用してアクションをテストします。認証情報は、個人統合に使用するようにマークされています。アクションはシステム統合ロールで実行されるため、REST ステップでエラーが表示されることがあります。

10. 新しいサブフローを作成し、ステップ 8 で作成したアクションを追加します。
    a. [サブフローのプロパティ] ウィンドウで、[実行方法] テキストフィールドで [セッションを開始するユーザー] を選択します。
    代わりに [別のユーザーとして実行] テキストフィールドから [システムユーザー] を選択しないでください。
11. サブフローをテストします。
    REST ステップは、セッションユーザー用に作成されたトークンを使用します。サブフローは、FlowAPI を使用して呼び出すこともできます。

    サブフローを呼び出すサンプルスクリプト:

```
try {
   // Execute synchronously in the foreground. Allows access to subflow
 outputs.
   var result = sn_fd.FlowAPI.getRunner()
      .subflow('global.getpersonalincidentssubflow')
      .inForeground()
      .run();

   var outputs = result.getOutputs();
} catch (ex) {
   var message = ex.getMessage();
```

```
    gs.error(message);
}
```

12. 欠落または期限切れのトークンを管理します。

    セッションユーザーがアクセストークンを持っていない場合、REST 要求は **HTTP 401 Unauthorized** ステータスコード 応答を返します。フローを開始する前に、トークンが作成されていることを確認します。

    アクセストークンの有効期限が切れていても有効なリフレッシュトークンが存在する場合、アクセストークンは自動的に更新されます。

個人の OAuth トークンを取得 (GlideOAuthClient を使用)

ユーザーが個人用の OAuth トークンを持っているかどうかを確認します。これを使用して、個人の OAuth 認証情報を必要とする REST ステップまたは統合を実行する前に、有効なアクセスを確認します。

始める前に

必要なロール：admin

このタスクについて

GlideOAuthClient API を使用して、現在ログインしているユーザーの個人用 OAuth トークンが存在するかどうかを確認します。ScopedPersonalAuthAPI を使用して、個人用 OAuth トークンを取得することもできます。詳細については、「PersonalAuthAPI - スコープ指定」を参照してください。

手順

1. 次のサンプル スクリプトを使用して、現在のセッション ユーザーに関連付けられている個人用アクセス トークンを確認します。

```
function dumpToken(token) {
  if (token) {
    gs.info("Access token: " + token.getAccessToken());
    gs.info("Expires in: " + token.getExpiresIn());
    gs.info("Refresh token: " + token.getRefreshToken());
  }
}

var oAuthClient = new sn_auth.GlideOAuthClient();
oAuthClient.setPersonal(true); // Returns the token for the logged-in user

var token = oAuthClient.getToken('<credential_sys_id>', '<oauth_profile_sys_id>');
dumpToken(token);
```

2. <credential_sys_id> と <oauth_profile_sys_id> を適切なレコード値に置き換えます。setPersonal(true) メソッドは、返されたトークンが現在ログインしているユーザーに属していることを確認します。

    注: OAuth リフレッシュトークンとアクセストークンを要求および取り消す方法の詳細については、「Glide OAuth クライアント API ドキュメント」を参照してください。

個人認証イニシエーター URL の生成

個人認証を構成するための認証情報ページにアクセスできないユーザーのために初期トークンを生成します。

始める前に
必要なロール：connection_admin

このタスクについて
connection_admin ロールを持たないユーザーは、[認証情報] ページにアクセスして OAuth トークンを生成することはできません。これらのユーザーは、 oauth_initiator URL と、そのトークンが個人用であり、セッションユーザーに対して要求されることを示す追加パラメーターを使用して、個人用トークンを生成する必要があります。

また、sn_personal_auth プラグインでスコープ対象の PersonalAuthAPI を使用して、イニシエータ URL を生成することもできます。詳細については、「 PersonalAuthAPI - getInitiatorURL(String aliasId)」を参照してください。

注: 個人認証プラグイン (com.snc.sn_ihub_personal_auth) が有効になっている場合は、スコープ付き API を使用してイニシエータ URL を生成します。これ。API は、プラグインがインストールされている場合にのみ使用できます。

手順

1. パスワード権限許可タイプのトークン生成 URL を作成するには、次の形式を使用します。

```
https://<instance-name>.service-now.com/oauth_password_input.do?
sysparm_oauth_requestor_context=oauth_2_0_credential&sysparm_oauth_requestor=
<credential   sys_id>&sysparm_oauth_provider_profile=<OAUTH profile sys_id>
&sysparm_oauth_personal=true
```

2. 認証コード権限許可タイプのトークン生成 URL を作成するには、次の形式を使用します。

```
https:// ://<instance-name>.service-now.com /oauth_initiator.do?
oauth_requestor_context=oauth_2_0_credentials&oauth_requestor=
<credential   sys_id>&oauth_provider_profile=<OAUTH profile sys_id>&respons
e_type=code&personal=true
```

個人認証ダッシュボードのアクティブ化

admin ロールを持っている場合は、統合ハブ の個人認証プラグイン (com.snc.sn_ihub_personal_auth) をアクティブ化できます。このアプリケーションには、デモデータが含まれています。まだインストールされていない場合は、関連する ServiceNow Store アプリケーションとプラグインをインストールします。

始める前に
統合ハブ では、他の ServiceNow AI Platform とは別のサブスクリプションが必要です。

サブスクリプションを購入するには、ServiceNow アカウントマネージャーにお問い合わせください。サブスクリプションを購入すると、特定のプラグインが自動的にアクティブになります。有料プラグインが自動的にアクティブになっていない場合は、インスタンスの [すべてのアプリケーション] リストから手動でアクティブ化できます。

注:

サブスクリプションを購入する前に、Now Support サービスカタログ から非本番インスタンスを要求することによって、その機能を課金なしに評価できます。

必要なロール：admin

このタスクについて
統合ハブ とともに次のアイテムがインストールされます。

手順

1. 移動先 すべて > システムアプリケーション > 利用可能なすべてのアプリケーション > すべて.
2. フィルター基準と検索バーを使用して 個人認証 プラグイン (com.snc.sn_ihub_personal_auth) を検索します。

   名前または ID でプラグインを検索できます。プラグインが見つからない場合は、ServiceNow 担当者から要求する必要があります。
3. [インストール] を選択して、インストールプロセスを開始します。

   注: ドメインセパレーションと代理アドミンがインスタンスで有効になっている場合、管理ユーザーはグローバルドメインに含まれている必要があります。それ以外の場合、次のエラーが表示されます： 「別の操作が実行されているため、アプリケーションのインストールは利用できません： <プラグイン名> のプラグインの有効化 (Application installation is unavailable because another operation is running: Plugin Activation for <plugin name>)」

インストールが完了するとメッセージが表示されます。プラグインとともにインストールされるコンポーネントの詳細については、「 アプリケーションとともにインストールされているコンポーネントの検索 」を参照してください。

ServiceNow インスタンスへの自己登録

外部ユーザーの自己登録を使用して、大量の外部ユーザーをインスタンスにオンボーディングします。この機能によって本人確認を強化して、カスタマーエクスペリエンスを向上させて一般的に使用される登録フローをサポートします。

探索

自己登録の機能とビジネス価値について説明します。

アクティブ化

外部自己登録をアクティブ化する方法を把握します。

自動翻訳

構成

自己登録を設定します。

有効化

自己登録のプロパティについて把握します。

自己登録の詳細

外部ユーザーの自己登録を使用して、大量の外部ユーザーをインスタンスにオンボーディングします。この機能によって本人確認を強化して、カスタマーエクスペリエンスを向上させて一般的に使用される登録フローをサポートします。

外部ユーザー自己登録によって、大規模なユーザーグループがアドミンのサポートなしで ServiceNow アプリに登録できます。たとえば、大学の構内に駐車場を必要とする学生グループが多い大学では、学生自身がキャンパスパーキングアプリに登録できます。各学生には駐車場に固有の制限付きの特権セットが付与されて、自動登録プロセスが完了します。このシステムは各学生に固有の駐車番号を生成して、駐車スペースを確保できるようにします。

自己登録フローは、カスタム ServiceNow アプリ、この場合はキャンパスパーキングアプリで構成されます。必要なテーブルでアプリを構成後、アドミニストレーターは登録前フロー、フィールドマッピング、登録後マッピング、キャプチャ、ロールマッピング、登録後フローを含む登録を構成します。

外部ユーザー自己登録のアクティブ化

admin ロールを持っている場合は、外部ユーザー自己登録プラグイン (com.snc.external_user_self_registration) をアクティブ化することができます。

始める前に

必要なロール：admin

このタスクについて

手順

1. 移動先 すべて > システムアプリケーション > 利用可能なすべてのアプリケーション > すべて.
2. フィルター基準と検索バーを使用してプラグインを検索します。

   名前または ID でプラグインを検索できます。プラグインが見つからない場合は、ServiceNow 担当者から要求する必要があります。
3. [インストール] を選択して、インストールプロセスを開始します。

   注: ドメインセパレーションと代理アドミンがインスタンスで有効になっている場合、管理ユーザーはグローバルドメインに含まれている必要があります。それ以外の場合、次のエラーが表示されます： 「別の操作が実行されているため、アプリケーションのインストールは利用できません： <プラグイン名> のプラグインの有効化 (Application installation is unavailable because another operation is running: Plugin Activation for <plugin name>)」

   インストールが完了するとメッセージが表示されます。プラグインとともにインストールされるコンポーネントの詳細については、「 アプリケーションとともにインストールされているコンポーネントの検索 」を参照してください。

自己登録の外部ロール

不注意で外部ユーザーにアクセスさせないように、すべての外部ユーザーに snc_external ロールを割り当てることができます。

自己登録する外部ユーザーには、最小限の権限を持つ snc_external ロールを割り当てる必要があります。snc_external ロールは、ユーザーが組織の外部にあり、リソースへのアクセス権を持たないことを示します。ただし、ACL を通じて snc_external ロールか、または snc_external ロールを継承する追加ロールに対して明示的に許可された場合は例外です。

デフォルトでは、snc_external ロールを持つユーザーはアクセスできません。

- snc_external ロールまたは公開ロールを継承するロールがないテーブル。
- snc_external ロールまたは snc_external ロールを継承するロールを持たないプロセッサーや UI ページなど、レコードタイプ以外のリソース。
- プラットフォームアナリティクス ダッシュボード

Paris リリース以降、snc_external ロールの明示的な割り当てを強制するには、exclude-list プロパティを有効にする必要があります。プロパティの有効化については、「外部ユーザーに対する今後の内部ロールの割り当てを防止する 」を参照してください。

外部ユーザーのユーザー登録構成の構成

ユーザー登録構成レコードを作成して、外部ユーザーのオンボーディングプロセスをカスタムの ServiceNow アプリケーションにブートストラップします。このフォームは、自己登録プロセスを通じて外部ユーザーをガイドします。

始める前に

- 必要なロール：admin
- 外部ユーザー自己登録のアクティブ化

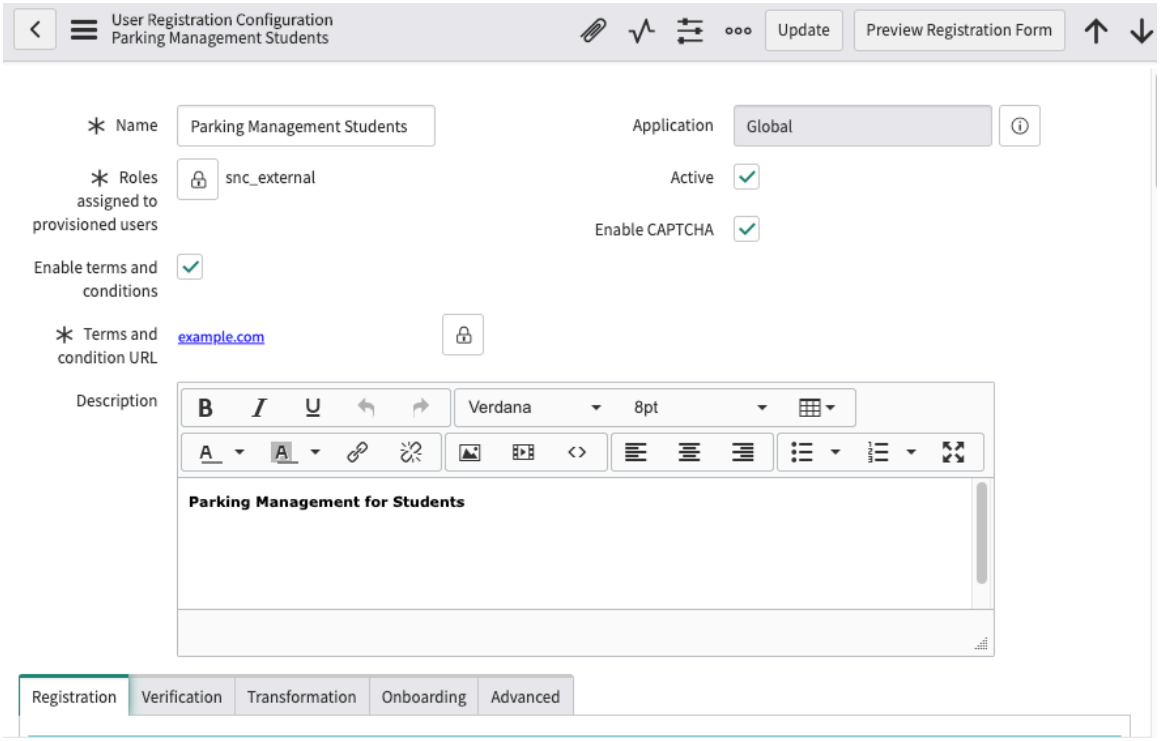
手順

1. 移動先 すべて > 外部ユーザー自己登録 > ユーザー登録構成 をクリックし、[新規] をクリックします。
2. フォームで、フィールドに入力します。

[ユーザー登録構成] フォーム

フィールド	説明
名前	登録フォームの名前。たとえば、学生の駐車場登録の場合は [学生駐車場管理 (Parking Management Students)] などになります。
プロビジョニングされたユーザーへアサインされたロール	プロビジョニングされたユーザーへアサインされたロール。指定されたロールは、snc_external ロールに拡張するか、このロールを含んでいる必要があります。指定されたロールは、snc_external ロールにすることもできます。外部ユーザーの場合は、外部ユーザーを記録するために各ロールに snc_external が必要です。事前設定されたロールがある場合は、ロールのロックを解除してユーザーを検索する際に、そのロールにアクセスできる必要があります。
契約条件を有効化	登録ページに契約条件の URL を追加するオプション
契約条件 URL	登録フォームの契約条件が記載されている公開の URL。このフィールドは、[契約条件を有効化] が選択されている場合にのみ表示されます。
説明	登録フォームの説明。このフィールドは、構成を保存または送信するときにのみ表示されます。このフィールドには、登録フォームに関する情報を追加できます。 注: 説明は、ユーザー登録構成を保存または更新した後にのみ追加できます。
アプリケーション	このレコードを含んでいるアプリケーションです。アプリケーションは自動的に [グローバル] に設定されます。
有効	ユーザー登録構成をアクティブにするオプション。このオプションがデフォルトで選択されています。
CAPTCHA を有効化	登録フォームに CAPTCHA を追加するオプション。デフォルトの CAPTCHA プロバイダーは Google reCAPTCHA です。 注: ユーザー登録の CAPTCHA を有効にするには、「外部ユーザー自己登録のための Google reCAPTCHA の設定」の手順に従います。

3. [送信] をクリックします。
デフォルト設定のユーザー登録構成が作成されます。



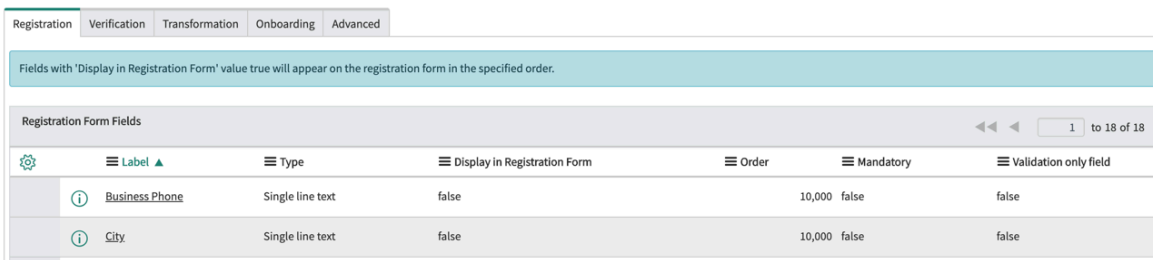
4. オプション: フィールドとその表示順序を表示するように [登録] タブを設定します。

登録フォームフィールド

列	説明
登録フォームに表示	表示したいフィールドをすべて True に設定します。
順番	フォームにフィールドを表示する順序番号を設定します。
必須	必須にしたいフィールドをすべて True に設定します。
検証のみのフィールド	検証にのみ使用するフィールドを設定します。たとえば、登録コードなどです。

デフォルトのフォームフィールドを表示することも、カスタムフォームフィールドを登録フォームに追加することもできます。詳細については、「デフォルトの登録フォームのフィールド」を参照してください。

[登録] タブ



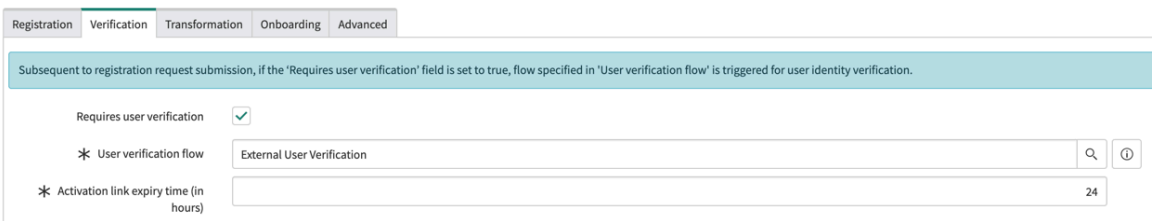
カスタムの登録フォームのフィールドを追加することもできます。詳細については、「カスタム登録フォームフィールドの追加」を参照してください。

5. オプション: [検証] タブを設定して、登録されたユーザーの本人確認を行います。ユーザー検証フローがトリガーされると、ユーザーの登録済みメールアドレスにアクティベーションリンクが送信されます。

[検証] タブのフィールド

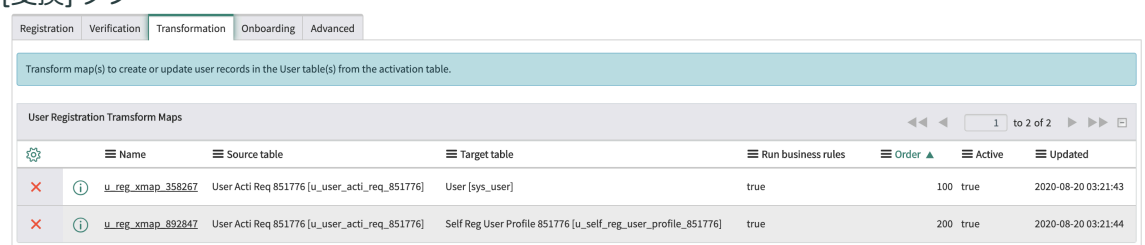
フィールド	説明
ユーザー検証を要求	ユーザー登録後に実行されるユーザー検証サブフローをトリガーするオプション。このサブフローはユーザーの本人確認用です。
ユーザー検証フロー	ユーザーの本人確認に使用されるサブフロー。サブフローは、ユーザー検証を有効にした場合にのみトリガーされます。 (Optional) 外部ユーザーの検証 サブフローはデフォルトで利用可能です。フローデザイナーにデフォルトのサブフローのコピーを作成し、要件に応じて変更できます。詳細については、「フローデザイナー」を参照してください。 注: 新しいタブで外部ユーザーの検証 サブフローをプレビューするには、次のショートカットを使用します。 ○ Macintosh：Command + クリック ○ Windows：Control + クリック
アクティベーションリンクの有効期限 (時間)	アクティベーションリンクが期限切れになるまでの時間数。デフォルト値は 24 です。

[検証] タブ



6. オプション: [変換] タブを設定し、自己登録ユーザーとアクティブ化されたユーザーをマップします。
登録済みユーザーをユーザーアクティビティ要求 [number] テーブルから自己登録ユーザー [number] テーブルに自動的にマップする変換マップ (u_reg_xmap_[number]) は 2 種類あります。デフォルトの変換マップのコピーを作成し、そのマップを要件に応じて変更できます。詳細については、「変換マップ」を参照してください。

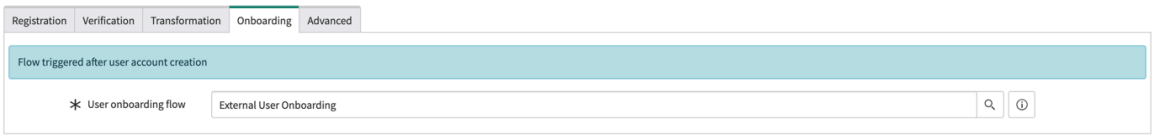
[変換] タブ



7. オンボーディングがアクティブ化されているユーザーのサブフローをトリガーするように、[オンボーディング] タブを設定します。
デフォルトの外部ユーザーのオンボーディング サブフローは、パスワードをリセットするためのリンクを記載したメールをユーザーに送信します。デフォルトのサブフローのコピーを作成し、要件に応じて変更できます。

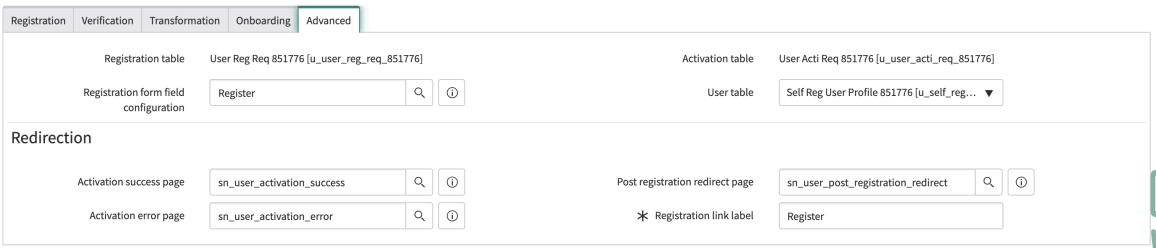
   注: 外部ユーザーのオンボーディングサブフローは、パスワードをリセットするためのリンクを記載したメールをユーザーに送信します。

   [オンボーディング] タブ



8. オプション: 登録フォームのユーザーテーブルとリダイレクトページをマップするように、[詳細] タブを設定します。

   [詳細] タブ

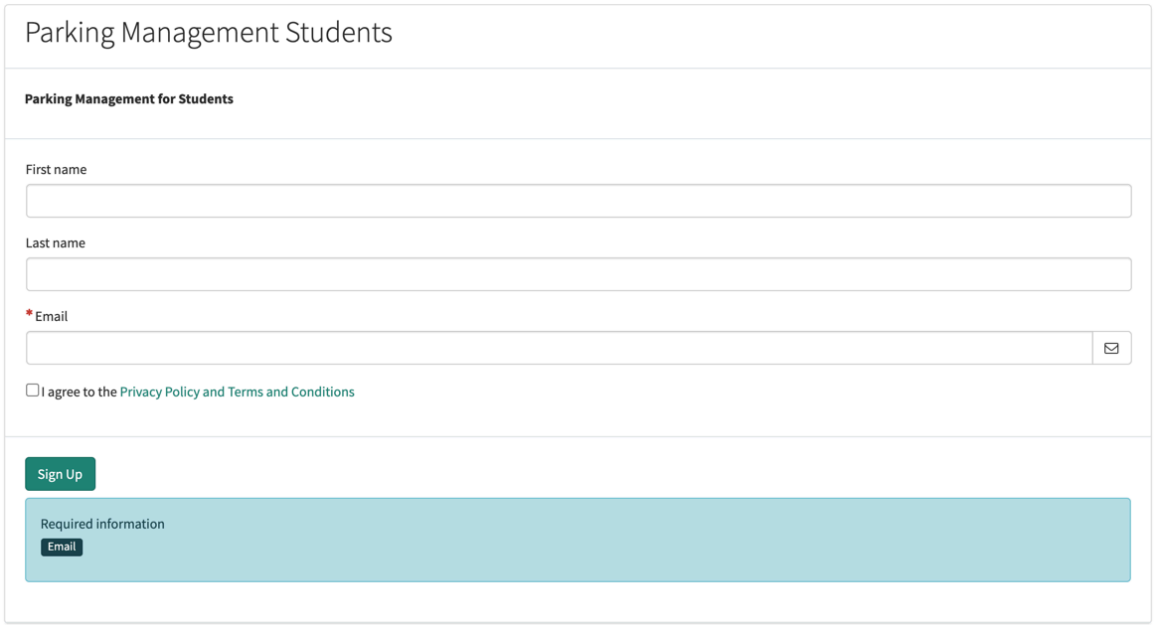


[詳細] タブ

フィールド	説明
登録テーブル	登録フォーム情報が保存されるテーブルの名前
登録フォームフィールド構成	レコードプロデューサーの登録フォームに関連付けられたレコード
アクティベーションテーブル	ユーザーのアクティベーションに使用されるテーブルのラベルと名前。アクティベーションテーブルには、検証が完了したユーザーのレコードが含まれています。
ユーザーテーブル	ユーザープロファイルテーブルのラベルと名前
アクティベーション成功ページ	アクティベーションが成功した後にユーザーがリダイレクトされるページ
アクティベーションエラーページ	アクティベーションに失敗したときにユーザーがリダイレクトされるページ。
登録後リダイレクトページ	登録後にユーザーがリダイレクトされるページ。
登録リンクラベル	サービスポータルからの登録に使用されるボタン名。デフォルト値は [登録] です。

ユーザー登録構成の変更中またはすべての変更が完了した後、[登録フォームのプレビュー] ボタンを使用して登録フォームの変更をプレビューできます。

登録フォームのプレビュー



外部ユーザー自己登録のための **Google reCAPTCHA** の設定

Google reCAPTCHA サービスを使用するには、Google に API キーペアを要求し、関連するシステムプロパティを設定する必要があります。

始める前に

- https://www.google.com/recaptcha で Google から API キーペア (サイトキーとシークレット) を要求します。
- 必要なロール：admin

このタスクについて

手順

1. 移動先 すべて > システムプロパティ > すべてのプロパティ.
2. 次のプロパティを検索し、値を設定します。

プロパティ	値
glide.user.registration.google.recaptcha.secret	シークレットは、アプリケーションと reCAPTCHA サーバー間の通信を許可します。 タイプ：password2
glide.user.registration.google.recaptcha.site_key	登録ページで reCAPTCHA サービスを呼び出すために使用されるサイトキー タイプ：文字列

プロパティ	値
glide.user.registration.captcha.widget	Captcha ウィジェットの Sys_ID タイプ：文字列

デフォルトの登録フォームのフィールド

デフォルトの登録フォームフィールドを使用することも、カスタムの登録フォームフィールドを作成することもできます。

登録フォームフィールド

デフォルトでは、次のフォームフィールドが登録フォームに追加されます。

フィールドラベル	タイプ
勤務先電話	1 行テキスト
市区町村	1 行テキスト
国	選択ボックス
EDU ステータス スタッフ、学生、訪問者を区別するために使用されます。 注: このフィールドは、登録フォームが教育機関用である場合に便利です。	選択ボックス
メール	メール
名	1 行テキスト
性別	1 行テキスト
自宅電話	1 行テキスト
言語	選択ボックス
姓	1 行テキスト
ミドルネーム	1 行テキスト
携帯	1 行テキスト
名前	1 行テキスト
プリフィックス ユーザーの敬称。たとえば「様」、「先生」などです。	選択ボックス
都道府県	1 行テキスト
番地	ワイドな 1 行テキスト
役職	選択ボックス
郵便番号	1 行テキスト

注: デフォルトの登録フォームフィールドは削除できません。

カスタム登録フォームフィールドの追加

ユーザー自己登録フォームにカスタムフィールドを追加できます。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > 外部ユーザー自己登録 > ユーザー登録構成.
2. 必要なユーザー登録構成のレコードを開きます。
3. [登録フォームフィールド] セクションの末尾に移動し、 [新規行を挿入...] をクリックします。
4. [ラベル] 列の下にフィールド名を入力し、チェックマークをクリックします。
   デフォルト値を用いて新しい行が追加されます。要件に基づいてカスタム登録フォームフィールドを設定できます。

登録フォームフィールドの列

列	説明
ラベル	登録フォームに表示されるフィールド名。
タイプ	ユーザーインターフェイス要素のタイプ。サポートされているタイプは次のとおりです。 ○ 1 行テキスト ○ メール ○ 日付 ○ 日付/時刻 ○ はい/いいえ ○ ワイドな 1 行テキスト ○ 複数選択肢 ○ 選択ボックス
登録フォームに表示	登録フォームにフィールドを表示するオプション
順番	フォームフィールドが登録フォームに表示される順序。順序値が最も小さいフィールドが最初に表示され、順序値が最も大きいフィールドが最後に表示されます。デフォルト値は 10,000 です。
必須	登録フォームでフィールドを必須にするオプション
検証のみのフィールド	検証目的に限定してフィールドを使用するオプション。たとえば、登録コードなどです。true に設定すると、このフィールドはユーザーテーブル (sys_user) に保存されません。

5. 変更を保存または更新します。

サービスポータル の外部ユーザー自己登録の有効化

サービスポータル によって外部ユーザーが ServiceNow アプリに登録できるようにします。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > サービスポータル > ポータル.
2. ポータルレコードを開きます。
3. フォームの [外部ユーザー登録構成] フィールドに入力します。

   ユーザー登録構成を選択します。
4. **[Update** (更新)] をクリックします。

結果
ログインウィジェットには、以前に構成した登録フォームへのリンクが含まれています。

ユーザー自己登録要求の確認

サービスポータル からユーザーを登録すると、ユーザーレコードが登録要求モジュールに追加されます。サービスポータル で正常に登録されたユーザーのリストを表示できます。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > 外部ユーザー自己登録 > 登録要求.
   ユーザーレコードリストが表示されます。
2. (オプション) 個々のユーザーレコードを確認し、要件に従って変更します。たとえば、ユーザーレコードフォームのステータスを [処理待ち] から [処理済み] に変更できます。

トークンベースの認証 (ユーザーログイン)

ユーザーがトークンベースの認証を使用してネットワークにアクセスするためのセキュリティメカニズムを強化します。

時間制限付き認証	ダイジェストトークン認証
	
トークンベースの認証は、ユーザーが自分の ID を確認し、代わりに一意のアクセストークンを受け取ることができるプロトコルです。 これは、ユーザーがネットワークにアクセスするためのセキュリティメカニズムを強化するのに役立ちます。	ダイジェストトークン認証は、暗号化されていない HTTP ヘッダー内のユーザー認証情報とダイジェストトークンを渡します。

時間制限付き認証

ServiceNow インスタンスの時間制限付き認証をサポートします。

探索

時間制限付き認証の機能とビジネス価値について学びます。

アクティブ化

時間制限付き認証のアクティブ化の方法について説明します。

チュートリアル:時間制限付き認証

自動翻訳

Zero Trust アクセスのプロパティについて説明します。	

時間制限付き認証の詳細

ServiceNow インスタンスの時間制限付き認証をサポートします。

注: 時間制限付き認証は、ServiceNow インスタンスに固有であり、ユーザー向けのカスタマイズされたリンクは ServiceNow 内でのみ作成できます。

アドミニストレーターは、ServiceNow インスタンスでリンクベースの認証を設定できます。設定されたリンクはメールまたは SMS で共有でき、ユーザーはそれらのリンクを使用してインスタンスにログインできます。

この認証スキームを使用したインスタンスへのログインは、ServiceNow インスタンスで構成された適応認証ポリシーを介してコントロールされます。

時間ベースの認証を使用すると、次のことができます。

- admin は、有効期限まで使用できるリンクベースの認証を設定できます。
- アプリケーションチームは、公開されたスクリプト可能 API を使用して、特定の構成アイテムのリンクとともに使用するノンスを取得できます。リンクの生成はアプリケーションチームが行います。
- アプリケーションチームは、リンクを生成し、既存のチャネルを介してユーザーに送信できます。
- リンクを持つユーザーは、構成の一部として指定された有効期限内であれば、インスタンスに 1 回ログインできます。
- アドミニストレーターは、認証スキーム用に低い権限のロールを設定できます。
- アドミニストレーターは、TLA リンクを使用した認証の第 2 要素として、認証スキーム用の MFA を強制できます。

時間制限付き認証のアクティブ化

Integration - Multiple Provider Single Sign-On Installer プラグインを使用して時間制限付き認証をアクティブ化します。

始める前に

必要なロール：admin

注: 時間制限付き認証は、ServiceNow インスタンスに固有であり、ユーザー向けのカスタマイズされたリンクは ServiceNow 内でのみ作成できます。

手順

1. 移動先 すべて > システムアプリケーション > 利用可能なすべてのアプリケーション > すべて.
2. フィルター基準と検索バーを使用して Time Limited Authentication (*com.snc.authenticate.time_limited_authentication*) プラグインを検索します。

   名前または ID でプラグインを検索できます。プラグインが見つからない場合は、ServiceNow 担当者から要求する必要があります。
3. [インストール] を選択して、インストールプロセスを開始します。

> **注:** ドメインセパレーションと代理アドミンがインスタンスで有効になっている場合、管理ユーザーはグローバルドメインに含まれている必要があります。それ以外の場合、次のエラーが表示されます： 「別の操作が実行されているため、アプリケーションのインストールは利用できません： <プラグイン名> のプラグインの有効化 (Application installation is unavailable because another operation is running: Plugin Activation for <plugin name>)」

インストールが完了するとメッセージが表示されます。プラグインとともにインストールされるコンポーネントの詳細については、「 アプリケーションとともにインストールされているコンポーネントの検索 」を参照してください。

SMS を使用した時間制限付き認証 - Twilio チュートリアル

Twilio を使用して、SMS などの MFA 要素で時間制限付き認証を設定します。

始める前に
必要なロール：admin

次のプラグインが必要です。

- *com.snc.authenticate.time_limited_authentication* (時間制限付き認証)
- *com.snc.authentication.sms_mfa* (SMS を使用したマルチファクター認証)

> **注:** 時間制限付き認証 (TLA) は、ServiceNow インスタンスに固有であり、ユーザー向けのカスタマイズされたリンクは ServiceNow 内でのみ作成できます。

提供されているチュートリアルの指示により、アドミニストレーターは、第 2 要素 (MFA) として SMS を使用したリンクベースのログインを、特定のロールを持つユーザーに提供できます。

構成が正常に完了すると、リンクが生成され、通知 (メール/SMS) チャネルを介してリンクがユーザーと共有されます。リンクをクリックすると、ユーザーロール (構成) に基づいてメールまたは SMS 要素に送信される OTP を指定するように求められます。

> **注:**
>
> - TLA には常に MFA が続き、アドミニストレーターは TLA ログインのために適応認証を使用して MFA を有効にする必要があります。適応認証を使用して MFA を構成する方法の詳細については、「マルチファクター認証コンテキスト」を参照してください。
> - TLA は、権限が制限されているユーザー用に使用する必要があります。

手順

1. Twilio 構成の作成
   a. Twilio テストアカウントを作成します。詳細については、次を参照してください。 Twilio .
   b. 移動先 すべて > 通知 > アドミニストレーション > **Twilio** ダイレクト構成.
   c. アカウント **SID** と認証トークン (Twilioから作成) を入力し、レコードを保存します。

   > **注:** 独自のプロバイダー構成を作成し、それを TLA に使用できます。この例では、Twilio です。MFA プロバイダー構成の作成方法の詳細については、「MFA プロバイダーの構成」を参照してください。

2. 時間制限付き認証 (TLA) レコードを構成して有効にします。

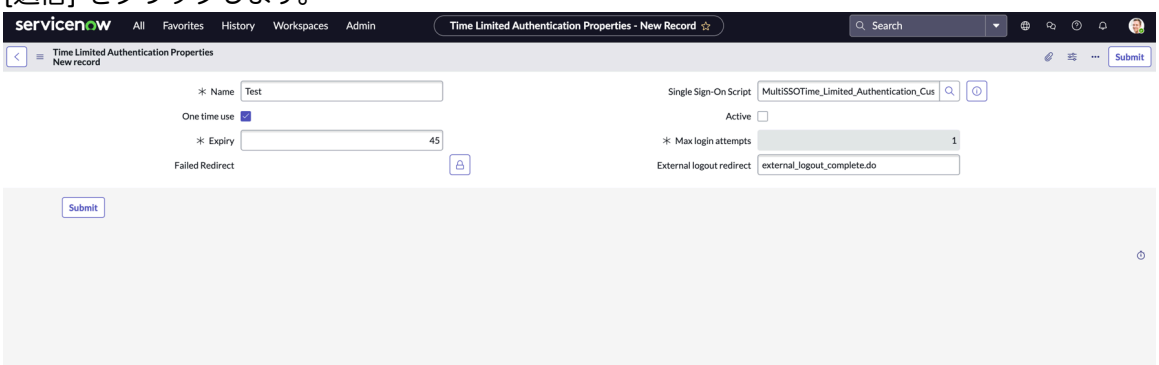
a. 移動先 すべて > 時間制限付き認証構成レコード をクリックし、[新規] をクリックします。

b. フォームの各フィールドに入力します。

時間制限付き認証プロパティ

フィールド	説明
名前	レコードの名前。
1 回のみ使用	TLA リンクを 1 回使用できるようにします。
有効期限	リンクの有効期限を秒単位で指定します。デフォルトは 45 分です。
リダイレクトに失敗しました	認証に失敗した後にユーザーをリダイレクトする URL を入力します。
シングルサインオンスクリプト	使用する SSO スクリプトの詳細。
有効	構成を有効にするオプション。
最大ログイン試行回数	生成された TLA リンクで許可されるログイン試行回数を指定します。**[1** 回のみ使用] チェックボックスをオフにして、最大試行回数を指定します。
外部のログアウトのリダイレクト	ログアウト後にユーザーをリダイレクトする URL を入力します。

c. [送信] をクリックします。



d. 移動先 すべて > マルチプロバイダー **SSO** > アドミニストレーション > プロパティ をクリックし、[マルチプロバイダー **SSO** を有効にする] プロパティを有効にして [保存] します。

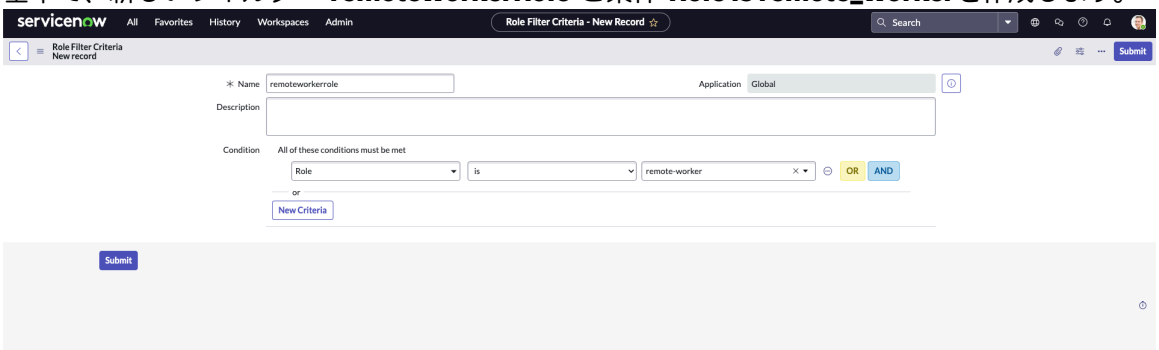
3. 認証後のコンテキストポリシーを使用して、特定のユーザーペルソナにのみ TLA を許可します。

a. [ロール] に移動し、ロールを作成します。例：remote_worker。

b. 有効なメール ID と携帯電話番号を持つユーザーを作成します。ユーザーの作成方法については、「ユーザーの作成」を参照してください。

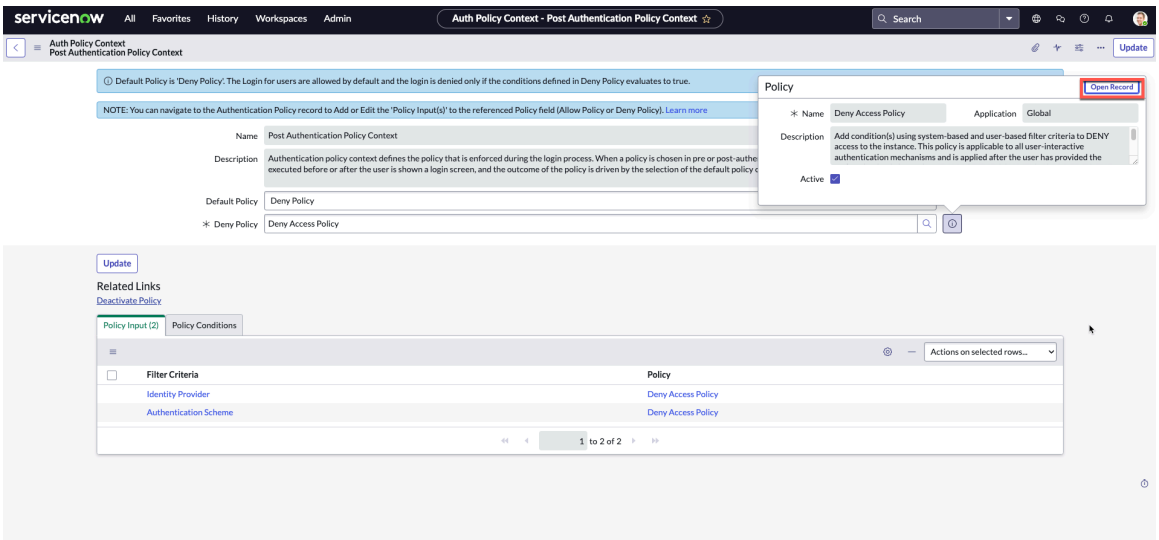
c. ユーザーにユーザーロールをアサインします。ユーザーにロールを割り当てる方法については、「ユーザーへのロールの割り当て」を参照してください。

d. ロールフィルター基準を作成するには、次に移動します: すべて > 適応認証 > ロールフィルター基準で、新しいフィルター **remoteworkerRole** と条件 **Role is remote_worker**を作成します。

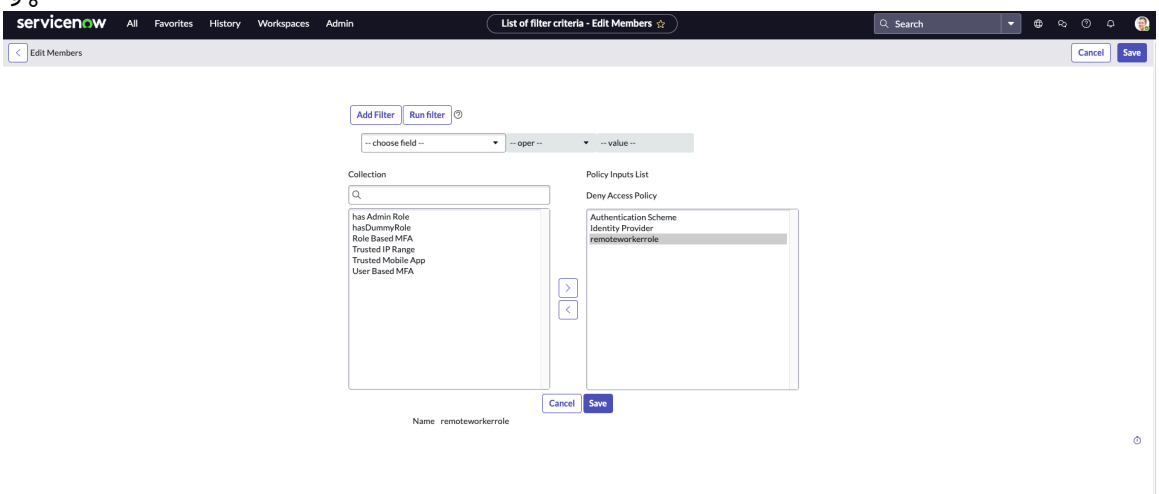


e. IdP およびロールフィルター基準に基づく拒否ポリシーコンテキストに基づいてポリシー条件を追加するには、 すべて > 適応認証 > 認証後のコンテキスト.

f. 情報アイコンをクリックして、レコードを開きます。



g. [ポリシーの入力] で [編集] をクリックし、ロール (remoteworkerrole) を追加して、保存します。

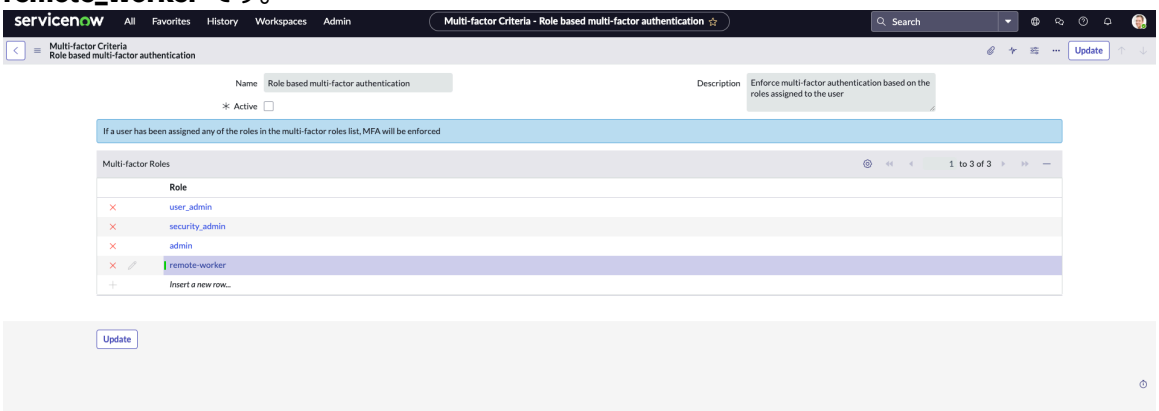


h. [ポリシー条件] で、ポリシー入力の条件を追加し、レコードを送信します。

4. ステップアップ認証ポリシー - MFA コンテキストを構成します。

   a. 移動先 すべて > 多要素基準.

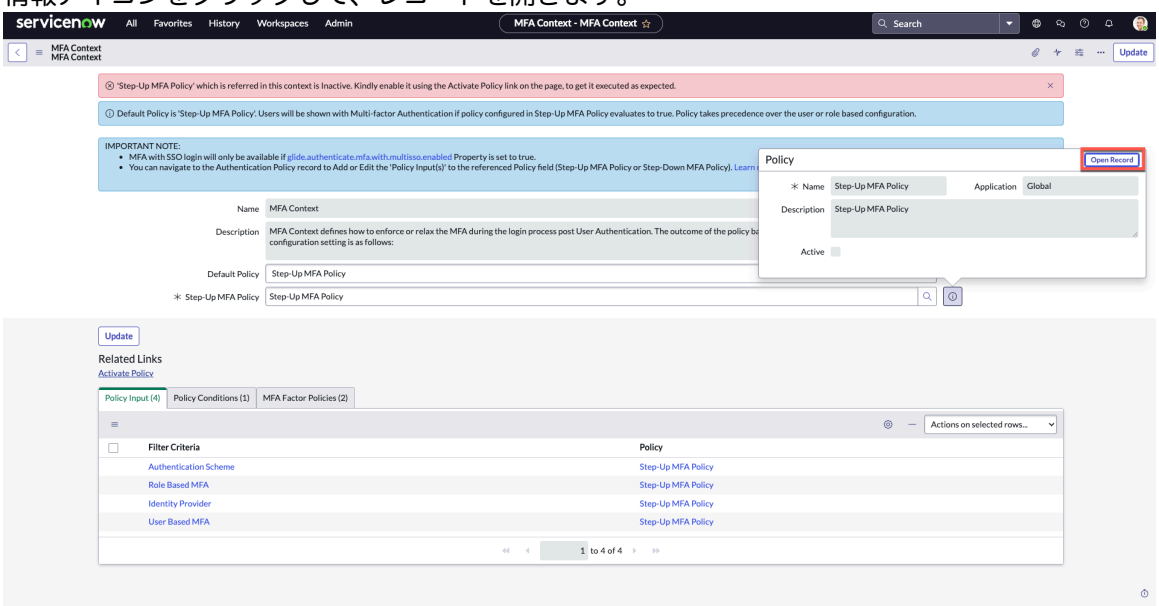
   b. [ロールベースのマルチファクター認証 **(Role-based multi-factor authentication)]** を選択し、[マルチファクターロール] セクションでロールを追加し、更新します。この例では **remote_worker** です。



   c. 移動先 すべて > 適応認証 > **MFA** コンテキスト.

   d. 次のことを確認してください。
   - [デフォルトポリシー] フィールドが [ステップアップ **MFA** ポリシー] になっている
   - ステップアップ MFA ポリシーがステップアップ **MFA** ポリシーになっている

   e. 情報アイコンをクリックして、レコードを開きます。



   f. [ステップアップ MFA ポリシー] フォームの [ポリシーの入力] で、[編集] をクリックします。

   g. ロールベースのマルチファクター認証をリストに追加し、保存します。この例では **remoteworkerrole**です。

h. [ポリシー条件] で、[ロールベースまたはユーザーベースの **MFA** 設定が **true** の場合は **MFA** を強制する **(Enforce MFA if Role based or User based MFA settings are true)]** をクリックします。

i. [ロールベースまたはユーザーベースの MFA 設定が true の場合は MFA を強制する (Enforce MFA if Role based or User based MFA settings are true)] ページで、 [ロールベースの **MFA]** が **true** になっていることを確認します。

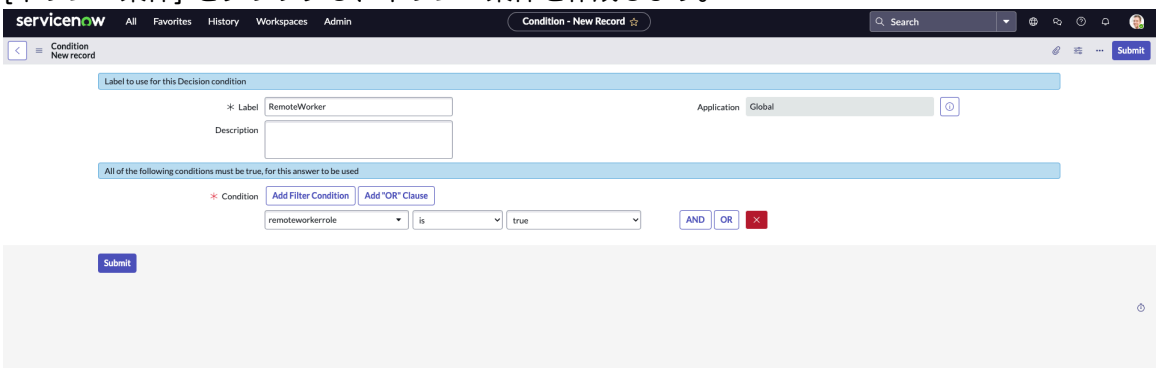
5. MFA 要素ポリシーとして SMS を使用するように MFA を強制します。

a. 移動先 すべて > 適応認証 > **MFA** コンテキスト.

b. [MFA コンテキスト] ページで、**[MFA** 要素ポリシー] をクリックし、**[SMS OTP** を **MFA** 要素ポリシーとして表示 **(Display SMS OTP as an MFA Factor Policy)]** をクリックします。

c. [ポリシーの入力] で [編集] をクリックし、**remoteworkerrole** を追加します。

d. [ポリシー条件] をクリックし、ポリシー条件を作成します。



e. [送信] をクリックします。

TLA リンクが生成され、**remoteworkerole** がロールとしてアサインされたユーザーと共有されます。このリンクは、インスタンスにログインするための第 2 要素として SMS コードを使用するように昇格されます。

6. 他の必要なプロパティを有効にします。

a. 移動先 すべて > 多要素認証 > プロパティ.

b. 次のチェックボックスをオンにします。

- マルチファクター認証を有効にする
- **[SSO** を使用したマルチファクター認証を有効にする]

c. レコードを保存します

d. 移動先 すべて > 適応認証 > 認証ポリシー > プロパティ.

e. [認証ポリシーの有効化 **(Enable Authentication Policy)]** チェックボックスをオンにします。

f. レコードを保存します。

7. TLA リンクの生成 – 例

a. 移動先 すべて > システム定義 > スクリプト – バックグラウンド.

b. ユーザーの sysid と構成 ID を指定して、次の API を使用します。
`var tla=new global.TimeLimitedAuthentication(); gs.info(tla.generateNonce("user_sysid", "config1_sys_id","IAR2"));`

> 注: ソース (IAR2) は必須パラメーターではありません。

c. クエリーパラメーターは次のように返されます。
`nonce=VCeinfboDt0M&glide_sso_id=b3277f1b44351110f8779b5a2d9909f3&user=3b0277d344351110f8779b5`

d. 次の形式で URL を作成します。
`https://<instance-url> /login_with_sso.do?uri=<encoded url>& nonce=2olIQSxdgkjs&glide_sso_id=0c15bf09c3711110c5ec4e483c40dd7a&user=62826bf03710200044e0bfc8`

8. URL をクリックすると、次のようなログイン用の MFA 画面が表示されます。

ダイジェストトークン認証

ダイジェストトークン認証は、暗号化されていない HTTP ヘッダー内のユーザー認証情報とダイジェストトークンを渡します。

探索

ダイジェストトークン認証の機能とビジネス価値について学びます。

構成

ダイジェストトークン認証の詳細

インスタンスは HTTP ヘッダー値を読み取り、ダイジェストトークンの計算されたハッシュ値と比較します。計算されたハッシュ値がダイジェストトークン値と一致する場合、インスタンスはユーザーテーブルで一致する値を検索します。ユーザーテーブルに一致する値がある場合、インスタンスはユーザーが事前認証されているとみなしてログインさせます。

ダイジェストトークン認証は暗号化されていない単純な HTTP ヘッダーよりも安全です。暗号化されていない HTTP ヘッダーを誤って (または意図的に) 変更すると異なるハッシュ値が生成されるからです。ハッシュ値が一致しない場合、インスタンスは要求されたインスタンスへのユーザー アクセスを拒否します。これにより、ユーザーは別のユーザーの認証情報でログインできなくなります。

ダイジェストリンクの有効期限の詳細については、この KB 記事 を参照してください。

注: 時間制限付き認証 (TLA) を使用して、時間ベースの有効期限リンクを設定します。詳細については、「時間制限付き認証」を参照してください。

統合の要件

ダイジェストトークン認証の統合には次のものが必要です。

- Web サーバー
- ローカルネットワークでユーザーを事前認証する SiteMinder または別のシングルサインオンアプリケーション
- 次のいずれかの形式でユーザー認証情報をターゲットインスタンスに渡す Web ページまたはポータル
  - HTTP ヘッダー
  - URL パラメーター
  - cookie
- 次のいずれかのエンコード技術を使用して、ダイジェストトークンを作成してターゲットインスタンスに渡す Web ページまたはポータル
  - SHA1
  - MD5
  - SHA 256 (推奨)

複数プロバイダーシングルサインオン (SSO) のダイジェストプロパティの構成

ダイジェストインストレーションイグジットスクリプトを有効にした後、複数プロバイダー SSO のプロパティを設定します。

始める前に
必要なロール：admin

このタスクについて
複数プロバイダーシングルサインオンを使用していない場合は、標準のシングルサインオンプロパティを設定します。

自動翻訳

手順

1. 移動先 すべて > マルチプロバイダー **SSO** > **ID** プロバイダー.
2. [ダイジェストプロパティ] フォームのフィールドに入力します。

オプション	説明
名前	ダイジェストトークンの名前を入力します。
ユーザー	受信ヘッダーの一致するデータを含む [sys_user] フィールドを入力します。
HTTP ダイジェストヘッダー名	生成された HTTP ヘッダーを入力します。例：DE_USER
HTTP ヘッダー名	作成したダイジェストトークン用に生成した HTTP ヘッダーを入力します。例：SM_USER。
秘密のパスフレーズ	ダイジェストキーのエンコードに使用する秘密キーを入力します。例： 32 文字以上。
[SSO リダイレクトに失敗しました] フィールド	認証に失敗した後にユーザーをリダイレクトする URL を入力します。
外部のログアウトのリダイレクト	ログアウト後にユーザーをリダイレクトする URL を入力します。
シングルサインオンスクリプト	**[MultiSSO_DigestedToken]** を選択します。
クライアントタイプ	クライアントのタイプに基づいて、クライアントタイプを選択します。オプション：**iFrame** 埋め込み。 注: 構成に [クライアントタイプ] フィールドが必要な場合は、フォームを編集してフィールドを追加できます。詳細については、「OAuth および SSO レコードのクライアントタイプの構成」を参照してください。

3. [**Update** (更新)] をクリックします。
4. ダイジェストトークンをデフォルトで true に設定します。
   デフォルトを true に設定すると、SSO に関連付けられたシステムのデフォルトダイジェストトークンレコードが上書きされます。最初の複数プロバイダー SSO 関連の IdP レコードがアクティブ化すると、複数プロバイダー SSO に関連付けられたレコードのみが使用されます。

   ダイジェストプロパティテーブルに存在するダイジェストトークンレコードは、IdP の Sys_ID を追加することによって個別に呼び出すことができます。たとえば、次の認証 URL のダイジェストトークンレコードです。https://<instance_name>.service-now.com/login_with_sso.do?glide_sso_id=<sys_id_of_Digest_token_record>&SM_USER=<user_name>&DE_USER=<digested_token>

ダイジェストトークンの実装例

ここでは、ダイジェストトークンの作成例をいくつか示します。

ダイジェスト認証の実装例

ダイジェストビルドのツール	秘密キーの値	ハッシュメソッド	例
Java	32 文字以上	SHA256	暗号化のためのサンプル Java ダイジェストアルゴリズム
C	sharedKey パラメーターの値	strEncryptionMethod パラメーターの値 (SHA256 または MD5)	サンプル C

暗号化のためのサンプル **Java** ダイジェストアルゴリズム

この Java アルゴリズムは、HTTP ヘッダーからダイジェストトークンを作成する方法を示しています。

この例では、以下を前提としています。

- Web サーバーは Java をサポート
- ハッシュ計算方法は SHA1
- 秘密キーの値は abc123
- 暗号化されていない HTTP ヘッダー名は user_name

別のハッシュ計算メカニズム (MD5 など) を使用するように Java コードを変更するか、秘密キー値または HTTP ヘッダー名を変更します。

```
public class DigestTest
{
   private static final String MAC_ALG = "HmacSHA256";
     fKey  = {32 or more characters};
   public byte[] getDigest(String acct) {
      try {
         byte[] bkey = fKey.getBytes();
         byte[] data = acct.getBytes();
         Mac mac = null;
         try {
            mac = Mac.getInstance(MAC_ALG);
            mac.init(new SecretKeySpec(bkey, MAC_ALG));
         } catch (Exception e) {
            e.printStackTrace();
         }
         byte[] sig = mac.doFinal(data);
         String signature = new String(sig);
         System.out.println("value:" + acct);
         System.out.println("digested value:  " + signature);
         return sig;
      } catch (IllegalStateException e) {
         e.printStackTrace();
      }
      return null;
   }
   public static void main(String[] args) {
      BASE64Encoder encoder = new BASE64Encoder();
      DigestTest test = new DigestTest();
      String userName = "user_name";
```

```
    System.out.println("base 64 digest username: " +
encoder.encode(test.getDigest(userName)));
  }
}
```

サンプル C

この C クラスは、3 つの入力パラメーターからダイジェストトークンを作成する方法を示しています。

- strEncryptionMethod – ハッシュ計算方法 (SHA1、SHA256、または MD5) をリスト
- message – ダイジェストトークンに変換する値をリスト
- sharedKey – 秘密キーをリスト

注: ダイジェストトークンの生成には、SHA256 などのより強力で安全なハッシュアルゴリズムを使用します。

この例では、以下を前提としています。

- Web サーバーは C をサポート
- 他のコードがこのクラスを呼び出し、予想されるパラメーターを渡す

サンプルコード

```
private stringdigestData(string strEncryptionMethod , string message , string sharedKey ) {
        UnicodeEncoding myUnicodeEncoding  = new UnicodeEncoding ( ) ;

        byte [ ] messageBytes  = System. Text. Encoding. ASCII. GetBytes (message ) ;
        byte [ ] sharedKeyBytes  = System. Text. Encoding. ASCII. GetBytes (sharedKey ) ;
        byte [ ] hashedMessage ;

        string b64SHA1Message ;

         if (this. DEBUG ) {
           TextBoxMessage. Text = message ;
           TextBoxSecret. Text = sharedKey ; }

         switch ( (strEncryptionMethod ) )

         { case "SHA1" :

              HMACSHA1 hmacsha1  = new HMACSHA1 ( ) ;
              hmacsha1. Key = sharedKeyBytes ;
              hashedMessage  = hmacsha1. ComputeHash (messageBytes ) ;
              b64SHA1Message  = Convert. ToBase64String (hashedMessage ) ; if (this.
DEBUG ) TextBoxDigest. Text = Convert. ToString (hashedMessage ) ; break ;

            case "MD5" :

              HMACMD5 hmacmd5  = new HMACMD5 (sharedKeyBytes ) ;
              hashedMessage  = hmacmd5. ComputeHash (messageBytes ) ;
              b64SHA1Message  = Convert. ToBase64String (hashedMessage ) ; if (this.
DEBUG ) TextBoxDigest. Text = Convert. ToString (hashedMessage ) ; break ;

            default :
```

```
            b64SHA1Message  = "Unknown Encryption Method" ; break ;

        }


        TextBoxBase64. Text = b64SHA1Message ; return b64SHA1Message ;

    }
```

Web サービスセキュリティ

ベーシック認証、相互認証、または WS-Security を使用してセキュリティを強制します。

Web セキュリティの詳細	相互認証の構成
	
Web サービスセキュリティの機能とビジネス価値について説明します。	SSL (Secure Sockets Layer) 証明書を交換することによって信頼を確立するには、相互認証を使用します。

Web サービスセキュリティの詳細

ベーシック認証、相互認証、または WS-Security を使用してセキュリティを強制します。

ベーシック認証

WSDL 文書または SOAP メッセージの投稿の各要求でベーシック認証を適用するには、*glide.basicauth.required* プロパティを true に設定します。その場合、WSDL または SOAP の各要求には、ベーシック認証 プロトコルで指定されているように「認証」ヘッダーを含める必要があります。要求は非インタラクティブであるため、要求の実行中は常に認証ヘッダーが必要になります。

必要かどうかに関わらずベーシック認証情報を提供すると、Web サービスの呼び出し結果として行われるデータの作成または更新が、ベーシック認証情報で提供されたユーザーの代理で実行されるというメリットがあります。たとえば、インシデントレコードを作成すると、ジャーナルフィールドには、デフォルトのゲストユーザーではなく基本認証ユーザーのユーザー ID が割り当てられます。

認証ヘッダーで大文字化ルールを無視するには、*glide.security.script.include.name.case.insensitive.list* プロパティを使用します。システムプロパティ [sys_properties] テーブルでこのプロパティを変更して、認証を処理するために必要なスクリプトインクルードを追加できます。デフォルトでは、このプロパティには次の値があります。

- BasicAuth
- CustomAuth

必要に応じて他のスクリプトインクルードを追加します。

Perl と SOAP::Lite ライブラリの使用時にベーシック認証を提供するには、次の機能を実装できます。

```
sub SOAP :: Transport :: HTTP :: Client :: get_basic_credentials { return 'user_name' =>
 'password' ; }
```

- C#.NET VS 2005 以前のバージョンを使用している場合は、Credentials オブジェクトを利用できます。

```
System.Net . ICredentials cred  = new System.Net . NetworkCredential ( "user_name",
 "password" ) ;

service . ServiceNow proxy  = new service . ServiceNow ( ) ;
service . get getService  = newservice . get ( ) ;
service . getResponse getServiceResponse  = new service . getResponse ( ) ;

 try {
  proxy . Credentials = cred ;
  getService . sys_id = "bf522c350a0a140701972dbf876f1610" ;
  getServiceResponse  = proxy . get (getService ) ; catch (Exception ex ) { }
```

- C#.NET VS 2008 を使用している場合は、ClientCredentials オブジェクトを利用できます。

```
Demo_Incident. ServiceNowSoapClient client  = new Test08WebService . Demo_Incident .
 ServiceNowSoapClient ( ) ;
client . ClientCredentials . UserName . UserName = "admin" ;
client . ClientCredentials . UserName . Password = "admin" ;
```

次に、app.config ファイルで以下を探し、「None」を「Basic」に変更します。

```
<transport clientCredentialType= "None" proxyCredentialType= "None" realm= "" />
```

- VB .NET を使用している場合は、Credentials オブジェクトを利用すると次のようになります。

```
Sub Main()
        Dim cred  As New System.Net.NetworkCredential( "user_name",  "password")

        Dim proxy  As New VB_Democm.incident.ServiceNow
        Dim getIncident  As New VB_Democm.incident. get Dim getResponse  As New
 VB_Democm.incident.getResponse

       proxy.Credentials = cred

       getIncident.sys_id =  "[your sysID here]"

       getResponse = proxy. get(getIncident)
```

自動翻訳

```
End Sub
```

ベーシック認証がオンになっており、認証情報が提供されていない場合の結果の応答は、次のようになります。

```
<html> <head > <title >Apache Tomcat/5.0.28 - Error report </ title > <style > <!--H1
{font-family:Tahoma,Arial,sans-serif;color:white;background-color:#525D76;font-size:22px;}
 H2
{font-family:Tahoma,Arial,sans-serif;color:white;background-color:#525D76;font-size:16px;}
 H3
{font-family:Tahoma,Arial,sans-serif;color:white;background-color:#525D76;font-size:14px;}
   BODY {font-family:Tahoma,Arial,sans-serif;color:black;background-color:white;}
 B {font-family:Tahoma,Arial,sans-serif;color:white;background-color:#525D76;}    P
{font-family:Tahoma,Arial,sans-serif;background:white;color:black;font-size:12px;}   A
{color : black;}   A.name {color : black;}   HR {color : #525D76;}--> </
style > </ head > <body > <h1 >HTTP Status 401 -\ </ h1 > <HR size = "1" noshade =
"noshade" > <p >< b >type </ b > Status report </ p > <p >< b >message </ b > <u >< /
u >< / p > <p >< b >description </ b > <u >This request requires HTTP authentication ().
</ u >< / p > <HR size = "1" noshade = "noshade" > <h3 >Apache Tomcat/5.0.28 </ h3 >
</ body > </ html >
```

相互認証の構成

相互認証では、SSL (Secure Sockets Layer) 証明書を交換することによって信頼を確立します。

SSL ハンドシェイク中に、サーバーはクライアントに証明書を提示します。その後、サーバーの構成によっては、クライアントからの証明書をサーバーが要求する場合があります。サーバーとクライアントの両方が証明書の検証手順を実行して、提示された証明書の信頼性と整合性を確保します。

検証が成功した後、HTTPS 接続を開始する前に確認応答が交換されます。

注: カスタム HTTPS プロトコルプロファイルを使用して相互認証を有効にする方法については、「プロトコルプロファイルの作成」を参照してください。

アドミニストレーターは、認定要求を実行する前に、クライアントキーストアを設定して証明書を生成する予備作業を行います。

警告: この機能で相互認証が可能になるのは、送信 HTTPS 接続だけです。

キーストアの作成

インスタンスは現在、秘密鍵と公開証明書のペアを含む Java キーストアファイルのアップロードをサポートしています。公開証明書にはルート証明書を含む完全なチェーンが含まれています。

クライアントキーストアを設定するには、

- 信頼できる認証局 (CA) によって署名された証明書が必要です。
- API エンドポイントプロバイダーがキーストアの生成を支援する場合があります。

キーストアを生成する必要がある場合、このプロセスには、コマンドラインインターフェイスコマンドを使用して新しいキーストアファイルを生成し、証明書署名要求 (CSR) を作成して、署名済み証明書をインポートするいくつかの手順が含まれます。ルート証明書や中間証明書はいずれも、ドメインのプライマリ証明書をインポートする前にインポートする必要があります。ステップバイステップの手順は次のとおりです。

1. Java キーストアとキーペアを生成します。

```
keytool -genkey -alias mydomain -keyalg RSA -keystore my.keystore
```

2. 既存の Java キーストアの CSR を生成します。

```
keytool -certreq -alias mydomain -keystore my.keystore -file mydomain.csr
```

3. CSR を CA 署名機関に送信します。CA 署名機関は証明書ファイルに署名し、署名済み証明書とともに中間証明書とルート証明書を含む証明書ファイルを返します。
4. ルートまたは中間認証局 CA 証明書を既存の Java キーストアにインポートします。

```
keytool -import -trustcacerts -alias root -file Thawte.crt -keystore my.keystore
```

> **注:** すべての証明書を 1 つのファイルにバンドルしてインポートできます。これが推奨されるオプションです。このようにすると、手順 5 をスキップできます。
>
> ```
> keytool -import -alias mydomain -file merged.crt -keystore my.keystore
> ```

5. 署名済みプライマリ証明書を既存の Java キーストアにインポートします。

```
keytool -import -trustcacerts -alias mydomain -file mydomain.crt -keystore my.keystore
```

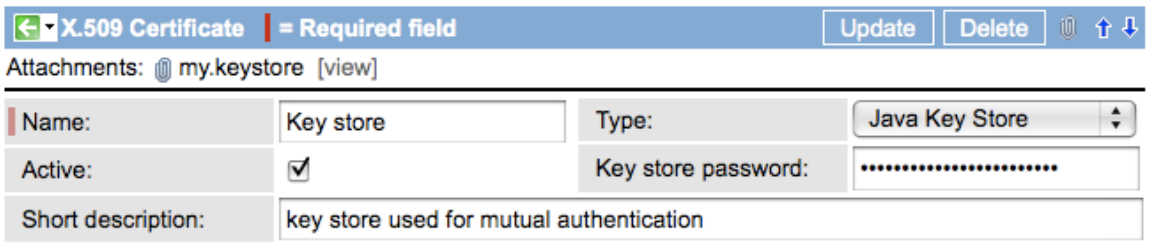
キーストアの設定

キーストアが作成されたので、証明書テーブルにアップロードできます。対象： システム定義 > 証明書 ページで、[新規] をクリックして次のフィールドを設定します。

- 証明書の名前を入力します。
- キーストアをアクティブとして格納します。
- タイプを **[Java** キーストア] に設定します。
- キーストアパスワードを入力します。これは、キーストアの作成に使用されたパスワードです。

[送信] をクリックして Java キーストアエントリを作成します。

キーストア



信頼できるサーバー証明書の指定

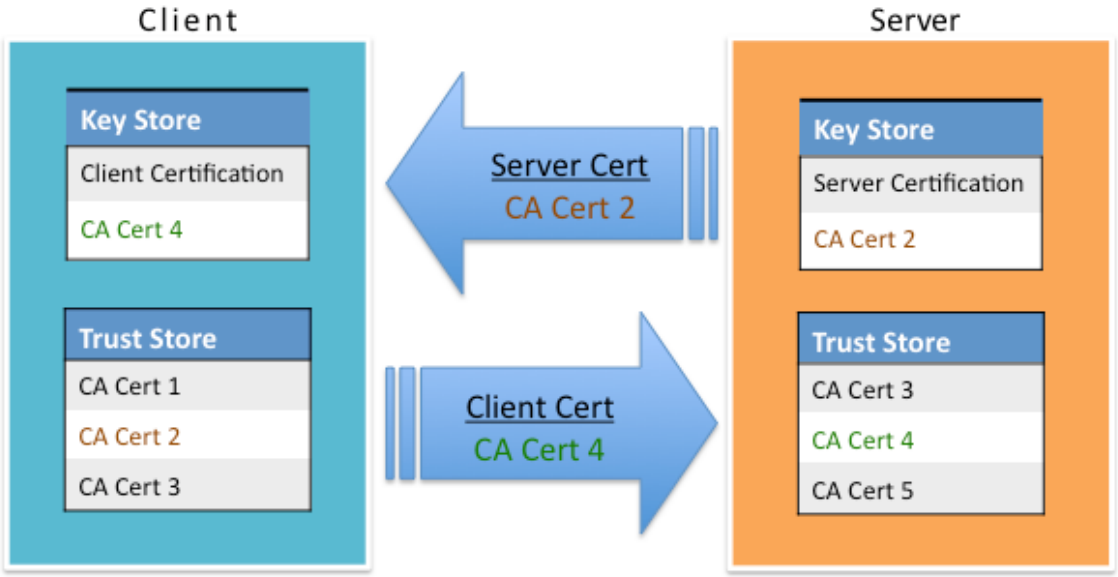
送信 SSL 接続 (HTTPS Web サービス呼び出し) では、サービスプロバイダーによって提供される証明書を指定して、SSL 接続中のサービスプロバイダーの妥当性を保証することができます。たとえば、証明書で識別される安全なサービスにブラウザが接続しようとしている場合などです。

信頼できるサーバー証明書をアップロードすることで、ServiceNow により、接続しているサービスの妥当性と安全性を確保できます。

「信頼ストア証明書」タイプで新しい証明書エントリを作成し、DER 形式の証明書を添付するか、その PEM 形式をコピーして **[PEM** 証明書] フィールドに貼り付けます。

プロトコルプロファイル

証明書の交換



- クライアントが認証のためにサーバー証明書を要求すると、証明書署名要求 (CSR) が生成されます。
- CSR に応答するために、サーバーは 2 つの一意の暗号化キー：サーバーへのメッセージを暗号化するために使用される公開鍵とメッセージを復号化するために使用される秘密鍵を生成します。両方のキーがキーストアに保持されます。
- キーを使用して、クライアントの安全なメッセージを復号化し、サーバーで読み取りができるようにします。HTTPS になるすべての送信接続は、キーストアを確認してその公開認定を提供することで、認定を検証し、トラストストア証明書を使用して相互の信頼性を検証します。
- クライアントとサーバーとの間で安全なリンクを確立するために、サーバーは証明書を対応する秘密鍵と照合します。サーバーのみが秘密鍵にアクセスできるため、サーバーはクライアントからのデータを復号化できます。

注: カスタム HTTPS プロトコルプロファイルを使用して相互認証を有効にする方法については、「プロトコルプロファイルの作成」を参照してください。

サーバーは証明書を送信して応答します。これがクライアントが受け入れる証明書ですか？そうである場合、証明書を受け入れるメッセージがサーバーに送信され、安全なチャネルが開始されます。証明書が承認されない場合は、認定にルート権限が必要であることを意味します。

アクセス制御リストのルール

アクセス制御リスト (ACL) のルールは、まず要件のセットをユーザーに要求し、その後でユーザーとやり取りできるようにすることで、データへのアクセスを制限します。

ACL の詳細

アクセス制御リスト (ACL) について説明します。

ACL の構成

ACL を設定します。

コンテキスト依存セキュリティマネージャー

コンテキスト依存セキュリティマネージャーについて学習します。

ACL の詳細設定

ACL の詳細設定とツールについて説明します。

自動翻訳

アクセス制御リストの詳細

アクセス制御リスト (ACL) の詳細について説明します。

すべてのアクセス制御リストのルールは、次のものを指定しています。

- ACL を定義する「意思決定タイプ」、「ルールタイプ」、および「操作」
- 保護される「オブジェクト」
- オブジェクトにアクセスするために必要な「条件」

ACL のコンポーネント

意思決定タイプは、条件が満たされた場合にユーザーのオブジェクトへのアクセスを許可するか、条件が満たされない限りオブジェクトへのアクセスを拒否するかを定義します。

決定タイプ	説明
Deny-Unless (次の場合を除き却下)	条件を満たさない限りアクセスを明示的に拒否することで、リソースへのアクセスを制限します。詳細については、「#unique_1461_Connect_42_section_qnd_snl_zbc」を参照してください。
次の場合に許可 (Allow-If)	条件を満たしている場合にリソースへのアクセスを許可します。

「オブジェクト」は、アクセスを制御する必要があるターゲットです。各オブジェクトは、特定のテーブル、フィールド、またはレコードを一意に識別するタイプと名前で構成されます。[適用先] フィールドを使用すると、ユーザーはこの ACL が適用される特定のレコードをきめ細かく制御できます。

たとえば、次のエントリはすべてオブジェクトを指定します。

タイプ	名前	保護されたオブジェクト
レコード	[incident].[--None--]	インシデントテーブル。
record	[incident].[active]	インシデントテーブルのアクティブフィールド。
レコード	[incident] 適用先：優先度 = P1	インシデントテーブル内の優先度 1 のインシデントのみ。
REST_Endpoint	user_role_inheritance	user_role_inheritance Scripted REST API のレコード。

各「操作」は、指定されたオブジェクトに対してシステムが実行できる有効な「アクション」を説明しています。レコードなどの一部のオブジェクトは複数の操作をサポートしますが、REST_Endpoint などの他のオブジェクトは 1 つの操作のみをサポートします。

たとえば、これらすべてのエントリは操作を指定します。

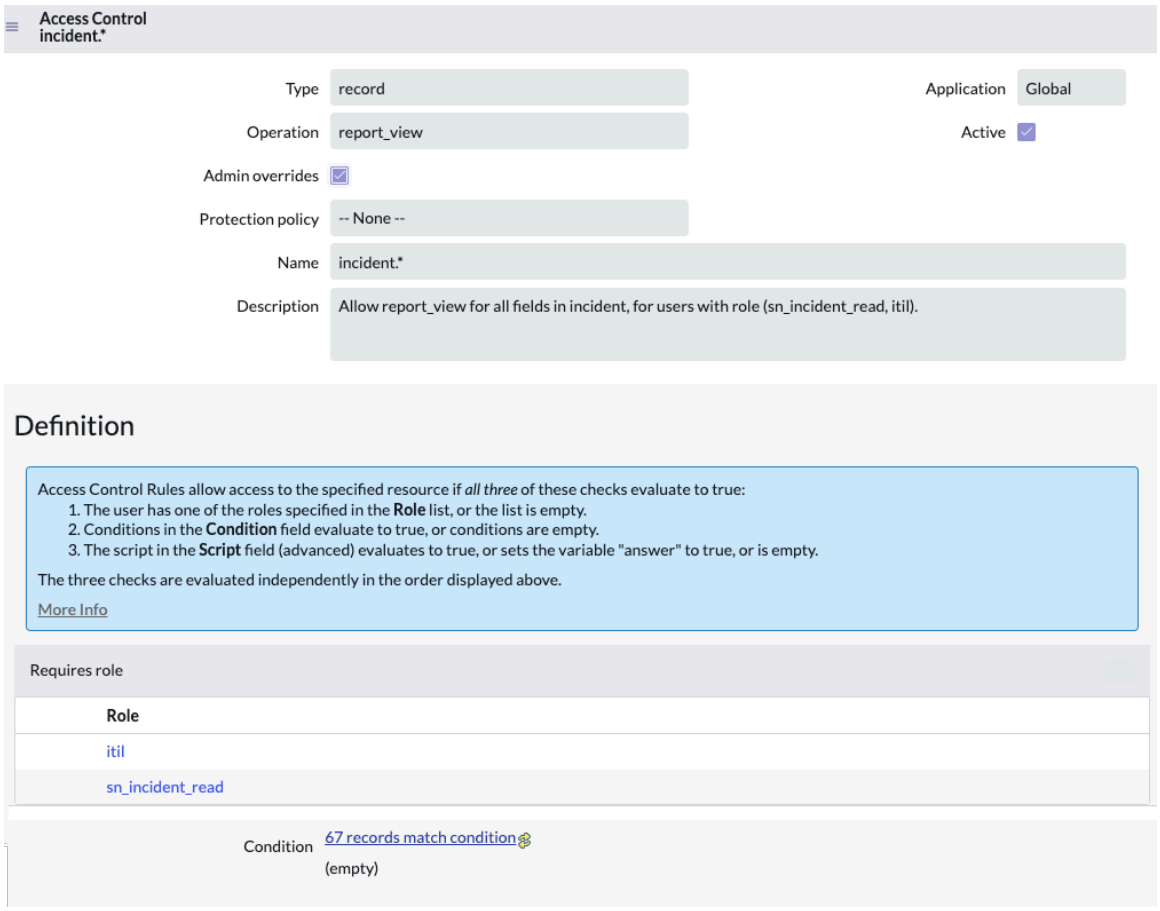
タイプ	名前	運用	保護されている操作
レコード	[incident].[-- None --]	create	インシデントテーブルでのレコードの作成。

タイプ	名前	運用	保護されている操作
レコード	[incident].[active]	write	インシデントテーブルでのアクティブフィールドの更新。
REST_Endpoint	user_role_inheritance	execute	user_role_inheritance Scripted REST API の実行。

「条件」は、指定のオブジェクトと操作に対して、ユーザーがどういった場合にアクセスできるかを指定します。セキュリティアドミニストレーターは、次のものを追加することで条件要件を指定できます。

- [ロールが必要] リストへの 1 つ以上のユーザーロール。
- 1 つ以上のセキュリティ属性が true と評価される必要があります。
- 1 つ以上のデータ条件。
- true または false に評価するスクリプト、または `answer` 変数を true または false に設定するスクリプト。

オブジェクトと操作にアクセスするには、ユーザーはアクセス制御にリストされているすべての条件を渡す必要があります。たとえば、このアクセス制御は、インシデントテーブルの表示操作へのアクセスを制限します。



インシデントテーブルのレコードを更新するには、ユーザーがリストされているロールを持ち、レコードが条件を満たす必要があります。

条件タイプ	要件	説明
必要なロール	必要なロール：itil	itil ロールを持つユーザーのみがインシデントを更新できます。
セキュリティ属性	UserIsAuthenticated	認証されたユーザーのみがインシデントを更新できます。
データ条件	[インシデント状況] [ではない] [クローズ済み]	アクティブなインシデントレコードの更新のみを許可します。

適用先の動作

[適用先] フィールドは ACL がレコードに適用されるかどうかを決定しますが、データ条件は既に適用されている ACL を評価します。適用先 は、ACL が特定のレコードに影響を与えるかどうかを指定します。空の場合、ACL はすべてのレコードに適用されます。適用先 はきめ細かな ACL の適用に使用できますが、「データ条件」は評価基準です。

注: *Applies-to* では大文字と小文字が区別されます。

デフォルトで却下する動作

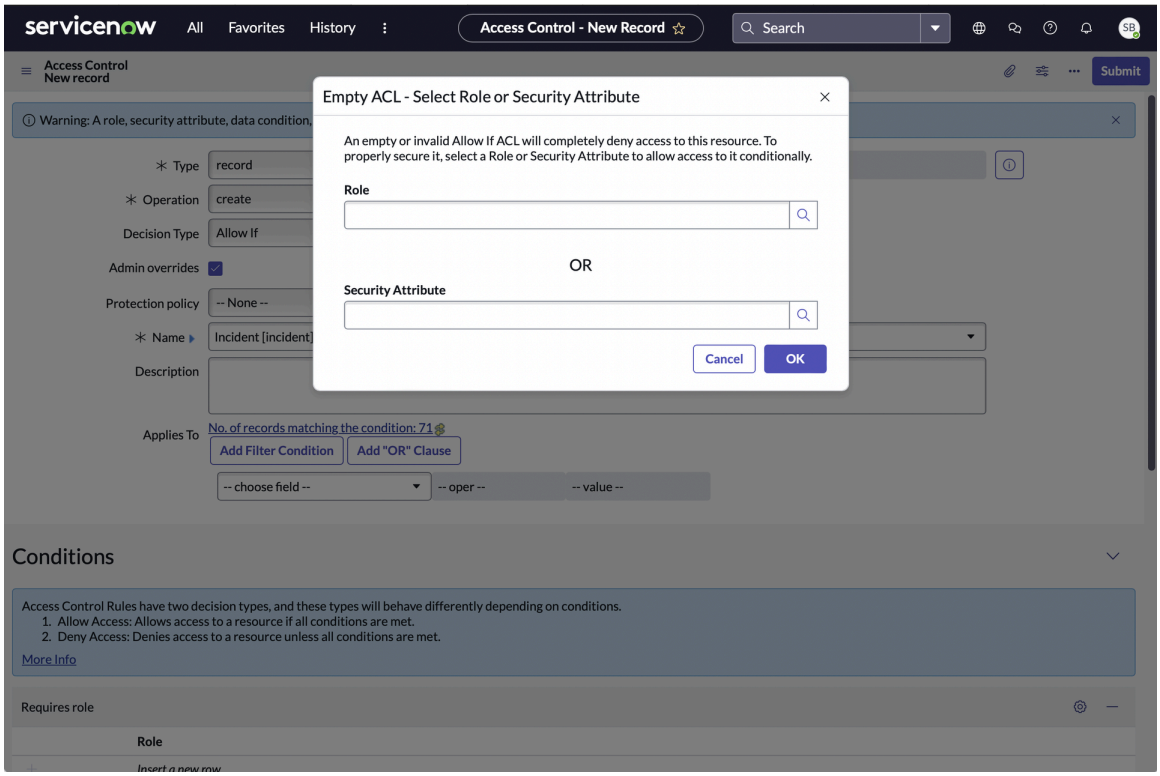
デフォルトでは、ACL が空または無効な場合、ACL エンジンはアクセスを完全に拒否します。空の ACL とは、次のコンポーネントを 1 つ以上持たない ACL として定義されます。

- 定義されたロール
- セキュリティ属性
- データ条件
- スクリプト

無効な ACL は次のように定義されます。

- 存在しないロールを持つ ACL (例：データベースに行がない)
- (データベースに行がないなど) 存在しないセキュリティ属性を持つ ACL
- 「answer=true」または「true」が含まれるスクリプトを持つ ACL

システムは、ACL を作成しているユーザーを検出すると、そのユーザーにロールまたは既存のセキュリティ属性を選択するように求めます。

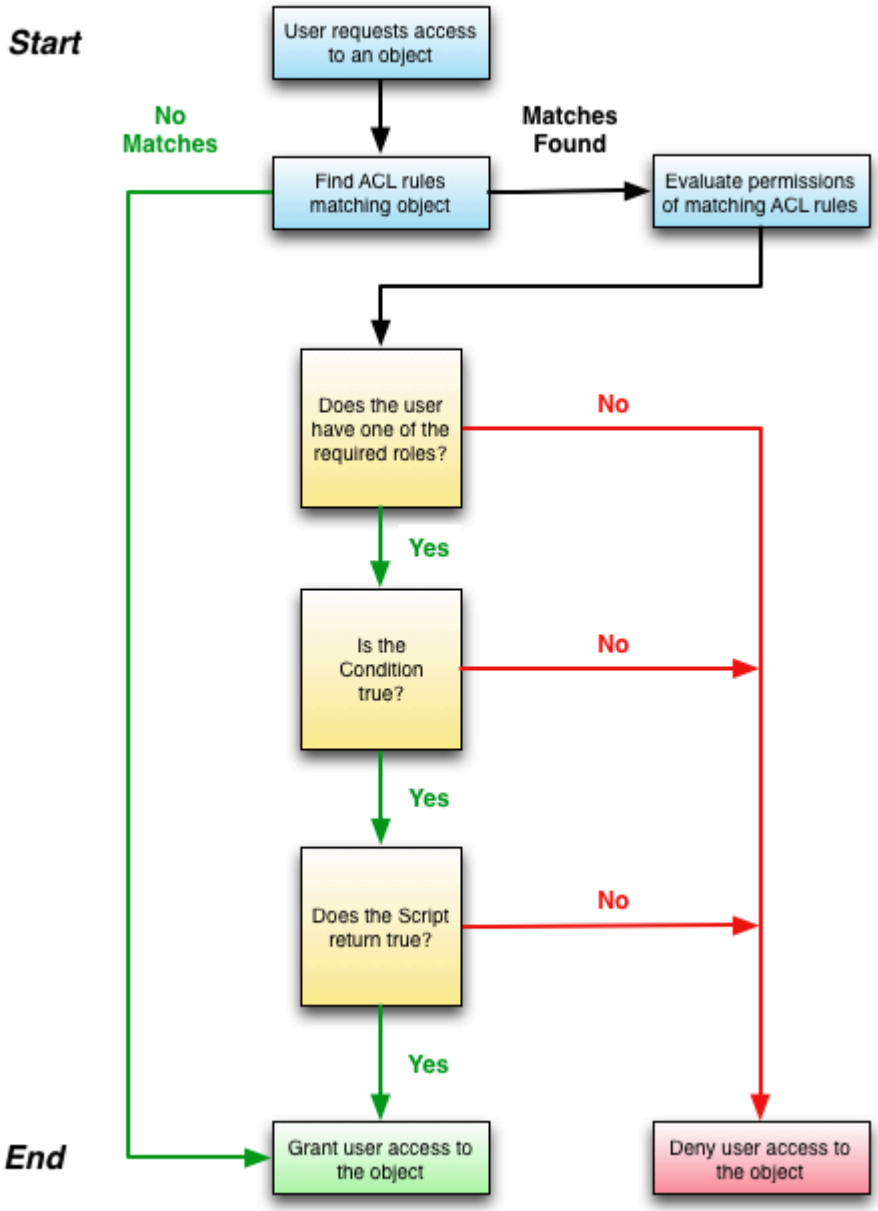


ACL 評価プロセス

ACL ルールは、ユーザーが一致する ACL ルールに必要なすべての条件を満たしている場合にのみ、オブジェクトへのアクセスを許可します。

- 条件は *true* と評価される必要があります。
- スクリプトは *true* と評価されるか、*true* の値を持つ answer 変数を返す必要があります。
- ユーザーは、必要なロールリストのいずれかのロールを持っている必要があります。リストが空の場合、この条件は *true* と評価されます。
- [レコード ACL ルールのみ] 一致するテーブルレベルの ACL ルールとフィールドレベルの ACL ルールの両方が *true* と評価される必要があります。

ACL 評価条件



セッションでデータが要求されるたびに、要求されたオブジェクトと操作に一致するアクセス制御ルールが検索されます。一致するアクセス制御ルールがある場合、ユーザーがオブジェクトと操作にアクセスするために必要な条件を持っているかどうかが評価されます。アクセス制御ルールで複数の条件が指定されている場合、ユーザーがオブジェクトと操作へのアクセス権を得るには、すべての条件に適合する必要があります。いずれかの条件チェックに失敗すると、ユーザーは一致するオブジェクトと操作にアクセスできません。

ユーザーが最初に一致したルールの条件を満たさない場合、アクセス制御処理順序で指定されたとおりに次に一致するアクセス制御ルールの条件が評価されます。一致したどのアクセス制御ルールの条件もユーザーが満たせない場合、システムは要求されたオブジェクトと操作へのアクセスを拒否します。

注: 要求されたオブジェクトと操作に一致するアクセス制御ルールがない場合、ユーザーにアクセス権が付与されます。実際には、システムにはすべてのレコード操作を保護する一連のデフォルトのアクセス制御ルールがあるため、一致するルールが見つからないことはほとんどありません。

オブジェクトへのアクセスが拒否された場合の影響は、ユーザーが満たしていない ACL ルールによって異なります。たとえば、読み取り操作の ACL ルールを満たしていない場合、ユーザーはオブジェクトを表示できません。保護されているオブジェクトに応じて、ACL ルールによってフォーム上のフィールドが非表示にされたり、リストの行が非表示にされたり、ユーザーが UI ページへのアクセスを禁止されたりします。次の表には、特定の操作とオブジェクトタイプの ACL ルールを満たしていない場合の結果の完全なリストが示されています。

クエリー前後の **ACL** チェック

インスタンスは、ユーザーがクエリーを実行する前と後に ACL ルールをチェックします。クエリーの前と後で使用可能な情報が異なるため、結果が異なる可能性があります。

クエリー前の ACL チェック

インスタンスはデータベースクエリを実行する前に、クエリ対象のテーブル内の各フィールドの ACL ルールをチェックして、ユーザーがアクセスできるフィールドを判断します。このチェックでは、ユーザーのロールのみを調べ、これらのロールがフィールドへのアクセスを許可されるかどうかを確認します。このチェックはクエリの前に実行されるため、ACL はテーブルのレコードにアクセスできず、そのデータを考慮することができません。レコードの内容を知ることに依存するスクリプトと条件は評価されません。

この時点でユーザーに読み取りアクセス権がない場合、フィールドの値はユーザーに表示されません。

クエリー後の ACL チェック

クエリーの後に、インスタンスはクエリーによって返された各レコードをチェックします。このチェック中には、ACL のコンテキストがあるため、ACL のロール、条件、およびスクリプトの部分が評価されます。この時点でユーザーが読み取りアクセス権を持っていない場合、フィールドの値はユーザーに表示されませんが、ユーザーのロールがフィールドへのアクセスを許可されているものである場合は、フィールドラベルが表示されます。

運用	オブジェクトに関する **ACL** ルールを満たしていない場合の結果
execute	ユーザーは、レコードまたは UI ページでスクリプトを実行できません。
作成	ユーザーはフォームから新しい UI アクションを表示できません。ユーザーは、Web サービスなどの API プロトコルを使用してテーブルにレコードを挿入することもできません。 フィールドに特定の値が含まれることを要求する条件を持つ 作成 ACL は、false と評価される場合があります。新しいレコードのフィールドは、レコードが保存されるまで空と見なされます。
read	ユーザーはフォームまたはリスト内のオブジェクトを表示できません。ユーザーは、Web サービスなどの API プロトコルを使用してレコードを取得することもできません。
write	ユーザーにはフォームとリストの読み取り専用フィールドが表示され、Web サービスなどの API プロトコルを使用してレコードを更新することはできません。
delete	ユーザーはフォームから削除 UI アクションを表示できません。ユーザーは、Web サービスなどの API プロトコルを使用してテーブルからレコードを削除することもできません。
edit_task_relations	ユーザーはタスクテーブル間の関係を定義できません。

運用	オブジェクトに関する **ACL** ルールを満たしていない場合の結果
edit_ci_relations	ユーザーは構成アイテム [cmdb_ci] テーブル間の関係を定義できません。
save_as_template	テンプレートの作成時に保存する必要があるフィールドを制御するために使用されます。
add_to_list	ユーザーはリストメカニックの特定の列を表示またはカスタマイズすることができません。
list_edit	ユーザーはリストからレコード (行) を更新できません。
report_on	ユーザーは ACL テーブルでレポートを作成できません。詳細については、「ACL ルールを使用してレポートの作成を制限する」を参照してください。
report_view	ユーザーは、ACL テーブルまたは ACL フィールドのレポートの内容を表示できません。詳細については、「レポート」を参照してください。
personalize_choices	ユーザーは [リスト] フィールドを右クリックして [設定の選択] を選択することはできません。

オブジェクトの **ACL** 一致要件

オブジェクトタイプ	アクセスオブジェクトに必要な一致する **ACL** ルール	既存のワイルドカード **ACL** ルール
クライアント呼び出し可能スクリプトインクルード プロセッサー	ユーザーは次の 2 つの ACL ルールの条件を満たす必要があります。 **1.** オブジェクトのすべてのワイルドカード ACL ルール (操作に対する ACL ルールが存在する場合)。 **2.** オブジェクトの名前に一致する最初の ACL ルール (操作に対する ACL ルールが存在する場合)。	デフォルトでは、これらのオブジェクトタイプにワイルドカード (*) ルールはありません。これらのオブジェクトの 1 つにワイルドカード ACL ルールを作成すると、ACL ルールがこのタイプのすべてのオブジェクトに適用されます。
UI ページ レコード	ユーザーは次の 2 つの ACL ルールの条件を満たす必要があります。 **1.** レコードのフィールドに一致する最初の ACL ルール (操作に対する ACL ルールが存在する場合)。 **2.** レコードのテーブルに一致する最初の ACL ルール (操作に対する ACL ルールが存在する場合)。	デフォルトでは、作成、読み取り、書き込み、および削除操作用のワイルドカードテーブルルール (*) と、personalize_choices、create、および save_as_template 操作用のワイルドカードフィールドルール (*.*) があります。テーブルを作成するときは、提供されているワイルドカード ACL ルールを使用する場合を除き、テーブルの ACL ルールを作成します。

注: セキュリティマネージャーのデフォルトの動作 (*glide.sm.default_mode*) プロパティは、ユーザーがワイルドカードテーブルの ACL ルールにのみ一致するオブジェクトにアクセスできるかどうかを決定します。このプロパティが [アクセスを拒否] に設定されている場合、アドミンのみがワイルドカードテーブルの ACL ルールにアクセスできます。

自動翻訳

注: 作成操作のワイルドカードフィールド ACL ルール (*.*) は、書き込み操作と同じ条件を再利用します。これは、明示的な作成操作の ACL ルールが定義されない限り、作成条件は書き込み条件と同じになることを意味します。

処理順序の同じ時点にある複数の **ACL** ルール

2 つ以上のルールが処理順序の同じ時点で一致する場合、ユーザーは、オブジェクトにアクセスするためにいずれかの ACL ルールの条件を渡す必要があります。たとえば、*incident.number* に対して 2 つのフィールド ACL ルールを作成した場合、1 つのルールを満たすユーザーは、ユーザーが処理順序の同じ時点で他のフィールド ACL ルールを満たしていないかどうかにかかわらず、その番号フィールドにアクセスできます。

必要なロール

通常のアドミンユーザーは、アクセス制御ルールを表示してデバッグできます。ただし、既存のアクセス制御ルールを作成または更新するには、admin が security_admin ロールに権限を昇格させる必要があります。手順については、「特権ロールへの昇格」を参照してください。

スコープ対象のアプリケーションの **ACL** ルール

ACL ルールと同じスコープ内にあるオブジェクトに対して ACL ルールを作成できます。また、ACL ルールと同じスコープ内にある少なくとも 1 つのフィールドを持つテーブルに対して、ACL ルールを作成できます。

ACL ルールレコードとは異なるスコープ内のテーブルの場合は、ルールのタイプが制限されます。

- ACL ルールと同じスコープ内にある任意のテーブル、UI ページ、またはその他のオブジェクトに対して ACL ルールを作成できます。
- ACL ルールと同じスコープ内にあるフィールドに対して ACL を作成できます。
  - テーブルが同じスコープ内にある場合は、スクリプトを使用して条件を評価できます。
  - テーブルが別のスコープ内にある場合は、スクリプトを使用して条件を評価できません。
- アプリケーションピッカーで選択したアプリケーションとは異なるスコープにあるオブジェクトの ACL ルールを作成または変更することはできません。これには、異なるスコープの ACL へのロールの追加も含まれます。
- ワイルドカードテーブルルール (*) は、グローバルスコープでのみ作成できます。
- ACL ルールと同じスコープ内のテーブルに対してのみ、ワイルドカードフィールドルール (*) を作成できます。

ACL ルールのタイプ

システムのさまざまなコンポーネントに関する ACL ルールを作成します。

レコードの **ACL** ルール

レコードの ACL ルールはテーブル名とフィールド名で構成されます。

- テーブル名は保護するテーブルです。他のテーブルがこのテーブルから拡張されている場合、そのテーブルは親テーブルと見なされます。親テーブルの ACL ルールは、親テーブルを拡張するすべてのテーブルに適用されます。
- フィールド名は保護するフィールドです。テーブル拡張により、一部のフィールドが複数のテーブルに含まれています。親テーブル内のフィールドの ACL ルールは、親テーブルを拡張するすべてのテーブルに適用されます。

ACL ルールは、次のレコード操作を保護できます。

操作	説明
実行	ユーザーがクライアント呼び出し可能スクリプトインクルードと REST エンドポイントを実行できるようにします。
query_match	ユーザーが一致クエリ (「次の値に等しい (=)」、「次の値と異なる (!=)」、「は (空) である」など) を送信できるようにします。
conditional_table_query_range	ユーザーが読み取り ACL に基づいて部分的な ACL アクセス権を付与できるようにします。 データ条件とスクリプトのない読み取り ACL を持つテーブルに対して作成されます。
query_range	ユーザーが範囲クエリ (「次で始まる」、「次の値で終わる」、「は次の値を含む」など) を送信できるようにします。ソートに制限はありません。
create	ユーザーが新しいレコード (行) をテーブルに挿入できるようにします。
read	ユーザーがテーブルのレコードを表示できるようにします。
write	ユーザーがテーブル内のレコードを更新できるようにします。
delete	ユーザーがテーブルからレコードを削除したり、テーブルを削除したりできるようにします。
edit_task_relations	ユーザーがタスク [task] テーブルを拡張できるようにします。
edit_ci_relations	ユーザーが構成アイテム [cmdb_ci] テーブルを拡張できるようにします。
save_as_template	ユーザーがレコードをテンプレートとして保存できるようにします。
add_to_list	ユーザーがリストメカニズムの特定の列を表示またはカスタマイズできないようにします。 **注:** 条件とスクリプトはサポートされていません。
list_edit	ユーザーがリストからレコード (行) を更新できるようにします。
report_on	ユーザーがテーブルでレポートできるようにします。
report_view	ユーザーがフィールド ACL でレポートできるようにします。
personalize_choices	ユーザーがテーブルまたはフィールドを設定できるようにします。
data_fabric	データファブリックテーブルがローカルテーブルを参照できるようにします。

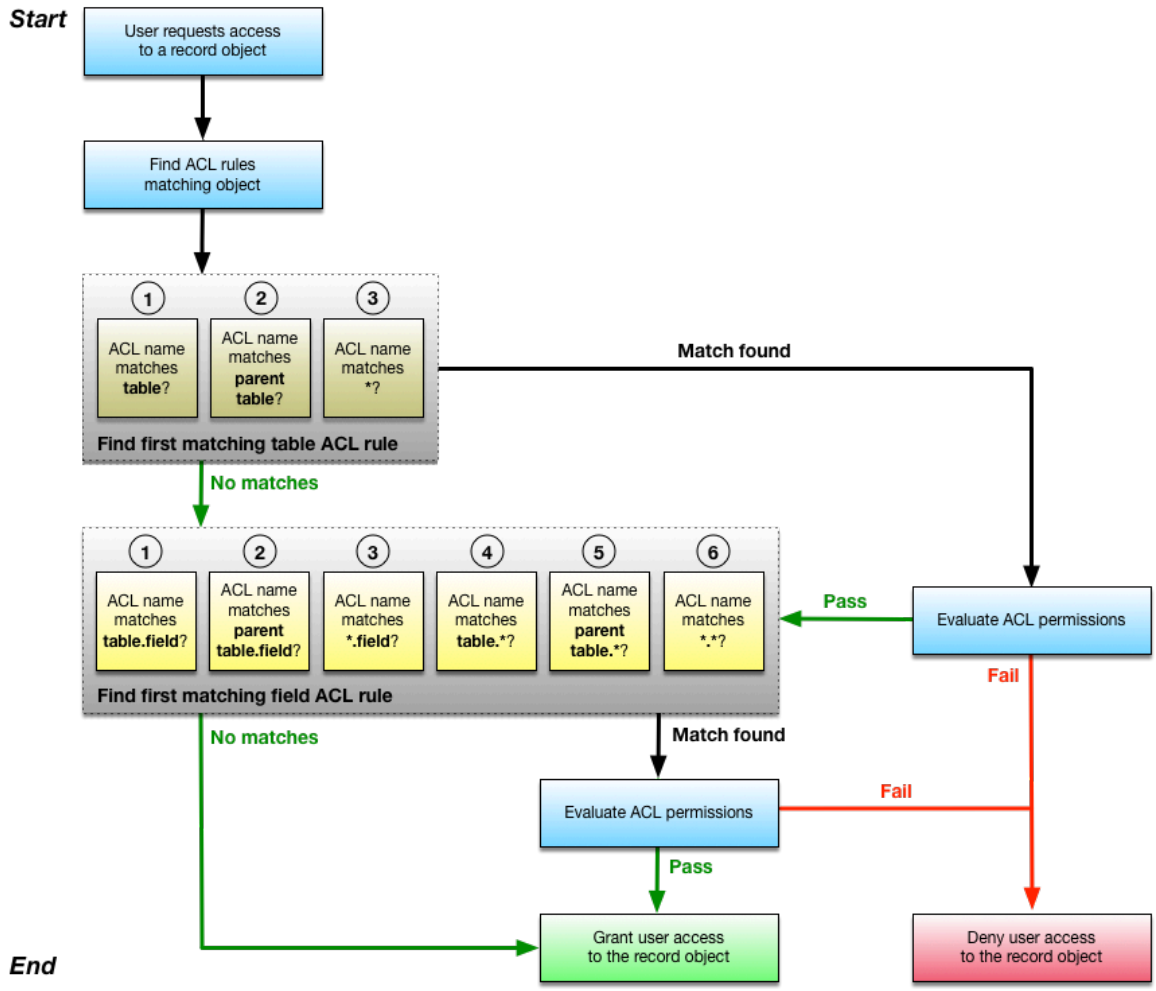
レコード ACL ルールは、次の順序で処理されます。

- オブジェクトをテーブルの ACL ルールと照合します。
- オブジェクトをフィールドの ACL ルールと照合します。

この処理順序により、ユーザーは、より一般的なオブジェクトへのアクセス権を得る前に、より具体的なオブジェクトへのアクセス権を得ることができます。レコードオブジェクトにアクセスするには、テーブルとフィールドの両方の ACL ルールを満たす必要があります。

- ユーザーがテーブルの ACL ルールを満たしていない場合、ユーザーがフィールドの ACL ルールを満たしていても、テーブル内のすべてのフィールドへのアクセスが拒否されます。
- ユーザーがテーブルの ACL ルールを満たしていても、フィールドの ACL ルールを満たしていない場合、ユーザーはフィールドの ACL ルールによって記述されたフィールドにアクセスできません。

ACL の一致



プロセッサの ACL ルール

プロセッサの ACL ルールは、保護するプロセッサを指定します。使用可能なプロセッサのリストについては、次に移動します: システム定義 > プロセッサ.

デフォルトでは、メールクライアントを itil ロールを持つユーザーに制限するために、EmailClientProcessor の ACL ルールが含まれています。

プロセッサの ACL ルールは、それらのリソースに対してより具体的な ACL が見つからない場合、星 (*) ルールに従います。

テーブル ACL ルール

ユーザーは最初にテーブルの ACL ルールに合格する必要があります。ベースシステムにはすべてのテーブルに一致する星 (*) テーブルの ACL ルールが含まれているため、ユーザーは常に少なくとも

1 つのテーブル ACL ルールを満たす必要があります。ベースシステムは、特定のテーブルへのアクセスを制御する追加のテーブル ACL ルールを提供します。

テーブルの ACL ルールは次の順序で処理されます。

1. テーブル名と一致します。たとえば、incident などです。
2. 親テーブル名と一致します。たとえば、task です。
3. いずれかのテーブル名 (*) と一致します。たとえば、* です。

ユーザーがすべてのテーブルの ACL ルールを満たしていないと、ユーザーはテーブル内のフィールドにアクセスできません。ユーザーがテーブルの ACL ルールを満たすと、フィールドの ACL ルールが評価されます。

フィールドの **ACL** ルール

ユーザーがテーブルの ACL ルールに合格すると、フィールドの ACL ルールが次の順序で処理されます。

1. テーブルおよびフィールド名と一致します。たとえば、incident.number です。
2. 親テーブルおよびフィールド名と一致します。たとえば、task.number です。
3. いずれかのテーブル (*) およびフィールド名と一致します。たとえば、*.number です。
4. テーブルおよびいずれかのフィールド (*) と一致します。たとえば、incident.* です。
5. 親テーブルおよびいずれかのフィールド (*) と一致します。たとえば、task.* です。
6. いずれかのテーブル (*) およびフィールド (*) と一致します。たとえば、*.* です。

テーブルのフィールドへのアクセスを許可するには、ユーザーがテーブル ACL ルールを満たす必要があります。たとえば、インシデントテーブルの [番号] フィールドにアクセスするには、まずインシデントテーブルのテーブル ACL ルールを満たす必要があります。

最初に成功したフィールド ACL 評価は、フィールドレベルで ACL ルール処理を停止します。ユーザーがフィールド ACL ルールを満たすと、他の一致するフィールド ACL ルールの検索を停止します。たとえば、ユーザーが incident.number のフィールド ACL ルールを満たしている場合、システムはインシデントテーブルの [番号] フィールドを保護する他の ACL ルールの検索を停止します。

保護されたフィールドでは推定データのクエリ情報へのアクセスが制限されているため、予測情報が返されません。

UI ページの **ACL** ルール

UI ページの ACL ルールは、保護する UI ページを指定します。利用可能な UI ページのリストについては、次に移動します: システム **UI** > **UI** ページ. UI ページの ACL ルールを定義するときは、完全にスコープ範囲内のページ名を使用します。たとえば、**x_myapp_mypage** です。

> 注: **ui_page** タイプの ACL の [名前] フィールドで星印 (*) 文字を使用して、任意の UI ページに一致させることができます。

UI ページの ACL ルールは、それらのリソースに対してより具体的な ACL が見つからない場合、星 (*) ルールに従います。たとえば、mysecretpage という名前の UI ページがあり、この UI ページの ACL が定義されていない場合は、UI ページプロセッサの星 (*) ルールがアクセスチェックに使用されます。

ACL ルールは、次の UI ページ操作を保護できます。

操作	説明
read	ユーザーが UI ページを表示できるようにします。

クライアント呼び出し可能スクリプトインクルードの **ACL** ルール

スクリプトインクルードの ACL ルールは、保護するクライアント呼び出し可能スクリプトインクルードを指定します。使用可能なスクリプトインクルードのリストについては、次に移動します: システム定義 > スクリプトインクルード. リストをカスタマイズして [クライアント呼び出し可能] 列を表示できます。

ベースシステムには、クライアント呼び出し可能スクリプトインクルードの ACL ルールが含まれていません。

クライアント呼び出し可能スクリプトインクルードの ACL ルールは、それらのリソースに対してより具体的な ACL が見つからない場合、星 (*) ルールに従います。

データタイプ **ACL**

データタイプ ACL を使用すると、特定のタイプのすべてのフィールドに適用される ACL ルールを記述できます。

データタイプ ACL は、データタイプに基づいてテーブルフィールドを制限することで、アクセス制御に的を絞ったアプローチを提供します。これにより、ワイルドカード (*) ACL よりも広範なセキュリティ制約が可能になります。データタイプ ACL の構文は、形式 * に従います。[(制限するフィールド)]。

従来のフィールド ACL は、特定のテーブルフィールド識別子、テーブル内のすべてのフィールド、または特定のフィールド名を持つすべてのテーブルに制限されていますが、データタイプ ACL を使用すると、特定のメタデータを共有するフィールドに一様にセキュリティを適用できます。これにより、すべてのフィールドにセキュリティを均一に適用するための追加の ACL を作成する必要がなくなります。

データタイプ ACL を実装する場合、予期しないセキュリティ上の問題を回避するために、フィールドを追加する前と後に影響を受けるすべてのフィールドを検証することが不可欠です。独自のデータ型 ACL を作成する データタイプ ACL の作成 を参照してください。

既存のデータタイプ ACL を確認するには、次の場所に移動します。 すべて > システムセキュリティ > アクセス制御 をクリックし、[名前] フィールドを使用して、 *.[で始まる ACL を検索します。

> **注:** Scripting Governance は、デフォルトでスクリプト制限のためにデータタイプ ACL を使用します。詳細は _ を参照してください。

データタイプ **ACL** の作成

データタイプ ACL の作成方法について説明します。

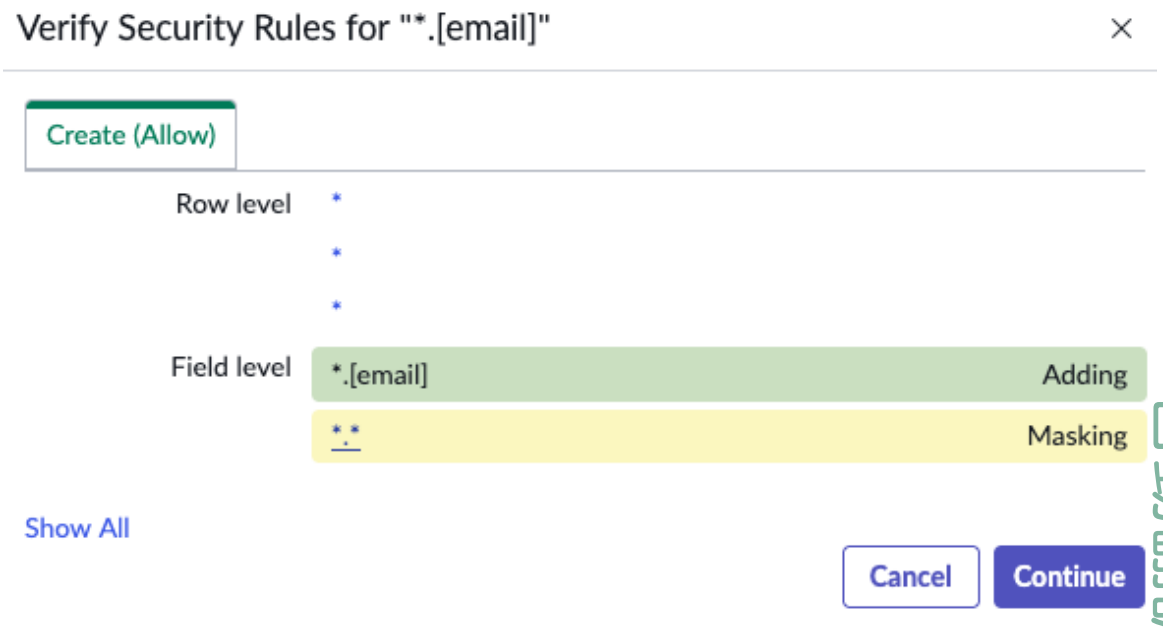
始める前に
必要なロール：admin

手順

1. 移動先 すべて > システムセキュリティ > アクセス制御.
2. **[New** (新規)] ボタンをクリックします。
3. [名前] フィールドの近くにある [モードの変更] ▸を選択します

4. dataype を * の形式で入力します 。ユーザーデータタイプ] を [名前] フィールドに入力します。
5. ACL に関連付ける ロール または セキュリティ属性 を選択します。
6. オプション: **[ACL** を作成] フォームにその他の詳細を入力します。
   詳細については、「ACL ルールの構成」を参照してください。
7. **[Submit** (送信)] を選択します。
8. ポップアップの [セキュリティルールを確認 **(Verify Security Rules**)] ウィンドウで、ACL のセキュリティルールを確認します。

セキュリティルールの検証の例



機能フィールドの **ACL** 制御

機能フィールドへのアクセスを評価する場合、機能フィールド自体へのアクセスがチェックされるだけでなく、機能の貢献フィールドへのアクセスもチェックされます。貢献フィールドは、指定された機能の定義の引数として使用されるフィールドです。

機能フィールドの詳細については、「機能フィールド」を参照してください。

Rome 以前では、(他のフィールドと同様に) 機能フィールド自体へのアクセスをチェックします。そのフィールドの ACL でアクセスが許可されている場合、ユーザーが貢献フィールドへのアクセス権を持っているかどうかにかかわらず、ユーザーは結果の値を受け取ります。

Zurich 以降では、機能フィールドへのアクセスを許可するには、すべての貢献フィールドへのアクセスも必要です。1 つ以上の貢献フィールド ACL でアクセスが拒否される場合、機能フィールドでもアクセスが拒否されます。

新しい要件の影響を受ける操作は、読み取りと report_view のみです。report_view には独自の追加要件があります。

操作	説明
読み取り操作	ユーザーは、次の両方の条件に当てはまる場合にのみ、機能フィールドの読み取りアクセス権があります。

操作	説明
	• ユーザーは機能フィールドの読み取りアクセス権を持っている。 • ユーザーは、機能で使用されるすべての貢献フィールドの読み取りアクセス権を持っている。
report_view 操作	ユーザーは、次のすべての条件に当てはまる場合にのみ、機能フィールドの report_view アクセス権があります。 • ユーザーは機能フィールドの report_view アクセス権を持っている。 • ユーザーは、各貢献フィールドの report_view アクセス権を持っている。 • 条件とスクリプトのないロールのみの読み取り ACL があり、ユーザーはそのロールを持っている。 • ユーザーは、条件またはスクリプトのない ACL のみが許可されるように、貢献フィールドの読み取り専用アクセス権を持っている。

例

前提：

- テーブル：給与
- 列：基本、ボーナス、合計 (この例ではすべて整数)
- 機能フィールド：合計列は機能フィールドとしてマークされ、機能の定義は glidefunction:add（基本、ボーナス）です。
- 貢献フィールド：機能の定義で使用されるため、基本とボーナス
- ロール：salary_admin、bonus_admin

例 1：すべてのフィールドでアクセスが可能

ACL	結果
合計、基本、ボーナス：salary_admin ロールの場合、読み取りおよび report_view (条件またはスクリプトなし)	必要なロールを持っているため、salary_admin ロールを持つユーザーには、合計の読み取りと report_view アクセス権が付与されます。

例 2：貢献フィールドで読み取りアクセスが拒否される

ACL	結果
• 合計、基本：salary_admin ロールの場合、読み取りおよび report_view (条件またはスクリプトなし) • ボーナス：salary_admin ロールの場合、report_view (条件またはスクリプトなし) • ボーナス：bonus_admin ロールの場合、読み取り (条件またはスクリプトなし)	salary_admin ロールではボーナスの読み取りアクセスが拒否されるため、salary_admin ロールを持つユーザーは、合計の読み取りと report_view アクセスは拒否されます。

例 3：貢献フィールド **ACL** にスクリプトがある

ACL	結果
• 合計、基本：salary_admin ロールの場合、読み取りおよび report_view (条件またはスクリプトなし) • ボーナス：bonus_admin ロールの場合、report_view (条件またはスクリプトなし) • ボーナス：salary_admin ロールの場合、読み取り、スクリプトあり (内容にかかわらず、スクリプトがあることのみが重要)	すべてのフィールドに対する必要なロールを持っているため、salary_admin ロールを持つユーザーには、合計の読み取りアクセス権が付与されます。 ただし、必要なロールを持っている場合でも、スクリプトのある ACL はデフォルトで読み取りアクセスを拒否するため、salary_admin ロールを持つ同じユーザーは report_view アクセスを拒否されます。

Security Jump Start - ACL ルールプラグイン

Security Jump Start アクセスコントロールレベル (ACL ルール) プラグインは、すべての新しいインスタンスに自動的にインストールされます。このプラグインを使用すると、複数のシステムテーブルを迅速に保護し、組織の本番起動プロセスを迅速化できます。

このプラグインは、本番環境で既に使用されているテーブルへのセキュリティアクセスを変更する可能性があるため、既存のインスタンスを対象としていません。アドミンがこのプラグインを既存のインスタンスにインストールすることを強く希望する場合は、最初にテストインスタンスで徹底的にテストを行う必要があります。これを行うことで、組織の現在の実装との互換性を検証できます。

このプラグインから提供される新しい ACL ルールにアドミンが注目している場合は、ACL のリストを指針として使用し、既存のインスタンスに 1 つ以上の ACL ルールを手動で作成できます。

このプラグインには次の ACL が含まれています。ヘッダー行のアイコンを選択して、その列を昇順または降順でソートします。操作キーは次のとおりです。

- R = 読み込み
- W = 書き込み
- D = 削除
- C = 作成

名前	運用	説明
cmdb_ci	WCD	構成アイテムレコードの書き込み/作成/削除には、asset または itil ロールが必要です
cmn_department	WD	部門レコードの書き込み/削除には user_admin ロールが必要です
cmn_location	WC	場所レコードの書き込み/作成には user_admin ロールが必要です
core_company	WD	会社レコードの書き込み/削除には user_admin ロールが必要です
kb_knowledge	作成	ナレッジレコードを作成するには「knowledge」ロールが必要です

名前	運用	説明
ldap_ou_config	RWCD	LDAP OU 定義レコードの読み取り/書き込み/作成/削除には、user_admin ロールが必要です
ldap_server_config	RWCD	LDAP サーバーレコードの読み取り/書き込み/作成/削除には、user_admin ロールが必要です
process_guide	WCD	プロセスガイドレコードの書き込み/作成/削除には、admin ロールが必要です
process_step	WCD	プロセスステップレコードの書き込み/作成/削除には、admin ロールが必要です
sc_category	作成	サービスカタログカテゴリレコードの作成には、catalog_admin ロールが必要です
sc_category	削除	サービスカタログカテゴリレコードの削除には、catalog_admin ロールが必要です
sc_category	書き込み	サービスカタログカテゴリレコードへの書き込みには、catalog_admin ロールが必要です
sc_cat_item	書き込み	カタログアイテムレコードに書き込むには「catalog_admin」ロールが必要です
sc_cat_item	削除	カタログアイテムレコードを削除するには「catalog_admin」ロールが必要です
sc_cat_item	作成	カタログアイテムレコードを作成するには、「catalog_admin」ロールが必要です
sysevent_email_action	読み込み専用	メール通知レコードは、すべてのユーザーが読み込めます (サブスクリプション目的)
sysevent_register	RWCD	イベント登録レコードの読み取り/書き込み/作成/削除には、admin ロールが必要です
sysevent_script_action	RWCD	スクリプトアクションレコードの読み取り/書き込み/作成/削除には、admin ロールが必要です
syslog	RWCD	ログエントリレコードの読み取り/書き込み/作成/削除には、admin ロールが必要です
sysrule	RWCD	ルールレコード (メール通知、受信メールアクション、承認ルールなど) の読み取り/書き込み/作成/削除には、admin ロールが必要です
sysrule	read	(サブスクリプションベースの通知) のメール通知レコードは、すべてのユーザーが読み込めます
sys_app_application	WCD	アプリケーションレコードの書き込み/作成/削除には、admin ロールが必要です
sys_app_category	WCD	アプリケーションカテゴリレコードの書き込み/作成/削除には、admin ロールが必要です
sys_app_module	WCD	モジュールレコードの書き込み/作成/削除には、admin ロールが必要です
sys_audit	RWCD	監査レコードの読み取り/書き込み/作成/削除には、admin ロールが必要です
sys_dictionary	RWC	辞書レコードの読み取り/書き込み/作成には、personalize_dictionary ロールが必要です

名前	運用	説明
sys_dictionary.*	read	「personalize_dictionary」ロールでは、辞書フィールドを読み込むことができます
sys_documentation	削除	フィールドラベルレコードを削除するには「personalize_dictionary」ロールが必要です
sys_documentation	作成	フィールドラベルレコードを作成するには「personalize_dictionary」ロールが必要です
sys_documentation	書き込み	フィールドラベルレコードへの書き込みにはpersonalize_dictionary ロールが必要です
sys_gauge	RWCD	ゲージレコードの読み取り/書き込み/作成/削除には、admin ロールが必要です
sys_gauge_count	RWCD	ゲージカウントレコードの読み取り/書き込み/作成/削除には、admin ロールが必要です
sys_group_has_role	read	グループロールレコードを表示するには、itil ロールが必要です
sys_home	WCD	ようこそページセクションレコードの書き込み/作成/削除には、itil_admin ロールが必要です
sys_installation_exit	WCD	インストレーションイグジットレコードの書き込み/作成/削除には、admin ロールが必要です
sys_job	WCD	システムジョブレコードの書き込み/作成/削除には、admin ロールが必要です
sys_nav_link	WCD	ナビゲーションリンクレコードの書き込み/作成/削除には、admin ロールが必要です
sys_perspective	WCD	メニューリストレコードの書き込み/作成/削除には、admin ロールが必要です
sys_portal	RWCD	ポータルレコードの読み取り/書き込み/作成/削除には、admin ロールが必要です
sys_portal_page	RWCD	ホームページレコードの読み取り/書き込み/作成/削除には、admin ロールが必要です
sys_portal_preferences	RWCD	ポータル設定レコードの読み取り/書き込み/作成/削除には、admin ロールが必要です
sys_processor	WC	プロセッサーレコードの書き込み/作成には、admin ロールが必要です
sys_properties	WC	システムプロパティレコードの書き込み/作成には、admin ロールが必要です
sys_properties_category	WCD	プロパティカテゴリレコードの書き込み/作成/削除には、admin ロールが必要です
sys_report	delete	レポートレコードを削除できるロール (レポート UI からの削除を制限しない)
sys_report	write	レポートレコードに書き込むことができるロール (レポート UI からの編集を制限しない)
sys_report	read	ユーザーは、各自のレポートレコード、所属グループのレポートレコード、およびグローバル属性のレポートレ

名前	運用	説明
		コードを読み込むことができます (レポート UI を介した参照には影響しません)
sys_report	read	レポートレコードを読み込むことができるロール (レポート UI からの表示を制限しない)
sys_reportroles	read	レポートロールレコードの読み込みには admin ロールが必要です
sys_script	WCD	ビジネスルールレコードの書き込み/作成/削除には、admin ロールが必要です
sys_script_ajax	WCD	AJAX スクリプトレコードの書き込み/作成/削除には、admin ロールが必要です
sys_script_client	WCD	クライアントスクリプトレコードの書き込み/作成/削除には、admin ロールが必要です
sys_script_include	WCD	スクリプトインクルードレコードの書き込み/作成/削除には、admin ロールが必要です
sys_security_acl	write	アクセス制御レコードへの書き込みには admin ロールが必要です
sys_security_acl_role	作成	アクセスロールレコードの作成には admin ロールが必要です
sys_security_acl_role	delete	アクセスロールレコードの削除には admin ロールが必要です
sys_security_acl_role	write	アクセスロールレコードへの書き込みには admin ロールが必要です
sys_security_operation	delete	セキュリティオペレーションレコードの削除には admin ロールが必要です
sys_security_operation	作成	セキュリティオペレーションレコードの作成には admin ロールが必要です
sys_security_operation	write	セキュリティオペレーションレコードへの書き込みには admin ロールが必要です
sys_security_operation	query_range	
sys_security_operation		
sys_security_type	write	セキュリティタイプレコードへの書き込みには admin ロールが必要です
sys_security_type	作成	セキュリティタイプレコードの作成には admin ロールが必要です
sys_security_type	delete	セキュリティタイプレコードの削除には admin ロールが必要です
sys_status	作成	システムステータスレコードの作成には admin ロールが必要です
sys_status	delete	システムステータスレコードの削除には admin ロールが必要です
sys_status	write	システムステータスレコードへの書き込みには admin ロールが必要です

名前	運用	説明
sys_template	write	テンプレートレコードへの書き込みには template_editor ロールが必要です
sys_template	作成	テンプレートレコードの作成には template_editor ロールが必要です
sys_template	delete	テンプレートレコードの削除には template_editor ロールが必要です
sys_template	read	テンプレートロールレコードの読み込みには template_editor ロールが必要です
sys_ui_action	作成	UI アクションレコードの作成には admin ロールが必要です
sys_ui_action	delete	UI アクションレコードを削除するには admin ロールが必要です
sys_ui_action	write	UI アクションレコードに書き込むには admin ロールが必要です
sys_ui_action_view	write	UI ビューアクションレコードへの書き込みには admin ロールが必要です
sys_ui_action_view	作成	UI ビューアクションレコードの作成には admin ロールが必要です
sys_ui_action_view	delete	UI ビューアクションレコードの削除には admin ロールが必要です
sys_ui_policy	作成	UI ポリシーレコードの作成には admin ロールが必要です
sys_ui_policy	delete	UI ポリシーレコードの削除には admin ロールが必要です
sys_ui_policy	write	UI ポリシーレコードへの書き込みには admin ロールが必要です
sys_ui_policy_action	作成	UI ポリシーアクションレコードの作成には admin ロールが必要です
sys_ui_policy_action	delete	UI ポリシーアクションレコードの削除には admin ロールが必要です
sys_ui_policy_action	write	UI ポリシーアクションレコードへの書き込みには admin ロールが必要です
sys_ui_script	write	UI スクリプトレコードへの書き込みには admin ロールが必要です
sys_ui_script	delete	UI スクリプトレコードの削除には admin ロールが必要です
sys_ui_script	作成	UI スクリプトレコードの作成には admin ロールが必要です
sys_user	write	ロールのないユーザーは、自分以外のユーザーレコードを更新できません
sys_user_grmember	delete	グループメンバーレコードを削除するには「user_admin」ロールが必要です

名前	運用	説明
sys_user_grmember	write	グループメンバーレコードに書き込むには「user_admin」ロールが必要です
sys_user_group	作成	「itil」と上記のロールのみがグループレコードを作成できます
sys_user_group	write	「itil」と上記のロールのみがグループレコードに書き込むことができます
sys_user_has_role	read	ユーザーロールレコードを参照するには「itil」ロールが必要です
sys_user_role	作成	ロールレコードの作成には admin ロールが必要です
sys_user_role	delete	ロールレコードの削除には admin ロールが必要です
sys_user_role	write	ロールレコードへの書き込みには admin ロールが必要です
sys_user_role_contains	read	包含ロールレコードを参照するには「itil」ロールが必要です
sys_user_role_contains	write	包含ロールレコードへの書き込みには admin ロールが必要です
sys_user_token	RWCD	ユーザートークンレコードの読み取り/書き込み/作成/削除には、admin ロールが必要です

注: このプラグインの詳細については、「インスタンスセキュリティ強化設定」の「Security Jump Start プラグイン (ACL ルール) を有効化する (Security Center 1.3 で更新)」を参照してください。

ACL ルールの構成

カスタムアクセス制御リスト (ACL) ルールを設定して、新しいオブジェクトへのアクセスを保護するか、デフォルトのセキュリティ動作を変更します。

始める前に
必要なロール：security_admin

このタスクについて
ACL ルールを作成するには、特権を security_admin ロールに昇格させる必要があります。

ACL ルールレコードとは異なるスコープ内のテーブルの場合は、ルールのタイプが制限されます。スコープマスターテーブルでスコープを派生させてスコープ付き ACL を実行するには、*glide.enforce_security_scope.<scope_name>* プロパティを **true** に設定します。すると、該当するテーブルで作成されたスコープ固有の ACL がある場合に、グローバルスコープ内の ACL が一致しなくなります。たとえば、sys_attachment テーブルや sys_question_answer テーブルといったグローバルスコープ内の共有アプリケーションテーブル内のデータを保護する場合などです。

手順

1. security_admin ロールに昇格された特権ロール。
2. 移動先 システムセキュリティ > アクセス制御 **(ACL)**.
3. [**New**] をクリックします。

ヒント: 新しい ACL を作成する場合は、 「次の場合を除き却下 (Deny-Unless)」ACLを確認すると便利です。

4. フォームを完了します。

アクセス制御フィールド

フィールド	説明
タイプ	この ACL ルールが保護するオブジェクトの種類を選択します。オブジェクトのタイプによって、オブジェクトの命名方法と使用可能な操作が決まります。このフィールドは、ACL ルールが作成された後に読み取り専用になります。タイプを変更する場合は、ACL を削除して正しいタイプで新しい ACL を作成する必要があります。
操作	この ACL ルールで保護する操作を選択します。各オブジェクトタイプには、独自の操作リストがあります。ACL ルールは 1 つの操作のみを保護できます。複数の操作を保護するには、それぞれに個別の ACL ルールを作成します。 report_view操作のルールを作成する場合は、「 report_view のアクセス制御 」も参照してください。
意思決定タイプ	ACL の意思決定タイプを選択します。次の場合に許可 評価が正常に完了したときにアクセスを許可します。[次の場合を除き却下 **(Deny Unless**)] は、評価が成功しない限りアクセスを拒否します。詳細については、 「次の場合を除き却下 (Deny-Unless)」ACL を参照してください。
管理者優先	admin ロールを持つユーザーがこの ACL ルールの権限チェックに自動的に合格するようにするには、このチェックボックスをオンにします。admin ユーザーは、適用されるスクリプトまたはロールに関係なく合格します。ただし、ServiceNow の担当者のみが割り当てることができる **nobody** ロールは、admin 上書きオプションよりも優先されます。ACL に **nobody** ロールが割り当てられている場合、**[admin** 優先] が選択されていても、admin ユーザーはリソースにアクセスできません。「ベースシステムロール 」を参照してください。 アドミニストレーターがこの ACL ルールで定義された権限を満たして、保護されたオブジェクトにアクセスする必要がある場合は、このチェックボックスをオフにします。アドミニストレーターは常にロールチェックに合格するため ([必要なロール] フィールドの説明を参照)、条件ビルダーまたは [スクリプト] フィールドを使用して、アドミニストレーターが合格する必要がある権限チェックを作成します。
保護ポリシー	ACL に保護ポリシーを設定するには、これを選択します
名前	保護されるオブジェクトの名前 (レコード名またはテーブル名とフィールド名) を入力します。名前が具体的であるほど、ACL ルールも具体的になります。レコード、テーブル、またはフィールド名の代わりにワイルドカード文字のアスタリスク (*) を使用して、レコードタイプ、すべてのテーブル、またはすべてのフィールドに一致するすべてのオブジェクトを選択できます。ワイルドカード文字とテキスト検索を組み合わせることはできません。たとえば、inc* は有効な ACL ルール名ではありませんが、incident.* および *.number は有効な ACL ルール名です。 **注:** 青い三角形をクリックし、レコード名または保護されるオブジェクトのテーブル名とフィールド名を手動で入力します。ドロップダウンに表示されないオブジェクトを保護するには、このオプションを使用します。
説明	この ACL ルールによって保護されるオブジェクトまたは権限の説明を入力します。

フィールド	説明
アクティブ	この ACL ルールを適用するには、このチェック ボックスをオンにします。
詳細	[詳細条件] フィールドを表示するには、このチェックボックスをオンにします。ステップ 6 を参照してください。

5. オプション: ACL の適用範囲を絞り込むには、必要に応じて [条件] フィールドに入力します。

ロールが必要	このリストを使用して、ユーザーがオブジェクトにアクセスするために必要なロールを指定します。複数のロールをリストする場合、リストされているロールのいずれかを持つユーザーがオブジェクトにアクセスできます。必要なロールリストが関連リストとして表示されます。 **注:** admin ロールを持つユーザーは、他のすべてのロールが自動的に付与されるため、常にこの権限チェックに合格します。
データ条件	この条件ビルダー を使用して、ユーザーがこのオブジェクトにアクセスするために true でなければならないフィールドと値を選択します。 **注:** [条件] フィールドでは大文字と小文字が区別されます

6. オプション: [詳細] ボックスがオンになっている場合は、必要に応じて [詳細条件] フィールドに入力します

参照による制御	関連レコードに ACL を適用します。詳細については、「関連レコードへのアクセス」を参照してください。
スクリプト	オブジェクトにアクセスするために必要な権限を記述するカスタムスクリプトを入力します。スクリプトでは、現在および 前のビジネスルールのグローバル変数 およびシステムプロパティの値を使用できます。スクリプトは、次の 2 つの方法のいずれかで true または false の応答を生成する必要があります。 ○ true または false の値に設定された回答変数を返します。 ○ true または false に評価します。 いずれの場合も、スクリプトが true と評価され、ユーザーが ACL ルールの条件を満たしている場合にのみ、ユーザーはオブジェクトにアクセスできます。ユーザーがオブジェクトにアクセスするには、条件とスクリプトの両方が true と評価される必要があります。 [スクリプト] フィールドにスクリプトがある場合。このスクリプトは、フィールドがフォームに表示されていない場合でも実行されます。

> **注:** 評価されたアイテムが関連リストにある場合、現在のポイントは、ACL の現在のアイテムではなく、関連リストがあるアイテムを指します。ただし、ACL を評価しているアイテムが関連リストにない場合、現在のポイントは実際のアイテムを指します。

---

7. フォームヘッダーを右クリックし、[保存] を選択します。

「次の場合を除き却下 (Deny-Unless)」ACL

「次の場合を除き却下 (Deny-Unless)」ACL の詳細について説明します。

「次の場合を除き却下 (Deny-Unless)」ACL は「deny-unless」アプローチで評価されます。ACL は却下されないユーザーを定義します。別の言い方をすると、ロール、条件、およびスクリプト要件が満たされない限り、ユーザーはアクセスを拒否されます。

> **重要:** 「次の場合を除き却下 (Deny-Unless)」ACL は、ACL 評価で「次の場合に許可 (Allow-If)」ACL より優先されます。

「次の場合を除き却下 (Deny-Unless)」ACL は、 2 つの 結果を生成します

評価結果	結果
合格	定義されたロール、データ条件、セキュリティ属性、およびスクリプト要件が満たされています。ACL はさらに評価を進めます。 **重要:** 「次の場合に許可 (Allow-If)」ACL は、対象がリソースにアクセスできるようにするために、アクセス権を付与する必要があります。
失敗	「次の場合を除き却下 (Deny-Unless)」ACL は失敗としてマークされ、アクセスは拒否されます。

次に「次の場合を除き却下 (Deny-Unless)」ACL の例を示します。

- ACL にはロール sn_hr_core.manager と itil があります
- 条件はアクティブ = true です。
- スクリプトには解答 = gs.isLoggedIn(); があります。

この ACL の 3 つ の要件をすべて満たさない限り、ユーザーはアクセスを拒否されます。この「次の場合を除き却下 (Deny-Unless)」ACL に合格するには、ユーザーは sn_hr_core.manager ロールまたは itil ロールのいずれか、active field = true のレコードにアクセスしてログインしている必要があります。3 つの要件のいずれかが満たされていない場合、「次の場合を除き却下 (Deny-Unless)」ACL は失敗します。

ACL を許可

ACL (アクセス制御リスト) を許可する方法について説明します。

許可 ACL は、特定のルールで拒否されない限り、デフォルトでリソースへのアクセスを許可するように設計されています。これは、「Deny-Unless」モデルの反対です。

クエリー ACL

クエリー ACL を使用すると、データをクエリーできるユーザーを明示的に定義することで、より詳細なアクセス制御を定義できます。

クエリー ACL とは

クエリー ACL の操作は、query_range または query_match に設定されています。クエリー ACL を使用すると、ユーザークエリーをより細かく制御でき、 セットアップに基づいてアクセスを制限または有効化できます。クエリー ACL は、ブラインドクエリー攻撃 (データの値が見えていなくても、攻撃者がクエリーを盲目的に実行し、結果から情報を抽出する攻撃) に対抗する強力なツールです。

クエリー ACL を使用する状況

列に機密情報の値が含まれていて、データへの部分的/条件付きアクセスが許可されている場合は、データの機密性に基づいて、クエリー ACL の使用を検討し、必要に応じて実装する必要があります。テーブル内の行とその列への部分的/条件付きアクセスがある場合、特にそのアクセスがデータフィルターによって強制されていない場合は、データの機密性に基づいて、必要に応じてクエリー ACL を実装する必要があります。

注: 一部の行または列にアクセスでき、他の行や列にはアクセスできないユーザーがいる場合は、クエリー ACL の使用を検討してください。

Example: 給与クエリーの制御

給与テーブルには、ユーザー自身の給与が含まれている行がありますが、給与が 2 つの境界内に含まれているユーザーをクエリーする範囲クエリーをユーザーが発行することはできません。給与に対して query_range ACL を設定すると、ユーザーはそのようなクエリーを発行できなくなります。

Example: HR クエリーの制御

ユーザーは、すべての hr_profiles を表示できますが、SSN は自身のものしか表示できません。ユーザーが SSN をクエリーする必要性はありません。クエリーACL を使用することで、ユーザーが他人の HR プロファイルの SSN に対してクエリーを実行して SSN マッピングを抽出できないようにする必要があります。

クエリー ACL の動作

クエリー ACL は、query_match および query_range 演算子を使用することで、安全で詳細なテーブルクエリー動作を実現します。それぞれの動作を以下に説明します。

query_match

query_match は、EQUALS、NOT_EQUALS、IN、NOT_IN、SAMEAS、NSAMEAS、ANYTHING、ISEMPTYSTRING、で構成されます。query_match は、「安全な演算子」で構成されており、特定のレコードを取得するように構築されているため、他のレコードを返すために悪用されることがありません。

評価結果	結果
合格	ユーザーは照合クエリーを送信できます。
失敗	ユーザーは次の照合クエリーを送信できません。

評価結果	結果
	• EQUALS • NOT_EQUALS • IN • NOT_IN • SAMEAS • NSAMEAS • ANYTHING • ISEMPTYSTRING • ISEMPTY • ISNOTEMPTY • ISNULL • ISNOTNULL

query_range

query_range は、他のすべての要素 (STARTS_WITH、CONTAINS、>=、<= など) で構成されており、境界値を調整することでユーザーがより多くのレコードを照会できるため、危険性がより高くなります。

評価結果	結果
合格	ユーザーは範囲クエリーを送信でき、無制限にソートできます。
失敗	ユーザーは、(STARTS_WITH、CONTAINS、>=、<=) などを使用して範囲クエリーを送信することはできません。列によるソートは制限されています。

重要:

クエリー ACL (query_match と query_range の両方) は、デフォルトで、読み取りアクセスを委任する star.star ACL になります。つまり、ACL がクエリーに強制される場合、クエリー ACL が作成されていない場合は、列への読み取りアクセスが評価されます。クエリー ACL が定義されている場合は、デフォルトの動作が上書きされます。

埋め込みリスト内のレコードを保護する

埋め込みリストのレコードにセキュリティを適用するには、埋め込みリストのレコードの編集と削除を特定のロールに制限します。

始める前に

必要なロール：security_admin

手順

1. 移動先 すべて > システムセキュリティ > アクセス制御 **(ACL)**.
2. 該当するテーブルの書き込みまたは削除レコードを開きます。
3. フォームの [必要なロール] セクションで、そのテーブルに対する書き込み権限または削除権限を持つロールを追加します。
4. 変更内容を保存します。
   関連付けられたテーブルのレコードが埋め込みリストに表示されると、指定されたロールを持つユーザーのみが編集および削除のオプションを使用できます。

関連レコードへのアクセス

関連レコードへのアクセスにより、関連テーブル間でユーザーがアクセスできるレコードを一貫して制御できます。

関連レコードアクセスの詳細

関連レコードへのアクセスでは、関連レコードへのアクセスを構成し、ACL からのアクセスが許可されているデータに関連する親テーブルの ACL を適用するかどうかを決定できます。次のユースケースを考えてみましょう。ユーザーは ACL を介してレコードにアクセスでき、関連レコードアクセスにより、ユーザーは参照または双方向の関係を介して、関連する他のすべてのレコードにアクセスできるようになります。

コンテキスト依存セキュリティマネージャー

コンテキスト依存セキュリティマネージャーは、読み取り、書き込み、作成、および削除の認証を制御することにより、データを保護します。

長所

コンテキスト依存セキュリティマネージャーは、システムテーブル階層を認識しているため、階層内の表示場所に基づいてフィールドに特定のセキュリティルールを作成できます。コンテキスト依存セキュリティマネージャーのメリットは次のとおりです。

- コンテキスト依存セキュリティ：レコードの内容に基づいてレコードを保護します。
- 階層セキュリティ：オブジェクト階層内の任意のレベルにセキュリティルールを適用します。

フィールドとテーブルの保護

従来のシンプルセキュリティマネージャーでは、適切な辞書エントリにロールを追加することで、フィールドとテーブルを保護することができました。コンテキスト依存セキュリティマネージャーでは、これらの辞書ロールはテストされなくなりました。代わりに、フィールドとテーブルで ACL ルールを検索します。

⚠ **警告:** コンテキスト依存セキュリティマネージャーをインストールした後、ACL ルールを使用してフィールドとテーブルを保護する必要があります。フォームレイアウト や辞書フォームを設定して辞書エントリにロールを追加しても、権限は変更されません。

コンテキスト依存セキュリティおよびロール

ユーザーまたはグループにロールを付与できます。ただし、コンテキスト依存セキュリティマネージャーのインストール後、ユーザーレコードの [ロール] フィールドをオンにできなくなり、ユーザーフォームとグループフォームに表示されなくなります。代わりに、ユーザーまたはグループレコードではなく、ロール関連リストにロールを追加する必要があります。

アプリケーションとモジュールには、それらを表示するために必要なロールのリストが含まれています。たとえば、システム定義アプリケーションを表示するには、admin ロールが必要です。アプリケーションとモジュールのセキュリティ権限は、引き続きロールアレイを使用して定義されます。

カタログアイテムとカタログ変数の両方に、それらを表示するために必要なロールのリストが含まれています。カタログアイテムとカタログ変数のセキュリティ権限は、引き続きこれらのロールアレイを使用して定義されます。

コンテキスト依存セキュリティマネージャーでは、ロールの継承フラグが *true* に設定されている場合でも、グループに付与されたロールが自動的に継承されます。

コンテキスト依存セキュリティマネージャーの有効化

コンテキスト依存セキュリティマネージャーはベースシステムで有効です。ユーザーロールテーブルに重複するエントリが多数ある場合は、重複するロールを排除するために Contextual Security: Role Management V2 へのアップグレードが必要な場合があります。プラグインは次のとおりです。

Contextual Security: Role Management [com.glide.role_management]

コンテキスト依存セキュリティ機能を提供します。このプラグインは、自動的にインストールされます。

Contextual Security: Role Management V2 [com.glide.role_management.inh_count]

ユーザーロール [sys_user_has_role] テーブルで継承されたロールによるエントリの重複を防ぎます。このプラグインは、新しいインスタンスで自動的にインストールされ、アップグレードに対してアクティブ化することができます。Contextual Security: Role Management 機能拡張プラグインは、このプラグインの以前のバージョンです。ロール管理機能拡張プラグインには、RoleManagementVerify() スクリプトは含まれません。このスクリプトは、アップグレードによって実行される変更のリストを返し、プラグインによる変更を監視できるようにします。

注: Role Management V2 を有効にした後、 glide.role_management.v2.audit_roles システムプロパティを設定して、監査ロールテーブルがユーザーロールに関連する監査レコードを作成できるようにする必要があります。このプロパティの設定と監査ロールテーブルの詳細については、以下を参照してください。

- Contextual Security: Role Management V2 でのロール監査を有効にする。
- インスタンスセキュリティ強化設定の ハードニング設定。
- 監査ユーザーロール

Contextual Security: Role Management V2 による重複エントリの防止

他のロールから継承されたロールは、ユーザーロールテーブル [sys_user_has_role] の個々のエントリとして追加されるため、1 つのロールでエントリが重複する可能性があります。Contextual Security: Role Management V2 は重複エントリを排除し、今後、重複が生じるのを防ぎます。

継承数による重複エントリを排除する

Contextual Security: Role Management V2 は、継承数 (inh_count) 列を使用して、ロールが別のロールまたはグループから継承された回数を追跡します。ユーザーロール [sys_user_has_role] テーブルでは、ユーザーは特定のロールを 1 回のみ継承できるため、エントリが重複することはありません。継承数 (inh_count) 列は読み取り専用で、ユーザーがロールを継承する回数を計算します。

アクティベーションの変更

Contextual Security: Role Management V2 は新しいインスタンスに自動的にインストールされ、アップグレードに対して有効にすることができます。有効にすると、コンテキスト依存セキュリティと Contextual Security: Role Management 機能拡張の両方が Contextual Security: Role Management V2 に置き換わります。

Contextual Security: Role Management V2 が有効になっている場合、次の列の使用は廃止されますが、下位互換性のためにユーザーロールテーブルに残ります。

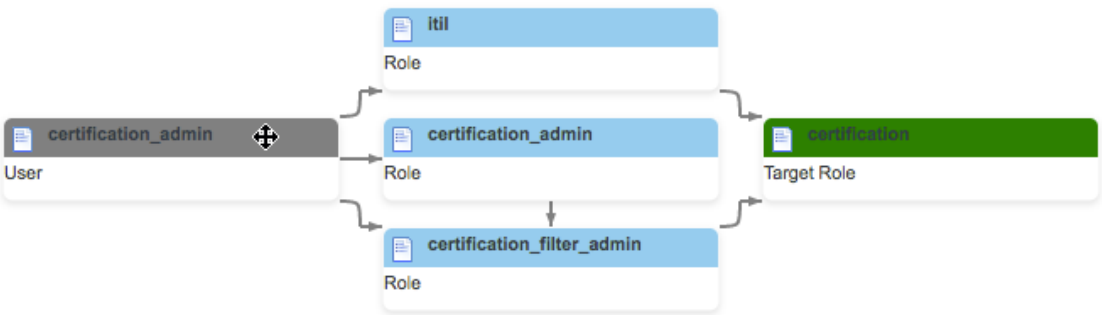
- granted_by (ロールの委任でのみ使用)
- included_in_role
- included_in_role_instance

警告: これらの列がインスタンスのカスタムスクリプトで使用されている場合は、Role Management V2 にアップグレードしないでください。

ロール継承マップによるロール継承の可視化

ロール継承マップには、継承されたロールが視覚的に表示されます。このマップを使用して、継承数 (inh_count) 列に表示されるロールを理解することができます。ロール継承マップを表示するには、ユーザーロール [sys_user_has_role] テーブルを設定してロール継承マップ列を表示します。

ロール継承マップ



注: グループまたはロールのアサインを同時に更新すると、継承数が正しくない可能性があります。正確な継承数を取得するには、 glide.security.inh_count_patcher.enabled プロパティを有効にする必要があります。

Contextual Security: Role Management V2 へのアップグレード

Contextual Security: Role Management V2 は新しいインスタンスに自動的にインストールされます。Contextual Security: Role Management から Contextual Security: Role Management V2 にアップグレードして、ユーザーロールテーブル内の重複するロールを排除し、今後、重複が起こらないように防ぐことができます。

始める前に
必要なロール：admin、security_admin
有効な security_admin ロールを取得するには、admin ロールと 特権ロールへの昇格 が必要です。

注: Contextual Security: Role: Management V2 にアップグレードする前に、監査ロールテーブルを有効にして、ユーザーロールに関連する監査レコードを作成する必要があります。必要なシステムプロパティの設定の詳細については、「Contextual Security: Role Management V2 でのロール監査を有効にする」を参照してください。

このタスクについて
この手順では、コンテキスト依存セキュリティロールのアップグレード方法と、次の表で説明されている関連プラグインを有効にする方法について説明します。

Contextual Security: Role Management V2 のプラグイン

プラグイン	説明
Contextual Security: Role Management V2 [com.glide.role_management.inh_count]	ユーザーロール [sys_ user_has_role] テーブルでエントリが重複しないようにします。 security_admin ロールまたは昇格した権限を持つユーザーは、プラグインを有効にするか、カスタマーサービス & サポート に連絡する必要があります。
Contextual Security: Role Management V2 REST API [com.glide.role_management.inh_count.rest_api]	ロール管理の API 機能を有効にします。

Contextual Security: Role Management から Contextual Security: Role Management V2 にアップグレードする前に、スクリプトを実行してアップグレードの結果をテストします。このスクリプトは、アップグレードによって実行される変更のリストを返します。変更を受け入れる場合は、Contextual Security: Role Management V2 プラグインをインストールします。変更を受け入れられない場合は、Contextual Security: Role Management V2 プラグインをインストールしないでください。または、アップグレードを実行してから、必要な変更を手動で行うこともできます。

手順

1. アップグレードの前に次のスクリプトを実行して、アップグレードの影響をテストします。

a. 移動先 システム定義 > スクリプト - バックグラウンド.

b. グローバルスコープで次のスクリプトを実行します。

```
new RoleManagementVerify().verifyInheritedRoles();
```

大きな sys_user_has_role テーブルの場合、実行が完了するまでに数時間かかることがあります。実行中にユーザーロールを編集したり、追加したりしないでください。
テストデータに基づく結果の例：

```
*** Script: 2016-12-01 19:58:54 Starting checking of inherited roles for all users...
*** Script: User: itam, inherited roles to be ADDED: financial_mgmt_user
*** Script: User: bernard.laboy, inherited roles to be DELETED:
 api_analytics_read,pa_viewer,rest_api_explorer,a123
*** Script: User: bernard.laboy, inherited roles to be ADDED: dependency_views
*** Script: Number of inherited-role records in sys_user_has role, current: 260, after
 re-calculation: 258
*** Script: Number of users with discrepancies for inherited roles: 2
*** Script: 2016-12-01 19:58:55 Finished checking of inherited roles for all users!
```

c. スクリプトの結果を評価して、提案された変更を受け入れられるかどうか判断します。

2. Contextual Security: Role Management V2 プラグインをアクティブ化します。

重要: security_admin ロールまたは昇格した権限を持つユーザーは、プラグインを有効にするか、カスタマーサービス & サポート に連絡する必要があります。

a. 移動先 システム定義 > プラグイン.

b. プラグイン名を見つけてクリックします。

c. [システムプラグイン] フォームでプラグインの詳細を確認し、[有効化/アップグレード] 関連リンクをクリックします。

d. [アクティブ化] をクリックします。

結果
Role Management V2 を有効にすると、スクリプトの結果に示されている変更が有効になります。ユーザーロールテーブルの継承数 (inh_count) 列は読み取り専用で、ユーザーがロールを継承した回数を自動的に反映します。

Contextual Security: Role Management V2 でのロール監査を有効にする

システムプロパティを設定して、監査ロールテーブルでユーザーロールに関連する監査レコードを作成できるようにします。

始める前に
必要なロール：admin

このタスクについて
有効にすると、監査ロール [sys_audit_role] テーブルでユーザーレコードへの変更が保持されます。ロールの監査の詳細については、「監査ユーザーロール」を参照してください。Contextual Security: Role Management V2 [com.glide.role_management.inh_count] プラグインがインストールされている場合は、システムプロパティを **true** に設定してロール監査を有効にする必要があります。

手順

1. システムプロパティ [sys_properties] テーブルに移動します。
2. *glide.role_management.v2.audit_roles* システムプロパティを追加し、**true** に設定します。

   Contextual Security: Role Management V2 [com.glide.role_management.inh_count] プラグインがインストールされている場合、このプロパティを **true** に設定すると、ユーザーロールを変更するときに監査ロール [sys_audit_role] テーブルを有効化して、レコードを作成することができます。

フォーム送信の再確認

現在のユーザーが特定のフィールド (task.number など) に書き込むべきではないとシステムで判断されると、そのフィールドは読み取り専用モードで表示されます。ほとんどのインシデントで番号フィールドに書き込みができないのはそのためです。

受信フィールドの値を書き込めるかどうか再確認するように設定すると、トランザクションの受信レグに同じ一連のセキュリティルールが適用されます。たとえば、インシデントを送信すると、変更を投稿する前に番号フィールドへの書き込みが可能かどうか再確認します。

受信トランザクションを再確認しないように指示すると、クライアントが送り返すトランザクションであれば、名目上の読み取り専用フィールドへの書き込みが許可されます。多くの展開でこれは望ましい動作です。たとえば、クライアントスクリプトを使用して、他の書き込み可能なフィールドでのユーザーの選択に応じて、名目上の読み取り専用フィールドを設定している場合などが該当します。

プロパティ	ロケーション	デフォルト
フォーム送信時に着信トランザクションのセキュリティを二重確認 (フォーム生成時には常に権限が確認されます)	システムプロパティ > セキュリティ	無効 (ダブルチェックなし)

デフォルトの拒否プロパティ

デフォルトの拒否プロパティ (glide.sm.default_mode) は、一致する ACL ルールがワイルドカードテーブル ACL ルールのみの場合に、セキュリティマネージャーのデフォルトの動作を制御します。

最も一般的なレコードベースの操作 (読み取り、書き込み、作成、および削除) 用の一連のワイルドカードテーブル ACL ルール。システムテーブルへのロールベースでのアクセスを提供する ACL も多数用意されています。たとえば、business_rule_admin ロールに sys_script アクセス権を付与する ACL があります。このロールはビジネスルールを管理できると文書化されています。

glide.sm.default_mode プロパティを使用して、すべてのテーブルでこれらの操作を拒否または許可します。

- アクセスを拒否：ワイルドカードテーブル ACL ルールは、ユーザーが admin ロールを所有しているか、別のテーブル ACL ルールの要件を満たしている場合を除き、読み込み、書き込み、作成、削除の操作をすべてのテーブルで制限します。report_on や personalize_choices などの他の操作は、この設定の影響を受けません。
- アクセスを許可：ワイルドカードテーブル ACL ルールは、そのような操作を制限する特定のテーブル ACL ルールが存在する場合を除き、読み込み、書き込み、作成、および削除の操作をすべてのテーブルで許可します。

[アクセスを拒否 **(Deny Access)]** に設定すると、glide.sm.default_mode を [アクセスを許可 **(Allow Access)]** にリセットすることはできません。

注: デフォルトでは、ワイルドカードテーブル ACL ルールのみが glide.sm.default_mode プロパティの値をチェックする ACL ルールです。この設定で他の操作を制御する場合は、独自の ACL ルールを作成してこのプロパティ値をチェックします。

このプロパティの詳細については、「インスタンスセキュリティ強化設定」の「空の ACL でデフォルトで拒否する (Security Center 1.3 で更新)」を参照してください。

ACL の詳細設定

新しい ACL の作成や既存の ACL の変更に加えて、ACL 機能の他の側面を設定できます。

ACL ルールの却下

次の場合を除き却下 (Deny-Unless)

ACL ルールのクエリー

タスク	説明
ACL スクリプト条件を参照フィールドに適用する	参照フィールドがフォームまたはリストに表示するデータへのアクセスを制御する場合は、プロパティを有効にしてスクリプト条件を許可し、参照

タスク	説明
	フィールドに適用します。プロパティを有効にすると、インスタンスのパフォーマンスに影響を与える可能性があります。
ACL を AJAXGlideRecord (クライアント側の Glide レコード) に追加	現在接続しているユーザーがアクセス権を持つデータのみを照会するように、`GlideAjax` API 呼び出しに ACL を適用します。
アクセスレベルでアドミン優先を評価する	アクセスレベルで admin の ACL 評価を強制します。デフォルトでは、アドミン優先オプションが ACL ルールフォームで選択されている場合、admin ロールのユーザーはこの ACL ルールの権限確認に自動的に合格します。
ACL のデバッグツールとトラブルシューティングツールを使用する	ACL 監視、フィールドレベルのデバッグ、アクセス ACL ルール出力メッセージなどのツールを使用すると、ACL のトラブルシューティングとデバッグに役立ちます。

外部ユーザーへのテーブルアクセス権の提供

snc_external ロールのみを持つユーザーがテーブルのリストビューにアクセスできるようにするには、一連の ACL を作成する必要があります。

始める前に
必要なロール：security_admin

手順

1. 特権ロールへの昇格。
2. 次の設定を使用して、ACL ルールを作成します。
   - タイプ：ui_page
   - 操作：読み取り
   - 名前：{table_name}_list
   - 必要なロール：snc_external
3. テーブルのデフォルトの読み取り ACL で、[必要なロール] リストに **snc_external** を追加します。
   ACL がまだない場合には作成します。
4. これらの設定を使用して、別の ACL を作成します。
   - タイプ：ui_page
   - 操作：読み取り
   - 名前：{table_name}
   - 必要なロール：snc_external
5. これらの設定を使用して、ユーザーにテーブル内のフィールドへの書き込みアクセス権を付与する別の ACL を作成します。
   - タイプ：レコード
   - 操作：作成

- 名前：{table_name} {column_name}
- 必要なロール：snc_external

ユーザーに書き込みアクセス権を付与するすべてのフィールドに対して、この手順を繰り返します。列名の代わりにアスタリスク (*) を使用すると、すべてのフィールドへのアクセスを一度に提供できます。

ACL スクリプト条件を参照フィールドに適用する

glide.sys_reference_row_check システムプロパティを使用して、参照フィールドのスクリプト化された条件を有効にします。

デフォルトの動作は、インスタンスのパフォーマンスを向上させることを目的としています。参照フィールドのスクリプト条件を有効にするには、次のシステムプロパティを追加します。

注: システムプロパティの作成の詳細については、「システムプロパティを追加する」を参照してください。

システムプロパティ

プロパティ	説明
glide.sys_reference_row_check	アクセス制御ルールのスクリプト条件がテーブルの [参照] フィールドに適用されるかどうかをコントロールします。 • タイプ：true | false • デフォルト値：**false** • 場所：システムプロパティ [sys_properties] テーブルにシステムプロパティを追加します。

注: *glide.sys_reference_row_check* システムプロパティが存在しない場合、または false に設定されている場合、アクセス制御ルールのスクリプト条件は適用されません。これは、他の ACL 基準 (ロール要件など) が満たされている限り、スクリプト化された条件を含む ACL がチェックに合格することを意味します。

ACL を AJAXGlideRecord (クライアント側の Glide レコード) に追加

クライアントスクリプト内で GlideAjax API を使用してサーバー側のレコード (テーブルなど) にアクセスする場合、システムプロパティを使用してアクセス制御リスト (ACL) ルール検証を実行します。

アクセス制御リスト (ACL) を *GlideAjax* API 呼び出しに適用することを選択すると、現在接続しているユーザーがアクセス権限を持つデータのみを照会できます。たとえば、ユーザーが cmn_location テーブルを読み込む権限のない ESS ユーザーとしてログインすると、*GlideAjax* API 呼び出しは失敗します。

ServiceNow AI Platform が GlideAjax ACL 呼び出しのチェックなしで実行されている場合、API は、現在ログインしているユーザーが他の方法ではアクセスできない情報を返す可能性があります。

データのクエリを実行するときに GlideRecordSecure を使用して、最高レベルのセキュリティを確保します。GlideRecord は構成による ACL の適用に依存しますが、GlideRecordSecure はより厳格なセキュリティ制御を適用します。GlideRecordSecure は、機密データを処理するための、より安全ですぐに使えるソリューションを提供します。

注: このプロパティを設定 システムプロパティ > セキュリティ.

プロパティ	デフォルト
スタンダードセキュリティ ACL を *AJAXGlideRecord* 呼び出しに適用する	ACL チェックの強制

⚠ 警告: sys_class_name、sys_id、および sys_domain は ACL チェックで無視されます。

このプロパティの詳細については、「インスタンスセキュリティ強化設定」の「AJAXGlideRecord ACL チェックを必須とする (Security Center 1.3 で更新)」を参照してください。

アクセスレベルでアドミン優先を評価する

アクセスレベルで admin 優先の ACL 評価を強制する場合は、システムプロパティを追加できます。

始める前に
必要なロール：security_admin

このタスクについて
ACL は累積的に評価されます。指定されたフィールドに多数の ACL があり、そのうちの 1 つで **admin** 優先オプションが **false** (未選択) の場合、すべての ACL の有効な admin 優先は **false**と見なされます。これによりアドミニストレーターは優先を有効にする必要がある ACL を渡すことができなくなります。

手順

システムプロパティテーブルに次のプロパティを追加します。

プロパティ	説明
glide.security.admin.override.accessterm	アクセス期間レベルで admin 優先条件を評価します。 • タイプ：true | false • デフォルト値：新しいインスタンスの場合は **true**、アップグレードされたインスタンスの場合は **false** • 場所：システムプロパティ *[sys_properties]* テーブルに追加 ℹ 注: インスタンスでプロパティが定義されていない場合、値は false と評価されます。

ACL デバッグツール

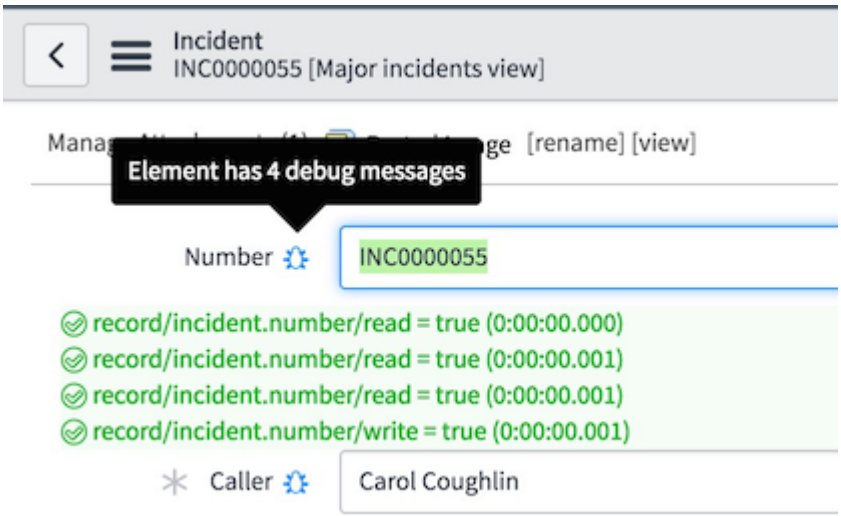
ACL のトラブルシューティングとデバッグに役立つ、フィールドレベルデバッグおよびアクセス ACL ルール出力メッセージを使用できます。ACL 構成監視を使用すると、変更時にどのような関連する ACL が存在するかがわかります。

フィールドレベルのデバッグ

デバッグを有効にすると、各フィールドの横に小さなバグアイコン (🐞) が ACL ルールと一緒に表示されます。アイコンをクリックすると、フィールドに適用される ACL ルールと評価結果が一覧

表示されます。に移動してデバッグを有効にする システムセキュリティ > デバッグ > セキュリティルールのデバッグ.

インシデントのフィールドレベルのセキュリティ



ACL デバッグを有効にした後、別のユーザーとして操作を行い、ユーザーが合格または不合格になった ACL ルールを確認することができます。ユーザーの代理操作を行うと、そのユーザーに表示が許可されているもののみが表示されます。たとえば、ACL が非表示にしているレコードを表示することはできません。デバッグを容易にするために、特定の ACL 関連テーブルへの読み取り専用アクセスは、テーブルへの読み取りアクセス権を持たないユーザーの代理操作を行う場合でも、デフォルトで有効になっています。この機能を変更するには、次のプロパティを **false** に設定します。

ACL ルールのデバッグを有効にするには、次に移動します: システムセキュリティ > セキュリティルールのデバッグ.

システムプロパティ	説明	デフォルト設定
glide.security.access_acl_as_impersonator	ユーザーの代理操作中に、sys_security_acl テーブル、sys_security_operation テーブル、sys_security_type テーブル、および sys_user_role テーブルへの読み取りアクセスを許可します。その結果、代理操作をしているユーザーは、代理操作されたユーザーが読み取ることができないデータを読み取ることができます。	true **注:** プロパティが false に設定されている場合、代理操作されたユーザーは ACL 関連のデータを読み取れない可能性があります。この場合、ACL をデバッグするには、admin または security_admin としてログインする 2 番目のセッションが必要となる可能性があります。

ACL ルール出力メッセージ

ACL デバッグでは、各リストとフォームの下部に ACL ルールの出力メッセージが表示されます。出力メッセージには以下が表示されます。

メッセージ要素	説明
TIME	この ACL ルールの処理に要した合計時間。
パス	<ACL ルールタイプ>/<ACL ルール名> /<操作> の形式で各 ACL ルールを一意に識別する情報。
コンテキスト	ACL ルールで評価されるオブジェクト。
RC	ACL ルールのリターンコード。true の値は、ACL ルールを満たしています。false の値は、ACL ルールを満たしていません。
ルール	プロセッサーとスクリプトの簡単な概要と、それに続く各テーブルレベルとフィールドレベルの ACL 評価の ACL 結果。ほとんどの ACL 評価では、総合的な合格または不合格の結果の後に、ACL 基準のタイプごとに結果のブレークダウンが表示されます。 • iAccessHandler：プラットフォーム上の非表示のソースコードを使用する内部システムチェック。これは変更できないシステムセキュリティチェックです。IAccessHandler は、ACL を評価せずにリソースへのアクセスを許可または拒否できます。IAccessHandler が無視されると、ACL が評価されます。どのような方法でも IAccessHandler チェックは変更できません。たとえば、IAccessHandler 実装はアプリケーションリソースのアクセスチェックに使用され、これは変更できません。 このメソッドは Istanbul 以降のリリースで利用できます。 • ロール：ユーザーが正しいロールを持っていることの検証。 • 条件：ユーザーが ACL ルールで指定された条件 (存在する場合) を満たしていることの検証。 • スクリプト：ユーザーが ACL ルール (存在する場合) で指定されたスクリプトを満たしていることの検証。

表示されるアイコンは、ACL がどのように評価されたかを示します。

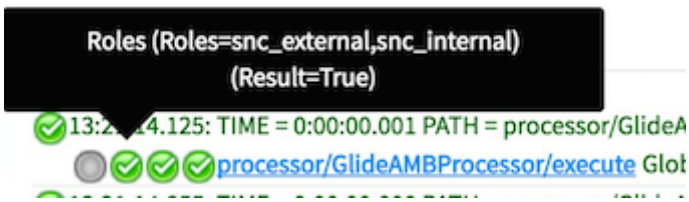
アイコン	説明
緑色のチェックマーク ()	基準を満たしているテーブルまたはフィールドを示します。
赤色の x アイコン ()	テーブルまたはフィールドが基準を満たしていないことを示します。
空の灰色の円アイコン ()	ACL 評価を実行する必要がなかったことを示します。
青色のチェックマーク、x、または空の円	ACL が以前の ACL チェックのキャッシュ結果から取得されたことを示します。アイコンの意味は上記と同じです。

ACL デバッグ出力では、次のアクションを実行できます。

- デバッグ出力の上部にあるチェックボックスをオンまたはオフにします。
  - セキュリティルール：ACL チェックの結果を表示または非表示にします。
  - その他：その他の警告またはメッセージを表示または非表示にします。

- 出力メッセージの横にある ACL の名前をクリックして、その ACL レコードを開きます。

- 4 つの ACL チェックのいずれかのアイコンにカーソルを合わせると、詳細が表示されます。



ACL トラブルシューティングリファレンス

一般的な ACL ルールのエラーとそのソリューションのリスト。

問題のトラブルシューティングに役立つデバッグを有効にします。

トラブルシューティング

エラーまたは症状	解決策
カスタムテーブルからレコードにアクセスすることはできません。	カスタムテーブルのテーブル ACL ルールを作成し、ユーザーにテーブルへのアクセスを許可します。明示的なテーブル ACL ルールがない場合、ユーザーはテーブルワイルドカード (*) ACL ルールの権限を渡す必要があります。これにより、デフォルトではアドミンのみにアクセスが制限されます。デバッグを有効にし、カスタムテーブルに対して評価する ACL ルールを決定します。
正しく機能しないカスタム ACL ルールを作成します。	最も可能性の高い問題は、処理順序でカスタムルールより別のルールが優先されるか、ユーザーがオブジェクトタイプのすべての権限要件を満たしていないといった問題です。デバッグを有効にし、ACL ルールが評価されていることを確認します。
フィールドの ACL ルールが正しく機能しません。	ユーザーが満たしていないテーブル ACL ルールが存在する可能性があります。デバッグを有効化し、どの ACL ルールをフィールドに対して評価するかを決定します。競合するテーブル ACL ルールまたは重複フィールド ACL ルールがないことを確認します。
テーブル ACL ルールが正常に機能していません。	処理順序の上位にある ACL ルール、またはテーブル ACL ルールを妨げる重複テーブル ACL ルールがあります。デバッグを有効にして、どの ACL をテーブルに対して評価するかを決定します。
リスト内のフィールドは表示できますが、フォームには表示できません。	ACL ルール条件またはスクリプトがリスト内でトリガーされていても、フォームではトリガーされていない可能性があります。デバッグを有効にして、ACL ルールが true と評価されるタイミングを決定します。リストとフォームで同じ動作になるように条件またはスクリプトを更新します。
プロセッサーまたはクライアント呼び出し可能スクリプトインクルードを実行しようとすると、エラーメッ	ユーザーが満たしていないプロセッサーまたはクライアント呼び出し可能スクリプトインクルードの ACL ルールがあります。ユーザーがオブジェクトにアクセスできる必要がある場合は、デバッグを有効にして、プロセッサーまたはスクリプトインクルードに対して評価される ACL ルール

トラブルシューティング (続く)

エラーまたは症状	解決策
セージが表示されます。	を決定します。オブジェクトにアクセスするために、必要に応じて ACL ルールまたはユーザーロールを更新します。

ACL 構成監視

ACL 構成監視を使用すると、同じテーブルで ACL を挿入、更新、または削除するときに、テーブルにどのような関連する ACL が存在するかがわかります。

ACL 構成監視は、アクセス制御 [sys_security_acl] テーブルに重要な変更を加えるたびに表示されるインターセプターウィンドウです。変更している ACL に関連する ACL を表示できるセキュリティルールのサマリーウィンドウが表示されます。セキュリティルールのウィンドウから ACL を変更することはできません。変更を加えるには、監視ウィンドウを閉じて、ACL に移動します。

次の状況では、ACL 構成監視は表示されません。

- 実際に変更を加えずに ACL レコードを保存または更新した場合。
- スクリプト、条件、アドミン優先オプションなどの (挿入や削除ではなく) マイナーな更新を行う場合。
- ACL レコードがアクティブではない場合。

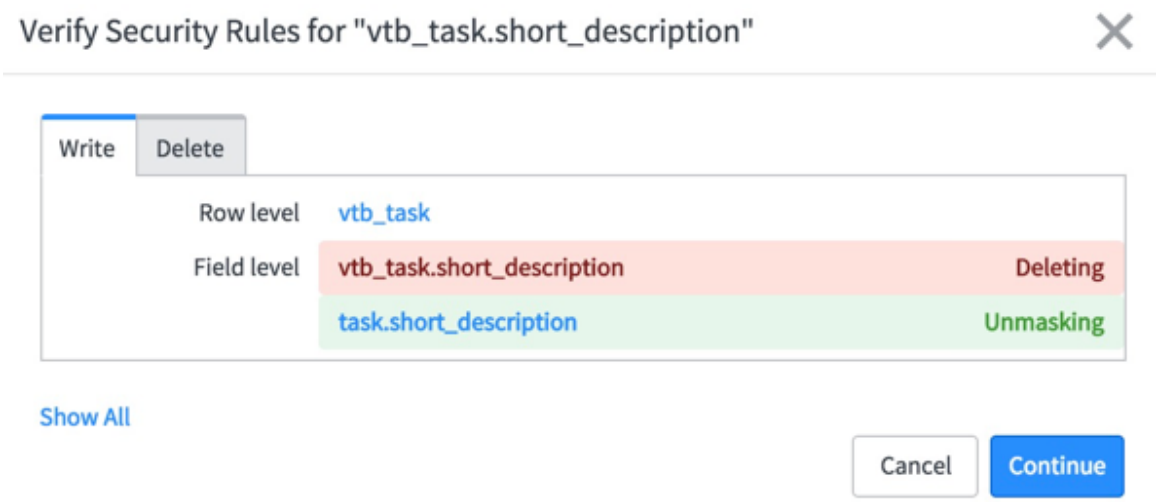
ACL セキュリティルールウィンドウ

構成監視では ACL 実行計画が表示されます。実行計画はセキュリティルールのポップアップウィンドウに表示されます。次のような情報を表示できます。

ACL 構成ウィンドウの要素

アイテム	説明
赤のハイライト	削除または非アクティブ化された ACL。
青のハイライト	変更された ACL。
緑のハイライト	追加された、またはアクティブになった ACL。
マスク済み	変更を加えるまで有効であった ACL。
マスク解除	変更を加えたときに有効だった ACL。

構成監視の例



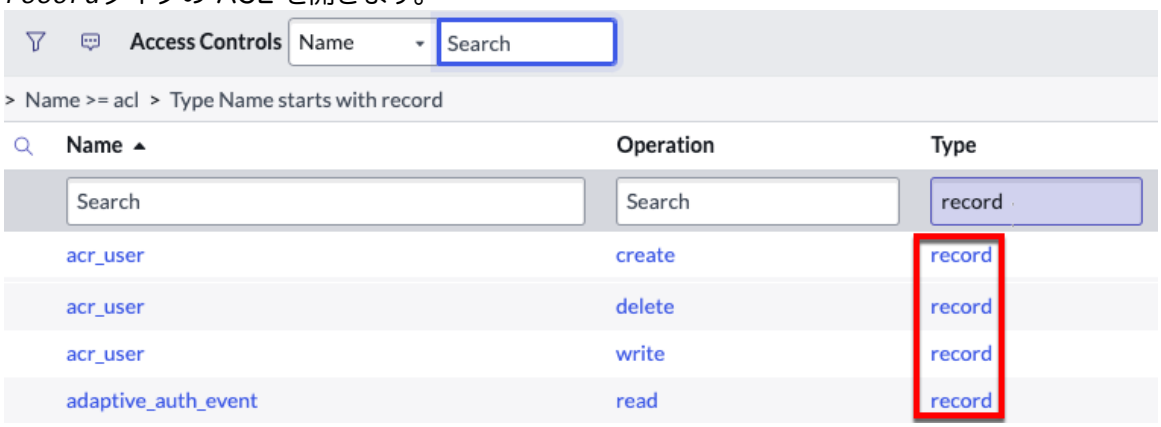
ACL 実行計画を表示する

アドミニストレーターは、インスタンス内の ACL の実行計画を表示することで、ACL の相互関係を確認できます。

始める前に
必要なロール：security_admin

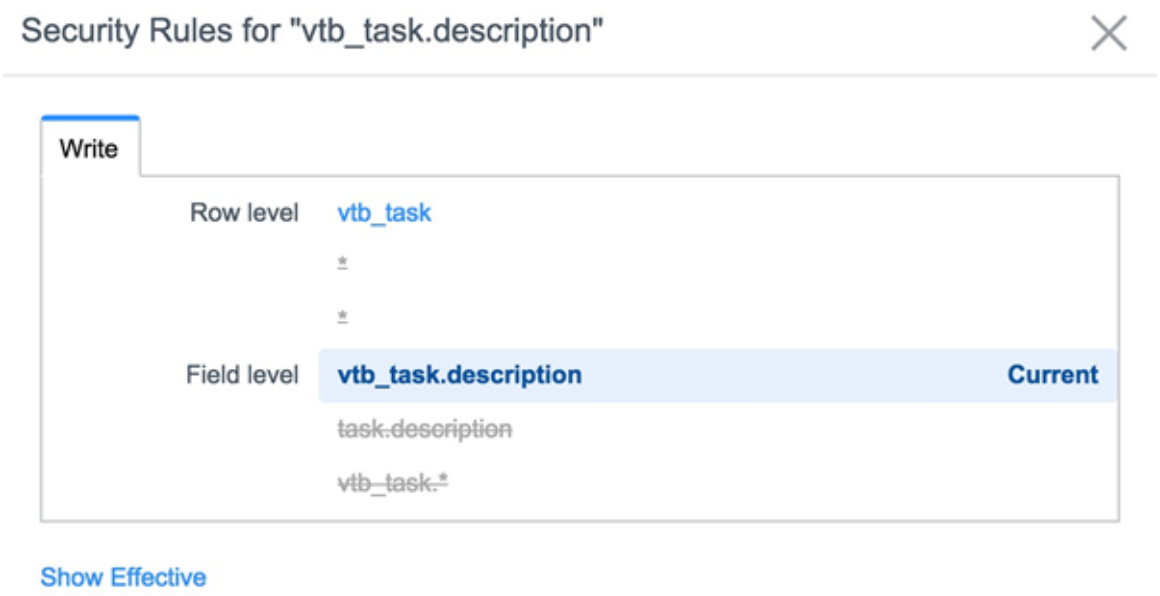
手順

1. 特権ロールへの昇格。
2. 移動先 システムセキュリティ > アクセス制御 **(ACL)**.
3. *record*タイプの ACL を開きます。



4. [関連リンク] セクションで **[ACL** 実行計画を表示] をクリックします。

   ACL のセキュリティルールウィンドウが表示されます。この例では、「vtb_task」の実行計画が表示されます。



ACL 実行計画ウィンドウ

UI アイテム	説明
タイトル	ACL の名前。
タブ名	ACL が作成、読み取り、書き込み、または削除の場合。
行レベル	このテーブルで実行される行レベルの ACL。
フィールドレベル	このフィールド (またはテーブル内の列) でのみ実行されるフィールドレベルの ACL。

5. [すべて表示] をクリックすると、上書きされる ACL やすべてのレコードに適用される汎用 ACL を含む、関連するすべての ACL が表示されます。
上書きされる ACL の名前には線が引かれています。汎用 ACL の名前にはワイルドカード文字のアスタリスク (*) が付いています。
6. [有効な **ACL** を表示 **(Show Effective)]** をクリックして、表示している ACL に関連する直接の ACL のみを表示し、ACL テーブルの拡張元のテーブルの ACL と汎用ワイルドカード (*) ACL を非表示にします。

ACL 構成監視を使用する

security_admin ロールに昇格した後、ACL 構成監視を使用します。

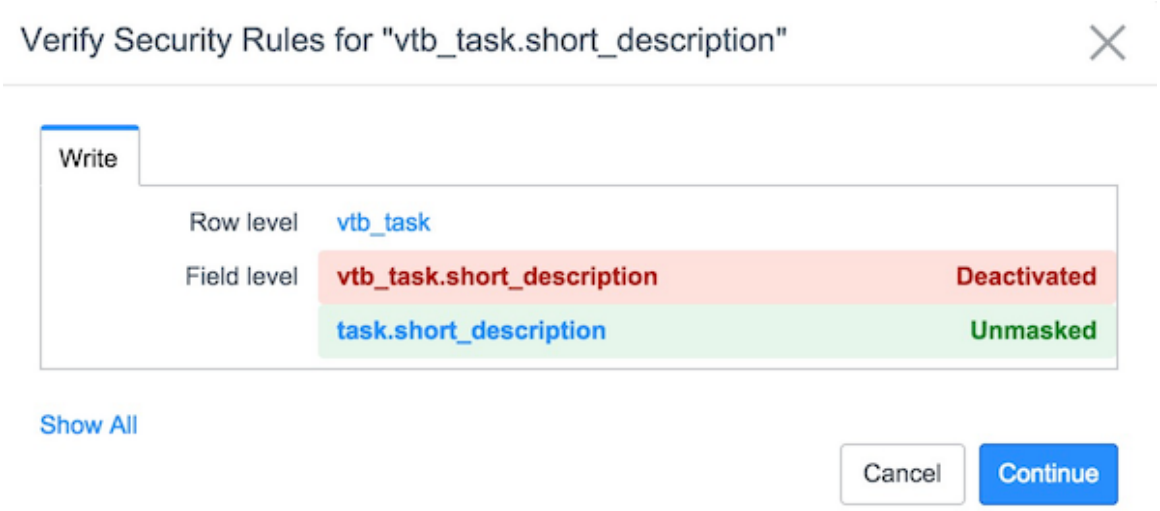
始める前に
必要なロール：security_admin
特権ロールへの昇格

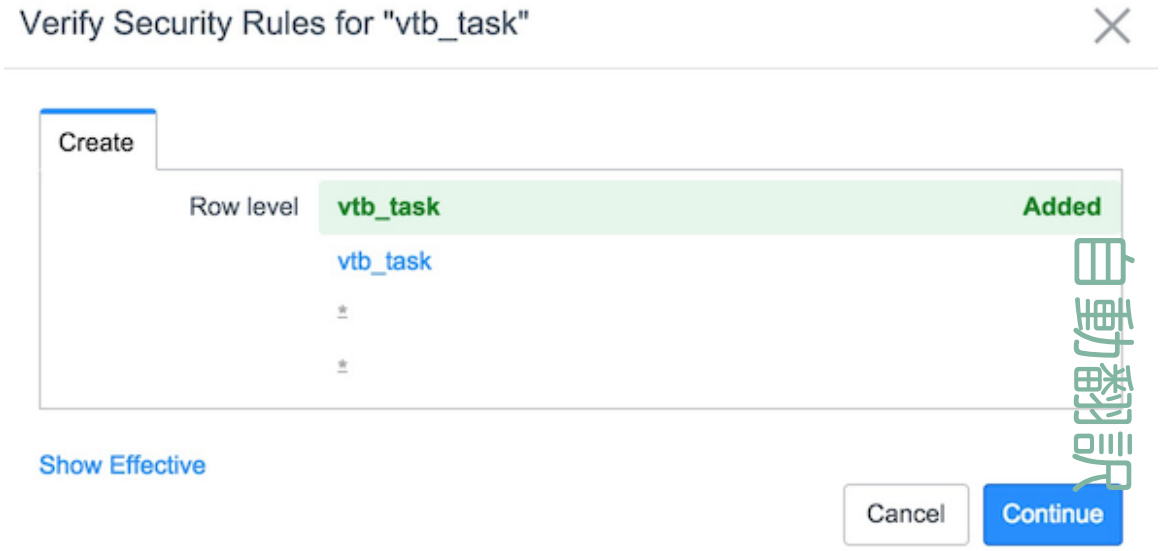
手順

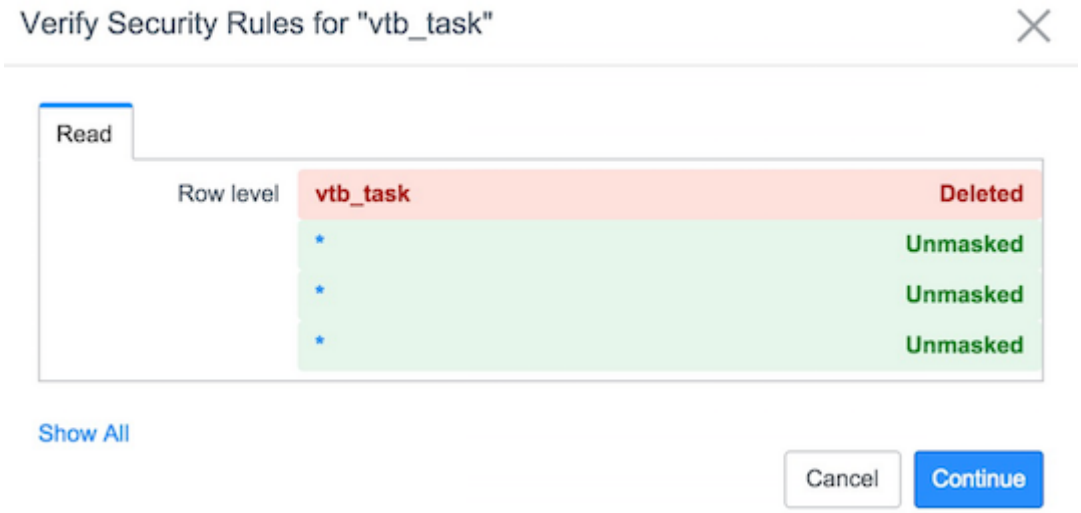
1. レコードタイプの ACL を開きます。
2. 変更や、[挿入] などのコンテキストメニューからのオプション選択など、ACL に対してアクションを実行します。
3. アクセス制御フォームで値を変更した場合は、ヘッダーを右クリックして [保存] を選択するか、[更新] または [削除] をクリックします。

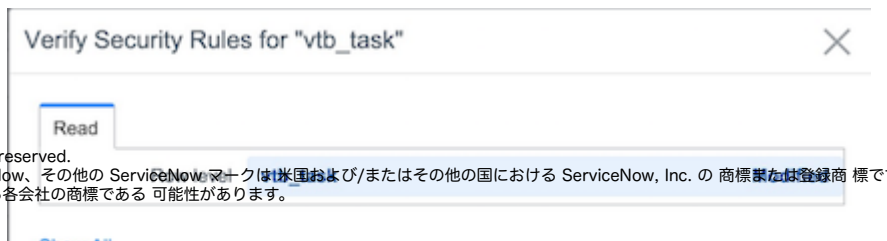
[セキュリティルール] ウィンドウが表示されます。ACL でデータベースアクションがまだ実行されていないため、変更はまだ保存されていません。

以下は、Visual Task Boards アプリケーションのプライベートタスク [vtb_task] テーブルのセキュリティルールの例です。このウィンドウのアイテムの説明については、「ACL 構成監視」を参照してください。







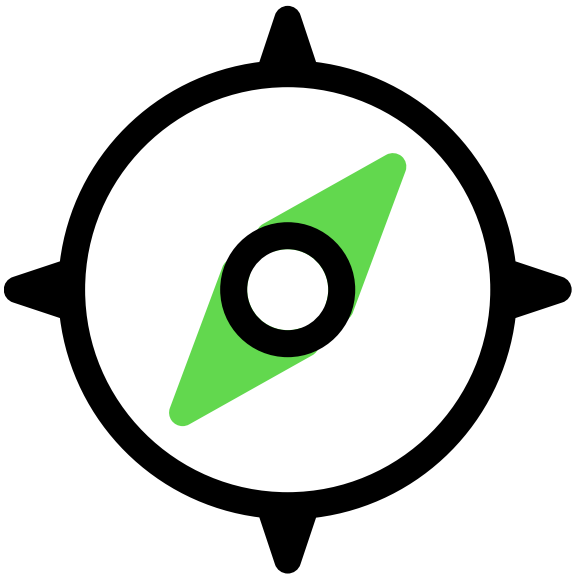
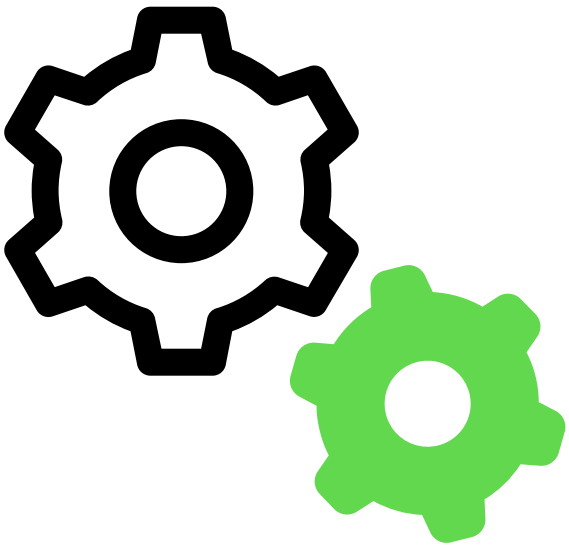


4. 実行計画の場合と同様に、[すべて表示] をクリックすると、上書きされたものやすべてのレコードに適用される汎用 ACL を含むすべての関連する ACL が表示されます。[有効な **ACL** を表示 **(Show Effective)]** をクリックすると、表示している ACL に関連する直接の ACL のみが表示されます。
5. いずれかの ACL にカーソルを合わせると、説明が表示されます。

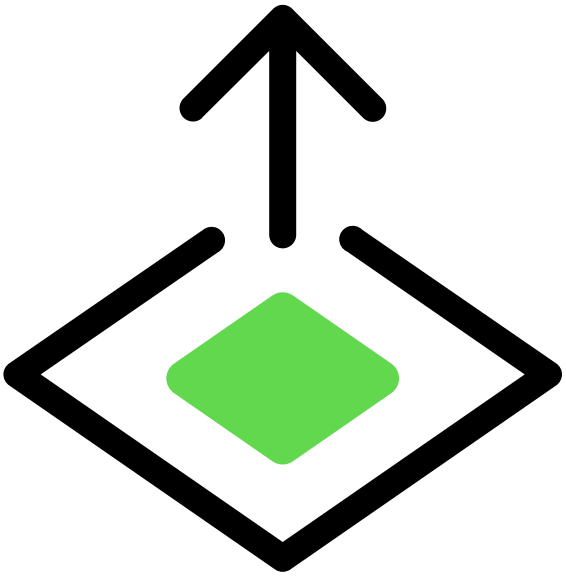
セキュリティ属性

セキュリティ属性は、アクセス制御リストの柔軟な代替手段を提供します。

開始するには

セキュリティ属性の詳細	セキュリティ属性の作成
	
セキュリティ属性の基礎を学ぶ	新しいセキュリティ属性を作成

OOB (Out-of-Box) セキュリティ属性



OOB セキュリティ属性の説明

セキュリティ属性のログ記録

セキュリティ属性のログ記録の確認

セキュリティ属性の基礎

セキュリティ属性は、サブジェクトまたはその環境に関する高度に構成可能な情報であり、アクセス制御で使用されると、複雑ではない方法できめ細かなセキュリティ構成が可能になります。

概要

セキュリティ属性は、現在使用されている ACL (アクセス制御リスト) 構成に対するロール定義を介したアクセス制御の代替手段を提供します。セキュリティ属性は、ACL ベースの設定にいくつかの利点をもたらします。

セキュリティの向上
: 権限の評価の難読化の改善により、組織のセキュリティが確保されます。

人間が判読可能
: セキュリティ属性は、セキュリティ権限の作成と使いやすさを簡素化するように設計されています。

柔軟なセキュリティ
: 顧客定義のペルソナと組み合わせて、すぐに使える構成からペルソナ定義をビルドします。

ログ記録と監査
: セキュリティ属性は、詳細な監査とログ記録を提供し、セキュリティ対策と理論に関するより多くのインサイトを提供します。

セキュリティ属性の作成

ステップバイステップガイドを使用して、新しいセキュリティ属性を作成します。

始める前に
必要なロール：security_admin

手順

1. 移動先 システム セキュリティ > セキュリティ属性.
2. セキュリティ属性リストで、[新規] を選択します。
3. セキュリティ属性フィールドに情報を記載します。

セキュリティ属性フィールド

フィールド	説明
ラベル	セキュリティ属性のラベル
名前	セキュリティ属性のラベル
タイプ	セキュリティ属性のタイプ ○ 複合 注: 複合セキュリティ属性の詳細については、「複合セキュリティ属性」を参照してください。 ○ 整数 ○ list ○ 文字列 ○ ブール (true | false)
動的 (Is dynamic)	それぞれでセキュリティ属性値を再評価する必要があるかどうか
説明	ユーザーが作成したセキュリティ属性についての説明
アプリケーション	静的フィールド、アプリケーションスコープ。
ルックアップ テーブル	評価のために外部テーブルを参照します。
ルックアップテーブル列	評価のためにテーブル列を参照します。
スクリプト	スクリプトから値を導出します。

OOB (Out-of-Box) セキュリティ属性

一般的に使用され、一般化されたすぐに使用できるセキュリティ属性ロール。

概要

OOB (Out-of-Box) セキュリティ属性は、一連の事前設定されたセキュリティ属性ロールを使用して、セキュリティ属性の機能の使用と学習を開始する簡単な方法です。OOB セキュリティ属性ロールは、一般的に使用されるアクセス制御ロールです。

独自のセキュリティ属性を作成するか、OOB セキュリティ属性機能を拡張するには、「複合セキュリティ属性」を参照してください。

OOB セキュリティ属性

属性	説明
グループ	ユーザーは指定されたグループのメンバー
GroupExplicit	ユーザーは特定のグループの明示的なメンバー
hasAdminRole	ユーザーにアドミンペルソナがある
代理操作ユーザー	ユーザーの代理操作
InteractiveSession	現在のセッションがインタラクティブ
Loggedin	ユーザーはログインおよび認証済み
NetworkCriteria	追加のネットワーククライテリア
ロール	ユーザーに特定のロールがある
RoleExplicit	ユーザーに特定のロールが明示的に定義されている

クライアントセッションのセキュリティ属性

以下のセキュリティ属性がクライアントセッション (プラグイン：com.glide.client_session_security_attributes) に追加されます。

- **IsIframeEmbeddedSession**：この属性は、サードパーティポータルの埋め込み iframe で使用されます。たとえば、エンゲージメントメッセンジャー、仮想エージェント Web クライアント、埋め込みセッションなどです。
- **IsIntegrationAsAServiceSession**：この属性は、Teams の仮想エージェント、Slack の仮想エージェント、WhatsApp の仮想エージェントなどのメッセージングアプリで使用されます。
- **IsIntegrationAsAUserSession**：この属性は、統合アカウントユーザーまたは Web サービスユーザーで使用されます。
- **ServiceNow Web** セッションである:この属性は Web インタラクティブセッションで使用されます。
- **Is Mobile App Session**：この属性は、**oauth_entity** レコードの **mobile_client** プロパティが true の場合に使用されます。

注:
IsIframeEmbeddedSession、**IsIntegrationAsAServiceSession**、**IsIntegrationAsAUserSession** は、OAuth レコードと SSO レコードのクライアントタイプを設定するときにのみ使用する必要があります。詳細については、「OAuth および SSO レコードのクライアントタイプの構成」を参照してください。

非明示的および明示的な動作が説明されている

セキュリティ属性は、ロール権限の明示的な評価と明示的でない (継承された) 評価により、微妙な権限のニーズに対応します。

複合セキュリティ属性

複合セキュリティ属性を使用すると、ビジネスニーズに合った一貫性のある再利用可能なセキュリティ属性プロファイルを作成できます

概要

複合セキュリティ属性は、1 つ以上の既存のセキュリティ属性から定義され、権限評価用のセキュリティ属性の単一の参照組み合わせを作成します。

複合セキュリティ属性の作成

複合セキュリティ属性を作成し、簡単に再利用できるようにします。

始める前に
必要なロール：security_admin

手順

1. 移動先 システムセキュリティ > セキュリティ属性.
2. セキュリティ属性リストで、[新規] を選択します。
3. [Type (タイプ)] フィールドで [複合] を選択します。
4. セキュリティ属性フィールドに情報を記載します。

セキュリティ属性フィールド

フィールド	説明
ラベル	セキュリティ属性のラベル
名前	セキュリティ属性のラベル
説明	ユーザーが作成したセキュリティ属性についての説明
条件	セキュリティ属性評価を定義する目的で使用される特定のセキュリティ属性条件
新しい条件	評価用に **OR** 条件のセットを追加します。

セキュリティ属性のスコープ

セキュリティ属性は、スコーピング機能をサポートしています。

セキュリティ属性のスコーピング

セキュリティ属性のスコープ付き動作は、プラットフォームのスコーピング動作と一致しています。スコープ内で作成されたセキュリティ属性は、同じスコープ内のアクセス制御でのみ使用できます。

ローカルおよび既存のセキュリティ属性

既存およびローカルのセキュリティ属性を使用すると、顧客はセキュリティ属性の条件セットを再利用できます。

ローカルおよび既存の属性

ACL セキュリティ属性条件ビルダーを使用すると、セキュリティ属性が既存かローカルかを指定できます。

注: セキュリティ属性条件のデフォルトはローカルです。

ローカルに定義されたセキュリティ属性は、それが作成された単一の ACL にのみ保存されます。

既存のオプションを使用すると、ユーザーは既存のセキュリティ属性条件を条件ビルダーで参照できます。

フィールドクエリロールとフィールドクエリ制限

[フィールドクエリロール **(Field Query Roles)]** 属性と [フィールドクエリ制御 **(Field Query Restrictions)]** 属性を使用すると、ユーザーがテーブルで使用できる情報をより適切に制御できます。

概要

[フィールドクエリロール **(Field Query Roles)]** 属性と [フィールドクエリ制限 **(Field Query Restrictions)]** 属性を使用すると、ユーザーがテーブルに対して実行できるアクションをより詳細に制御できます。

フィールドクエリロール **(Field Query Roles)**

ユーザーロールに基づいて、機密フィールドのクエリを防止します。

フィールドクエリ制限 **(Field Query Restrictions)**

該当するフィールドへのアクセスが拒否された場合に、レコードへのアクセスを制限します。

フィールドクエリロールの構成

[フィールドクエリロール (Field Query Roles)] 属性を有効にして構成する方法について説明します。

始める前に

必要なロール：admin

手順

1. 移動先 すべて > システム定義 > テーブル.
2. テーブルを選択します。
3. [列ラベル] を選択して、その [辞書] エントリを表示します。
4. [属性] タブを選択します。
5. ヒント: [フィールドクエリロール **(Field Query Roles)]** フィールドがない場合は、[新規] を選択して作成します。

   [フィールドクエリロール **(Field Query Roles)]** 属性を選択します。
6. このテーブル列を表示するために必要なロールを [値] に入力します
7. [更新] を選択します。

結果

テーブル列をクエリできるのは、[フィールドクエリロール **(Field Query Roles)]** 属性で定義されたロールだけです。定義されたロールのないユーザーがテーブル列をクエリすると、その操作に対するアクセスが拒否されます。

フィールドクエリ制限の構成

フィールドクエリ制限を構成する方法について説明します。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > システム定義 > テーブル.
2. テーブルを選択します。
3. [列ラベル] を選択して、その [辞書] エントリを表示します。
4. [属性] タブを選択します。
5. ヒント: [フィールドクエリレコードアクセス制限 **(Field Query Restrict Record Access)**] フィールドがない場合は、[新規] を選択して作成します

   [フィールドクエリレコードアクセス制限 **(Field Query Restrict Record Access)**] 属性を選択します。
6. レコードアクセスを、フィールドを表示できるユーザーのみに制限するには、[値] フィールドに `true` と入力し、それ以外の場合は `false` と入力します。
7. [更新] を選択します。

結果
テーブル列は、あらゆるクエリの結果を制限します。制限のあるユーザーには、クエリが返した結果の件数が通知されますが、それ以上の情報は知らされません。

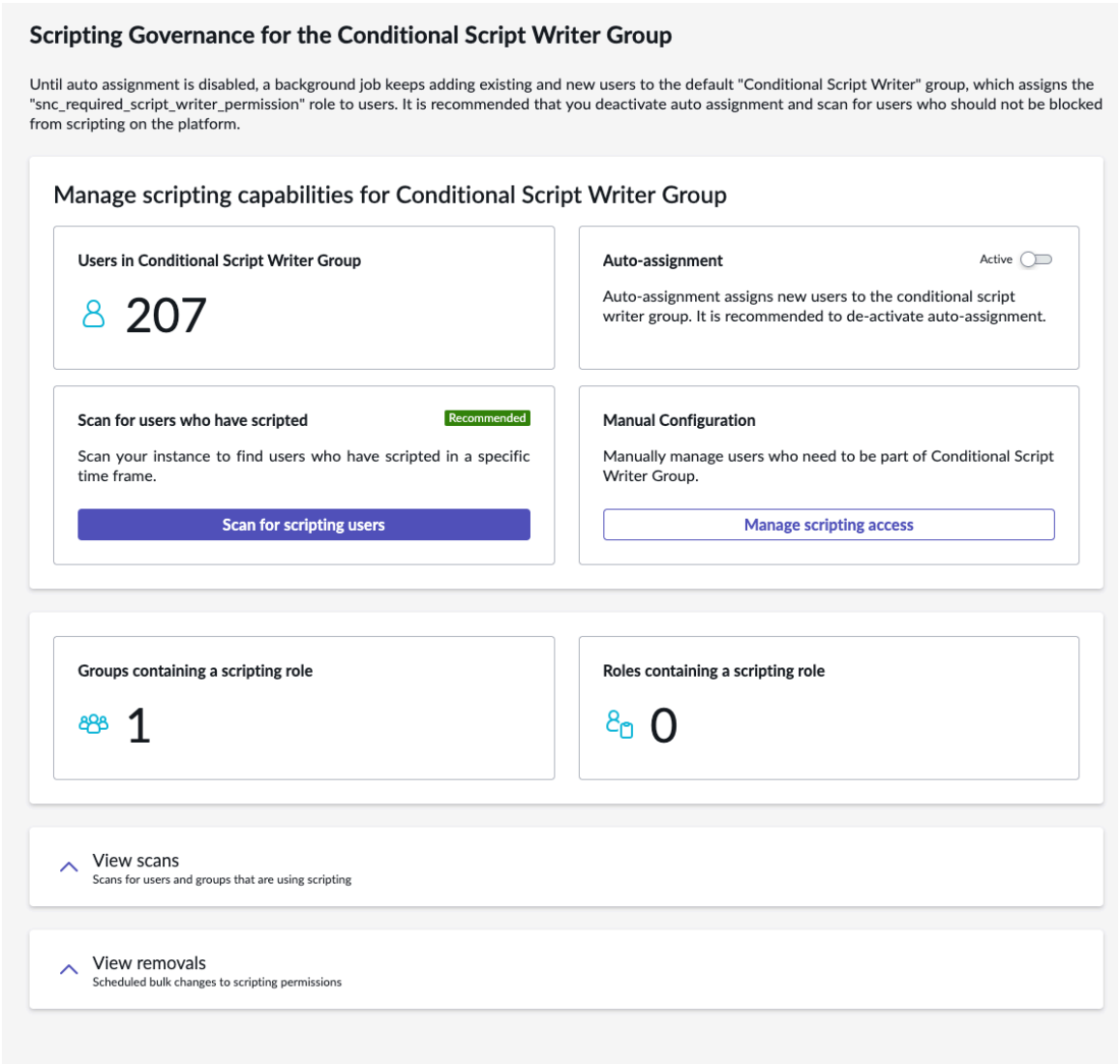
スクリプティングガバナンスツール

スクリプティングガバナンスツールから、プラットフォーム上のスクリプティングのユーザー権限を管理します。

スクリプティングガバナンスツールは、ユーザースクリプティングガバナンスメトリクスを確認し、条件付きスクリプトライターグループを管理するのに役立ちます。このグループは、snc_required_script_writer_permission ロールを介してスクリプティング権限をメンバーに付与します。ユーザーは、自動アサインまたは手動構成のいずれかによってグループに追加されます。これらの設定は両方とも **Scripting Governance Tool** から管理できます。

Scripting Governance ツールにアクセスするには、次の場所に移動します。 すべて > システムセキュリティ > スクリプティングガバナンスツール.

スクリプティングガバナンスツールダッシュボード



スクリプティングガバナンスツールの概要

条件付きスクリプトライターグループのユーザー

条件付きスクリプトライターグループのライターの数を表示します。これらのユーザーには、グループから削除されるまでsnc_required_script_writer_permissionが付与されます。

自動アサイン

自動割り当てによって新しいユーザーが条件付きスクリプトライターグループにアサインされます。自動割り当てを有効または無効にするには、スライダーを選択します

スクリプトを作成したユーザーのスキャン

最近の期間にスクリプトを作成した条件付きスクリプトライターグループ内のユーザーをスキャンします。[スクリプティングユーザーのスキャン] ボタンを選択し、短いフォームに入力してスキャンをスケジュールします。

手動構成

条件付きスクリプトライターグループにとどまるユーザーを手動で選択できます。[スクリプトアクセスを管理] ボタンを選択し、テキストフィールドにユーザーを入力します。

> **注:** 条件付きスクリプトライターグループの一員として選択されていないユーザーは、グループからの削除がスケジュールされます。

スクリプティングロールを含むグループ

スクリプティングロールを含むグループの数を表示します。デフォルトでは、条件付きスクリプトライターグループには常にスクリプティングロールがあります。

スクリプティングロールを含むロール

スクリプティングロールを含むロールの数を表示します

スキャンを表示

スクリプティングを使用しているユーザーとグループを含む展開可能なテーブル。[スクリプティングユーザーをスキャンする] ボタンによって設定されます。

スクリプティングスキャンテーブルのフィールド

ラベル	説明
スキャン番号	スキャンジョブの番号 ID。
ステータス	スキャンのステータス
完了	スキャンの完了ステータス
予約済み	スキャンの予定実行時間
作成者	スキャンを開始したユーザー
アクション	スキャンが実行したアクション

削除を表示

スクリプティング権限に対するスケジュールされた一括変更を含む展開可能なテーブル。これは、[スクリプティングユーザーのスキャン **(Scan for scripting users**)] ウィジェット、または 手動構成を使用して入力されます。

削除テーブルのフィールド

ラベル	説明
削除ジョブ番号	削除ジョブの番号 ID。
ステータス	削除ジョブのステータス
完了	削除ジョブの完了ステータス
予約済み	削除ジョブの実行時間をスケジュール
作成者	削除ジョブを開始したユーザー

マシン **ID** アクセス制御

統合ユーザーの特定のリソースに対して詳細なアクセス制御ポリシーを定義して適用します。

マシン ID アクセス制御を使用すると、アドミニストレーターは統合ユーザーに対して詳細なアクセス制御ポリシーを作成できます。この追加のセキュリティレイヤーにより、アドミニストレーターは統合ユーザーがアクセスできる正確なリソース (REST API、SOAP API、およびテーブル) を指定できるため、より厳格なアクセスガバナンスを確保できます。

> **注:** Web サービスアクセスのフラグが付けられたユーザーのみに適用されます。

マシン ID アクセス制御を作成する方法については、「 マシン ID アクセス制御を作成する 」を参照してください

マシン ID アクセス制御を作成する

アドミニストレーターは、ユーザーアクセスプロファイルを導入することで、統合ユーザーのきめ細かな制御を定義して適用できます。この機能により、セキュリティと制御の追加レイヤーが提供されるため、アドミニストレーターは統合ユーザーがアクセスできる正確なリソース (REST API および SOAP API) を指定できるため、より厳格なガバナンスが確保され、セキュリティリスクが最小限に抑えられます。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > システムセキュリティ > マシン **ID** アクセス制御.
2. **[New** (新規)] ボタンをクリックします。
3. フォームのフィールドに入力します。

マシン **ID** アクセス制御のフィールド

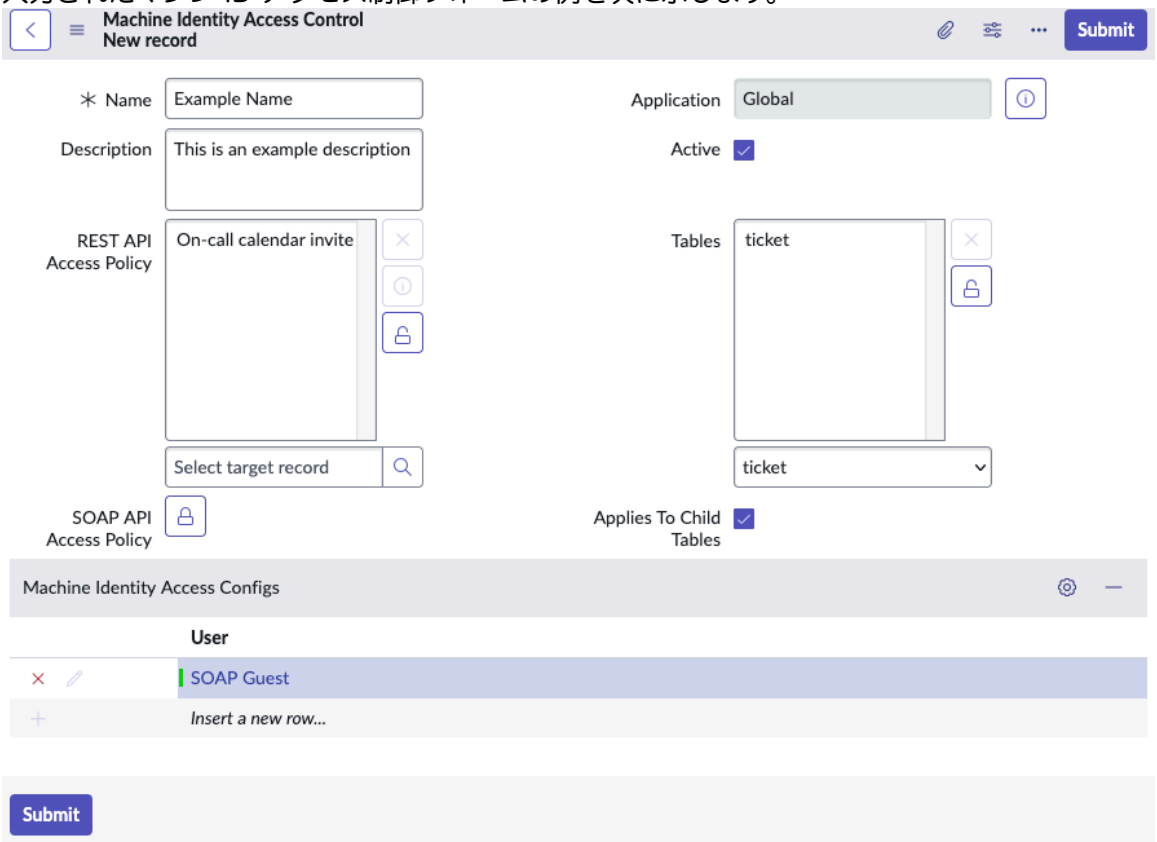
フィールド	説明
名前	アクセス制御レコードの名前。
アプリケーション	このレコードを含んでいるアプリケーション。
説明	レコードの説明です。
アクティブ	ポリシーがアクティブかどうかを決定します
REST API ポリシー	ターゲット REST API ポリシーを選択します。 注:] アイコンと] アイコンを選択して、ポリシーを追加します。
SOAP API ポリシー	ターゲット SOAP API ポリシーを選択します。 注:] アイコンと] アイコンを選択して、ポリシーを追加します。
テーブル	このポリシーが適用されるテーブルを選択
子テーブルに適用	これをオンにすると、[テーブル] フィールドの子テーブルにポリシーが適用されます

4. [下に行を挿入] プロンプトを選択し、コントロールを適用するユーザーを追加します。
アクセス制御に複数のユーザーを追加できます。

   注: Web サービスへのアクセス権を持つユーザーのみを選択できます。

5. **[Submit** (送信)] を選択します。

結果
入力されたマシン ID アクセス制御フォームの例を次に示します。



マシン ID アクセス制御を持つユーザーは、他の API (REST または SOAP) にアクセスできず、必要なロールを持っている場合でも、アクセス制御に明示的に指定されたリソースにのみアクセスできます。

データフィルタリング

読み込みクエリを実行するときに、データフィルタリングを使用し、対象の属性に基づいてテーブルとレコードへのアクセスを制御します。

重要:

Yokohama リリース以降、データフィルタリングは将来の廃止に向けて準備されています。これは非表示になり、新しいインスタンスではアクティブ化されなくなります。セキュリティデータフィルターを使用して、ロールまたはセキュリティ属性関連のアサーションに基づいてレコードへのアクセスを制限することを検討してください。詳細については、「セキュリティデータフィルターの作成」を参照してください。

データフィルタリングの詳細

データフィルタリングについて学習します。

データフィルタリングルールの作成

ニーズに合わせて独自のデータフィルタリングルールを作成します。

デバッグデータ

データフィルタリングの結果をデバッグする方法について説明します。

自動翻訳

データフィルタリングの詳細

読み込みクエリを実行するときに、Data Filtration (データフィルタリング) を使用し、対象の属性に基づいてテーブルとレコードへのアクセスを制御します。

データフィルタリングは、インスタンスの既存のアクセス制御ルール (ACL) と連携して機能するように設計された、独立した形式のアクセス制御です。データフィルタリングは、アドミニストレーターによって定義された対象属性と一致しないテーブルおよびレコードへのアクセスを拒否します。データフィルタリングは、監査、レポート、およびトラブルシューティングを簡易に行えるように設計されています。

これは、アドミニストレーターがインスタンスで有効にできるオプションの機能です。

データフィルタリング機能

データフィルター

データフィルターを使用し、レコード内の情報に基づいてアクセス権を付与します。データフィルターは、テーブルフィールドのデータを使用して、ユーザーがレコードを使用できるかどうかを判断します。

対象属性ベースの条件ビルダー

対象属性を使用して、ユーザーロール、グループ、対象基準、または IP ネットワークアドレスを評価します。

データフィルタリングで拒否ベースのモデルを使用する

データフィルタリングでは、拒否ベースのモデルを使用してレコードへのアクセスを制御します。データフィルタリングでは、レコードがデータフィルタリングで定義された基準を満たしていない限り、インスタンスはレコードへのアクセスを拒否します。

データフィルタリングの適用

データフィルタリングルールは、読み取り操作のデータベースクエリの後に実行され、ACL の前に評価されます。データフィルタリングルールによって拒否されたレコードは処理されず、ACL ルールによって評価されます。データフィルタリングルールの適用は、読み取り ACL の適用と一致しています。

データのフィルタリングとレポート

データフィルタリングと ACL はどちらも、リストビューレポートの作成時にのみ適用されます。レポートでは、集計データを収集するときにアクセス制御は適用されません。この場合、データフィルタリングも ACL もチェックされません。

集計されたレポートの場合、データフィルタリングは既存の *Report_view access control list* 動作と連携して機能します。これらのレポートコントロールの構成の詳細については、「Report_view アクセス制御」を参照してください。

セッションのデバッグ

データフィルタリングはセッションのデバッグをサポートします。セッションデバッグを使用して、特定のクエリーに適用されるデータフィルタリングレコードを確認します。アドミニストレーターは、この情報を使用してレコードへのユーザーアクセスのトラブルシューティングを行うことができます。

データフィルタリングのコンポーネント

データフィルタリングは、次のレコードタイプを使用して機能します。

データフィルタリングレコード

データフィルタリング [sys_df_data_filtration] レコードを作成して、インスタンスのテーブルアクセスを許可します。データフィルタリングレコードには、ルールのスコープと影響を受けるユーザーを制限するための、前述の データフィルター と 対象属性 の条件が含まれています。

対象基準レコード

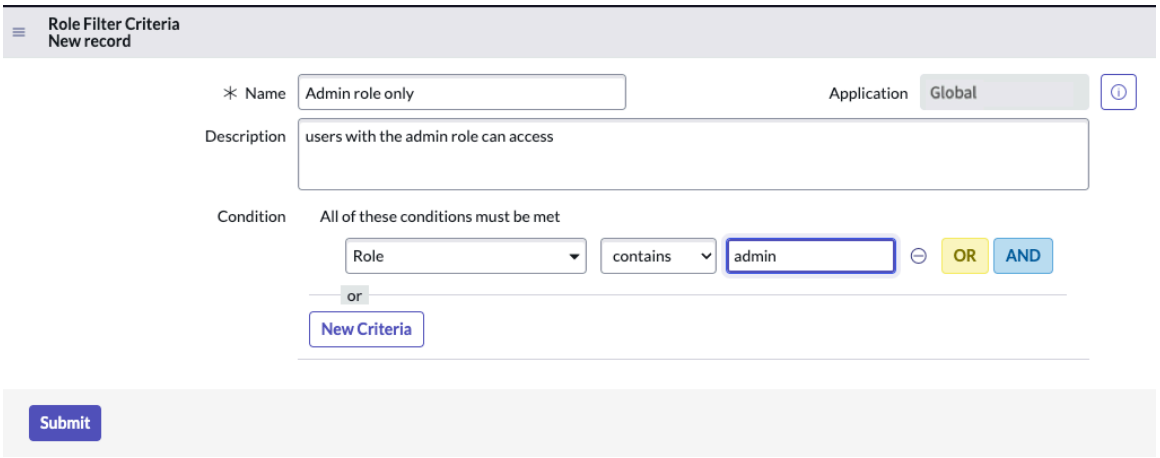
対象基準 [sys_df_subject_criteria] レコードは、データフィルタリングルールを使用してアクセスを許可するかどうかを決定するために使用できる特定のユーザー属性を表

します。これらの属性は、ユーザーのグループ、ロール、または IP アドレスです。対象基準を作成するには、対象基準レコードと基準入力および基準条件レコードを作成する必要があります。このプロセスの詳細については、「対象基準の作成」を参照してください。

対象基準レコードを作成したら、それをルールに適用できます。これは、データフィルタリングルールの [対象条件] タブで行います。

基準入力レコードの例

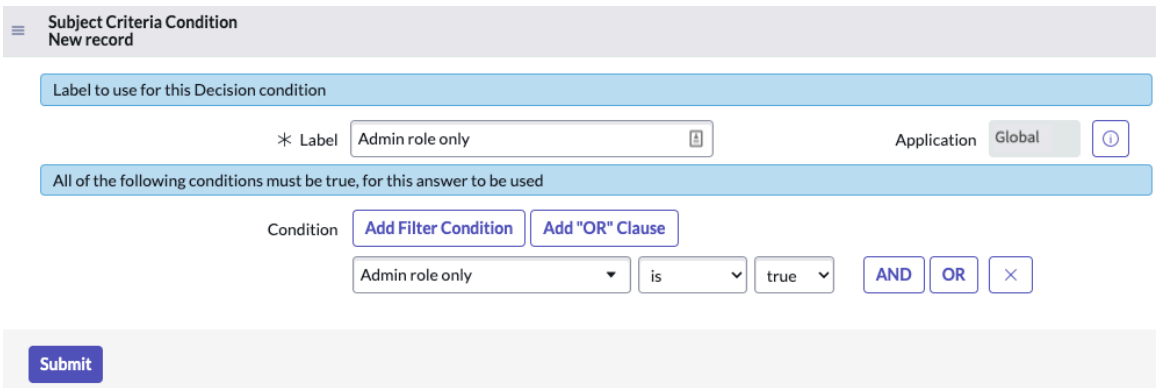
admin を含むすべてのロールの基準入力の例



基準入力 [sys_df_subject_filter_criteria_m2m] は、ユーザーと比較する基準を含むレコードです。これは、ユーザーグループまたはロール、IP アドレス範囲、または IP アドレスサブネットのリストにすることができます。これらのレコードは、対象基準条件レコードとともに使用され、ユーザーのグループ、ロール、または IP アドレスに対して評価されて、テーブルまたはそのレコードへのアクセスが決定されます。

対象基準の条件レコード

アドミンのみの基準入力を使用した基準条件の例



対象基準条件 [sys_df_subject_criteria_condition] レコードを使用して、ユーザー属性を、基準入力で定義されたロール、グループ、または IP アドレスと比較する方法を定義します。単一の対象基準条件で複数の基準入力を使用して、レコードへのアクセスをさらに絞り込むことができます。

データフィルタリングを有効にする

インスタンスでデータフィルタリングを有効にする方法について説明します。

始める前に
必要なロール：security_admin

手順

1. 移動先 すべて > システム定義 > プラグイン.
2. 検索フィールドを使用して、Data Filtration (com.glide.data_filtration) プラグインを検索します。
3. [インストール] をクリックして、[プラグインをアクティブにする] ダイアログボックスで、[アクティブ化] をクリックします。

データフィルタリングルールの作成

レコードへのアクセスをユーザーに許可するデータフィルタリングルールの作成方法について説明します。

始める前に
必要なロール：security_admin

注: データフィルタリングルールを作成または変更するには、 特権ロールに昇格される必要があります。このプロセスの詳細については、「特権ロールへの昇格」を参照してください。

手順

1. 移動先 すべて > データフィルタリング > データフィルタリングレコード.
2. [データフィルタリング] リストで [新規] をクリックします。
   新しいデータフィルタリングフォームが表示されます。
3. 必要に応じて、フォームのフィールドに入力します。

データフィルタリングフォーム

フィールド	説明
テーブル	このデータフィルタリングルールが適用されるテーブル。 注: メンテナンス以外のユーザーは、一部のテーブルでデータフィルタリングを作成できません。これを回避するには、tableChoicesScript=DataFiltrationTableList 属性を削除しますが、sys_df_xxxテーブルまたはsys_df_table_exclusion内のテーブルにフィルターが作成されないようにします。
アクティブ	データフィルタリングルールをアクティブに設定します。 注: 意図せずにユーザーをレコードからロックアウトしないように、テストの準備ができるまでデータフィルタリングルールを無効にします。
説明	データフィルタリングルールの説明。
カスケード	選択して、拡張テーブルに適用されるようにデータフィルタリングルールを設定します。 たとえば、Task[task] テーブルを選択し、カスケードを有効にします。この場合、データフィルタリングルールは、インシデント [incident] や変更要求 [change_request] などのタスクから拡張されたすべてのテーブルにも適用さ

フィールド	説明
	れます。テーブル拡張の詳細については、「テーブル拡張とクラス」を参照してください。 注: このフィールドはデフォルトで有効になっています。

4. オプション: ルールの適用範囲を絞り込むには、必要に応じて [条件] フィールドに入力します。

フィールド	説明
対象条件	アクセスするには、すべての条件を満たす必要があります。
セキュリティ属性条件	アクセスするには、すべての条件を満たす必要があります。 ローカル この属性は、データフィルタリングルールの範囲内でのみ定義されます。 既存の 属性は、既存のセキュリティ属性への参照によって定義されます
データ条件	データがルールの対象となる条件を定義します。 注: 空のデータ条件は、選択したテーブルのすべてのレコードに適用されます。

5. フォームメニューで、[保存] を選択します。
データフィルタリングルールを保存すると、データ条件で特に指定されていない限り、このルールは選択したテーブルのすべてのレコードに自動的に適用されます。

データフィルタリングルールのデータフィルターの追加

必要に応じて、データフィルターを使用して、データフィルタリングルールのスコープをテーブルの特定のレコードにのみ適用されるように絞り込むことができます。

始める前に

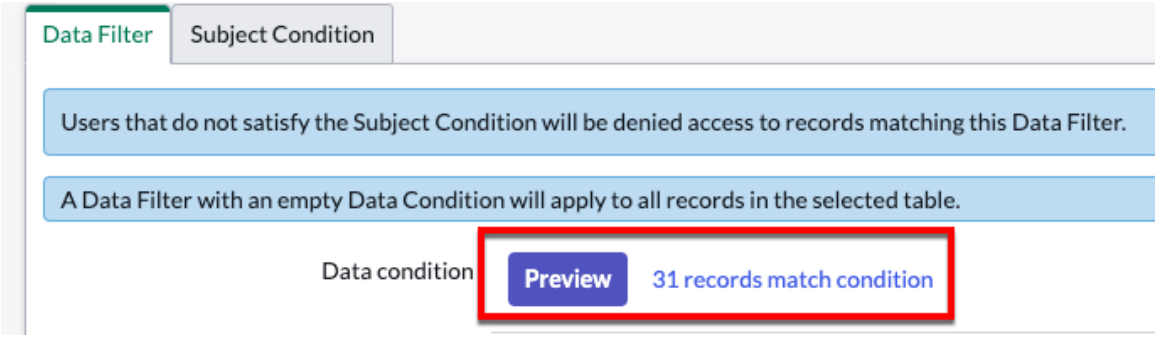
必要なロール：admin

手順

1. データフィルタリングレコードで、[データフィルター] タブを開きます。
2. 条件ビルダーを使用して、テーブルレコードのフィールド値をフィルタリングします。
データフィルターは、プラットフォームの他の部分で使用されているものと同じ条件ビルダーを使用します。このインターフェイスの使用の詳細については、「条件ビルダー」を参照してください。

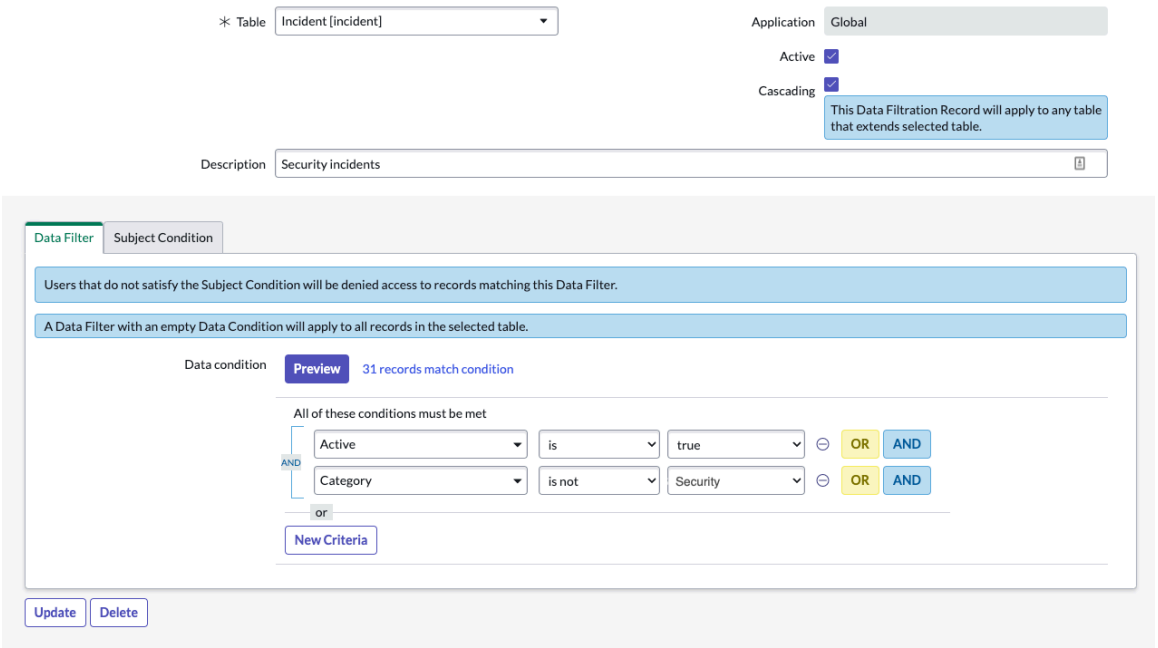
重要: [テーブル] フィールドでテーブルを選択するまで、[データフィルター] タブは空で表示されます。

3. [プレビュー] ボタンを使用して、データフィルターに一致するレコードの数を確認します。
4. レコードの数を選択すると、一致するレコードのリストが開きます。



5. [保存] を選択します。

Example:
この例は、Incident[incident] テーブルのデータフィルタリングルールを示しています。データフィルターは、セキュリティカテゴリに含まれていないすべてのアクティブなレコードを選択するように設定されています。このルールをアクティブにすると、ユーザーはこれらのレコードを表示できます。レコードのコンテンツの外部の基準を使用するには、以下のセクションを参照してください。



重要:

インスタンスで使用するデータベースのタイプによっては、条件の **not** 操作で予期しない結果が返される場合があります。たとえば、次のような条件を使用します。

この条件では、結果セットは、会社が ServiceNow ではないすべてのレコードと、**[company]** フィールドに値がないすべてのレコードになることが予想されます。MySQL と Maria 以外のデータベースを使用しているインスタンスは、**[company]** フィールドが空になっているレコードを返しません。これらのインスタンスに対して **not** クエリーを使用する場合は、空の値が返されるようにするための条件を含めます。

データフィルタリングルールに対象属性を追加する

必要に応じて、対象属性を使用し、IP ネットワークアドレス、ユーザーグループとロール、対象基準などの属性に基づいてデータフィルタリングルールのスコープを絞り込みます。

始める前に
必要なロール：admin

手順

1. データフィルタリングレコードで、[対象基準] タブを開きます。
2. 条件ビルダーを使用し、次の 1 つ以上の条件に基づいてテーブルレコードをフィルタリングします。

対象基準オプション

オプション	説明
ネットワーク基準	ネットワーク IP 範囲または IP サブネットに基づいてレコードへのアクセスを許可します。
対象基準	対象基準に基づいてレコードへのアクセスを許可します。対象基準レコードを選択して、その条件をデータフィルタリングルールに適用します。対象基準レコードの作成の詳細については、「対象基準の作成」を参照してください。
対象グループ	ユーザーが特定のグループのメンバーである場合にアクセスを許可します。**Groups**[sys_user_group] テーブルからグループを選択します。
対象ロール	ユーザーが特定のグループのメンバーである場合にアクセスを許可します。**Roles**[sys_user_role] テーブルからグループを選択します。

重要: 対象基準条件は **is** 演算子のみをサポートします。

3. 対象基準を追加した後に、[保存] をクリックします。

対象基準の作成

データフィルタリングルールで使用するための対象基準レコードを作成します。

始める前に
必要なロール：security_admin

重要: データフィルタリングルールを作成または変更するには、 特権ロールに昇格される必要があります。このプロセスの詳細については、「特権ロールへの昇格」を参照してください。

手順

1. 移動先 すべて > データフィルタリング > 対象基準.
2. [対象基準] リストで [新規] ボタンをクリックします。
   新しい対象基準フォームが表示されます。
3. 必要に応じて、フォームのフィールドに入力します。

対象基準のフィールド

フィールド	説明
名前	対象基準の名前。
アプリケーション	対象基準のスコープ対象のアプリケーション。このフィールドは、読み取り専用であり、現在のスコープ対象のアプリケーションが自動的に入力されます。
説明	対象基準の説明。

4. フォームヘッダーを右クリックし、コンテキストメニューから [保存] をクリックします。
保存した後に、[基準の入力] フィールドと [基準の条件] フィールドが表示されます。

対象基準入力の作成

対象基準の入力を作成して、データフィルタリングルールでフィルタリングする基準を定義します。

始める前に
必要なロール：security_admin

手順

1. 対象基準レコードで、[基準の入力] タブを開きます。
2. [基準の入力] リストで [新規] をクリックします。
3. 作成する基準のポリシーの入力を選択します。

ポリシーの入力

ポリシーの入力	説明
IP フィルター基準	IP アドレスに基づいてポリシー入力を作成します。
ロールフィルター基準	使用ロールに基づいてポリシーの入力を作成します。
グループフィルター基準	ユーザーグループに基づいてポリシーの入力を作成します。

このステップでの選択に応じて、**[IP** フィルター基準]、[ロールフィルター基準]、または [グループフィルター基準] フォームが表示されます。

4. 必要に応じて、フィルター基準フォームのフィールドに入力します。

フィルター基準のフィールド

フィールド	説明
名前	フィルター基準の名前
アプリケーション	対象基準のスコープ対象のアプリケーション。このフィールドは、読み取り専用であり、現在のスコープ対象のアプリケーションが自動的に入力されます。
説明	フィルター基準の説明

5. フォームフィールドの下には、フィルター基準入力の IP アドレス、グループ、またはロールを定義するために使用されるタブがあります。

特定の入力タイプのフィルター基準フィールド

ポリシーの入力のタイプ	説明	作成
IP フィルター基準	IP アドレスの範囲またはサブネットを作成するためのユーザー IP フィルター基準。対象基準は、この範囲またはサブネットに対してユーザーの IP アドレスと比較できます。	**IP** 範囲またはサブネット **(CIDR)** を使用して、入力の IP アドレスを定義します。 IP 範囲 **[IP** 範囲] リストで [新規行を挿入] をダブルクリックし、[開始 **IP]** に開始 IP アドレスを入力します。次に、Tab キーを押して、[終了 **IP]** フィールドに終了 IP アドレスを入力します。最後に、Enter キーを押してリストエントリーを保存します。 サブネット (CIDR) [サブネット] リストで、[新規行を挿入] をダブルクリックし、[ネットワーク **IP]** フィールドにネットワーク IP を入力します。次に、Tab キーを押して、[ネットマスク] フィールドにネットマスクを入力します。最後に、 Enter キーを押してリストエントリーを保存します。 **注:** スケジューラーによってトリガーされるスケジュール済みジョブは、要求元のクライアント IP のコンテキストを持たないため、ネットワーク基準を使用してデータをフィルタリングすることを意図していません。ロール/グループ対象条件のタイプのフィルタリングの方が適している場合があります。
ロールフィルター基準	ロールフィルター基準を使用して、ロールの選択を作成します。その後で対象基準は、この選択に対してユーザーのアサイン済みロールと比較できます。	[条件] フィールドの条件ビルダーを使用して、入力のロールを選択します。

ポリシーの入力のタイプ	説明	作成
グループフィルター基準	グループフィルター基準を使用して、ユーザーグループの選択を作成します。その後で対象基準は、この選択に対してユーザーのアサイン先グループと比較できます。	[基準のグループ] テーブルで、[新規行を挿入] をダブルクリックし、ユーザーグループを選択します。Enter を押すか、緑色のチェックマークアイコンをクリックしてグループを保存します。 最初のエントリーの下の [新規行を挿入] テキストをクリックして、追加のエントリーを作成します。

6. 入力を定義したら、[送信] をクリックします。

> **注:** 対象基準の基準入力を作成するだけでなく、既存の基準入力を追加することもできます。[基準の入力] タブで [編集] をクリックし、既存の入力から選択します。

対象基準条件の作成

ユーザー情報を既存の対象基準入力と比較するための条件を作成します。

始める前に

必要なロール：security_admin

基準条件は、ユーザー属性を既存の基準入力と比較して、レコードへのアクセスを許可するかどうかを決定します。基準条件を作成するには、対象基準を作成している必要があります。このプロセスの詳細については、「対象基準入力の作成」を参照してください。

手順

1. 対象基準レコードで、[基準の条件] タブを開きます。
2. [基準の条件] リストで [新規] をクリックします。
3. 必要に応じて、[対象基準の条件] フォームのフィールドに入力します。

[対象基準の条件] フォーム

フィールド	説明
ラベル	条件を説明するラベル
アプリケーション	対象基準のスコープ対象のアプリケーション。このフィールドは、読み取り専用であり、現在のスコープ対象のアプリケーションが自動的に入力されます。

4. 条件ビルダーを使用し、次の条件オプションから選択して、対象基準の条件を作成します。これらの条件オプションには、作成した任意の対象基準の入力が含まれます。
5. オプション: [フィルター条件の追加] または [「OR」節を追加] ボタンをクリックして、さらに条件を作成します。

> 注: 条件が **or** 節で区切られていない限り、対象基準の条件が true と評価されるには、すべての条件が true と評価される必要があります。

6. [送信] をクリックして、対象基準の条件を保存します。

次のタスク
データフィルタリングルールで対象基準を使用して、テーブルとレコードへのアクセスを制限します。データフィルタリングルールでの対象基準の使用方法の詳細については、「対象基準の作成」を参照してください。

データフィルタリングのデバッグ

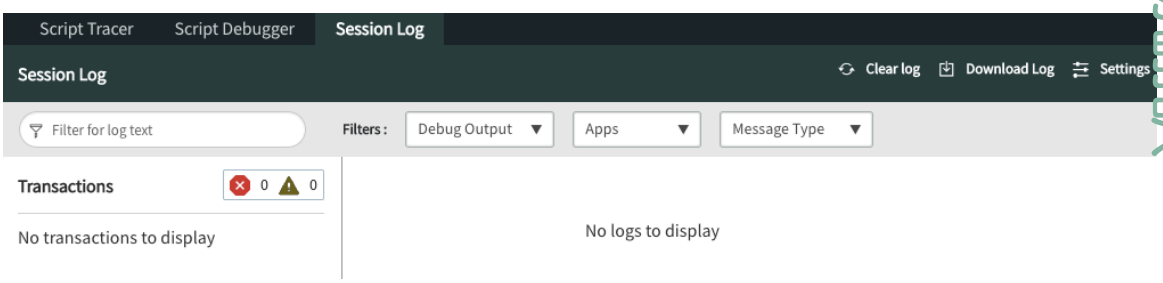
セッションログを使用して、データフィルタリングがレコードにどのように影響するかを確認し、ユーザーアクセスの問題をデバッグします。

始める前に
必要なロール：admin

ユーザーがレコードにアクセスすると、出力情報がセッションログに表示されます。このログ記録情報を代理操作とともに使用して、ユーザーにレコードが表示されるかまたは表示されない理由を調べることができます。次にその情報を使用してデータフィルタリングルールを調整し、意図したレコードのみがユーザーに表示されるようにすることができます。

手順

1. 移動先 すべて > システムセキュリティ > デバッグ > すべてのセキュリティをデバッグ.
   スクリプトデバッガーが新しいブラウザータブまたはウィンドウで開きます。



2. [スクリプトデバッガー] ウィンドウで、[セッションログ] タブを選択します。
3. 別のブラウザータブまたはウィンドウで、ユーザーの代理操作を行ってそのユーザーのアクセスをトラブルシューティングします。

> 注: 代理操作により、アドミニストレーターは別のユーザー設定とアクセス権を持つインスタンスを表示できます。代理操作の詳細については、「ユーザーの代理操作を行う」を参照してください。

4. ユーザーの代理操作中に、予期しない動作が発生したリストまたはレコードにアクセスします。
   この動作は、表示されるべきではないにもかかわらずユーザーに表示されるレコード、またはユーザーに期待どおりに表示されないレコードのリストです。
   代理操作されたユーザーでレコードにアクセスすると、セッションデバッガーで出力が表示されるようになります。
5. セッションデバッガーでデータフィルタリング情報を探します。

この例は、データフィルターがレコードへのアクセスを拒否した 2 つのログメッセージを示しています。ログエントリーは赤色のテキストとして表示され、データフィルターがアクセスを拒否した理由と、データフィルターの sys_id が含まれます。この sys_id をクリックすると、データフィルタリングレコードを開くことができます。

この例は、データフィルターがレコードへのアクセスを許可したログメッセージを示しています。これらのログエントリーは緑色のテキストとして表示されます。最初のメッセージと同じように、この sys_id をクリックすると、データフィルタリングレコードを開くことができます。

6. この情報を使用して、データフィルタリングルールを調整します。
これらの手順を繰り返してルールを調整し、ユーザーに必要なアクセス権を付与します。

セキュリティデータフィルター

セキュリティデータフィルターは、ロールまたはセキュリティ属性関連のアサーションに基づいてレコードへのアクセスを制限します。

セキュリティデータフィルターの詳細

セキュリティデータフィルターを使用すると、ユーザーのロールやその他のセキュリティ属性関連のアサーションに基づいて、レコードへのアクセスを制限できます。セキュリティデータフィルターにより、データへのアクセス方法に関係なく、許可されたユーザーのみがレコードを表示できます。

セキュリティデータフィルターはクエリの実行前に適用されるため、制限されたデータがデータベースから出ることはありません。対照的に、 条件付き ACL は、クエリの実行後にデータをフィルタリングし、データが漏洩する可能性があります。

注: セキュリティデータフィルターと 「次の場合を除き却下 (Deny-Unless)」ACL を組み合わせて、一貫したセキュリティを確保します

セキュリティデータフィルターの機能

セキュリティデータフィルターの主な機能は次のとおりです。

- セキュリティデータフィルターはクエリ内に適用されます。
- セキュリティデータフィルター条件とターゲットテーブルのクエリに対する 条件 と相互の関係。
- セキュリティデータフィルターが canRead でチェックされません。詳細については、 セキュリティデータフィルターを使用する場合 を参照してください
- データフィルターのスコーピングルールはテーブルのスコープに基づいており、データフィルターは ScopeMaster またはsys_scopeスコープルールに従っていません

セキュリティデータフィルターの適用と適用

一般に、セキュリティデータフィルターは、絶対 ACL (テーブルレベル ACL とも呼ばれる) の後と行 ACL の後に適用されます。セキュリティデータフィルターはデフォルトで適用され、慎重に使用しないとシステムの動作に影響します。デフォルトのセキュリティデータフィルターのリストについては、 デフォルトのセキュリティフィルター を参照してください。

セキュリティデータフィルターは、デフォルトでは GlideRecordSecure、GlideRecordSandbox、および GlideAggregateSandbox クエリにのみ適用されます。Java とサーバー側の両方のスクリプトで使用できる 2 つの新しい GlideRecord API `enableSecurityFeature` と `disableSecurityFeature` を使用して、特定のクエリのデータフィルターを有効または無効にすることができます。

重要: GlideRecordSecure を使用していないユーザー向けクエリのデータフィルターを明示的に有効にする必要があります。

セキュリティデータフィルターを使用する場合

セキュリティデータフィルターは、次の場合に最適です。

- 機密データがデータベースから流出しないようにする
- 「セキュリティにより非表示にされる行」メッセージを非表示にする
- レポートを介した機密データの漏洩を防ぐ

セキュリティデータフィルターを使用しない場合

セキュリティデータフィルターは使用しないでください:

- 可視化コントロールとして
- ACL の代替として
- 多数のフィルター条件を使用する場合
- インデックスのない列

セキュリティデータフィルターの動作

`AND` でオペランドを組み合わせるように、複数のセキュリティデータフィルターを組み合わせて評価します。たとえば、次の 3 つのセキュリティデータフィルターがあるとします。

- フィルター 1:'active=true
- フィルター 2: `優先度 = 1`
- フィルター 3: `ステータス = オープン`

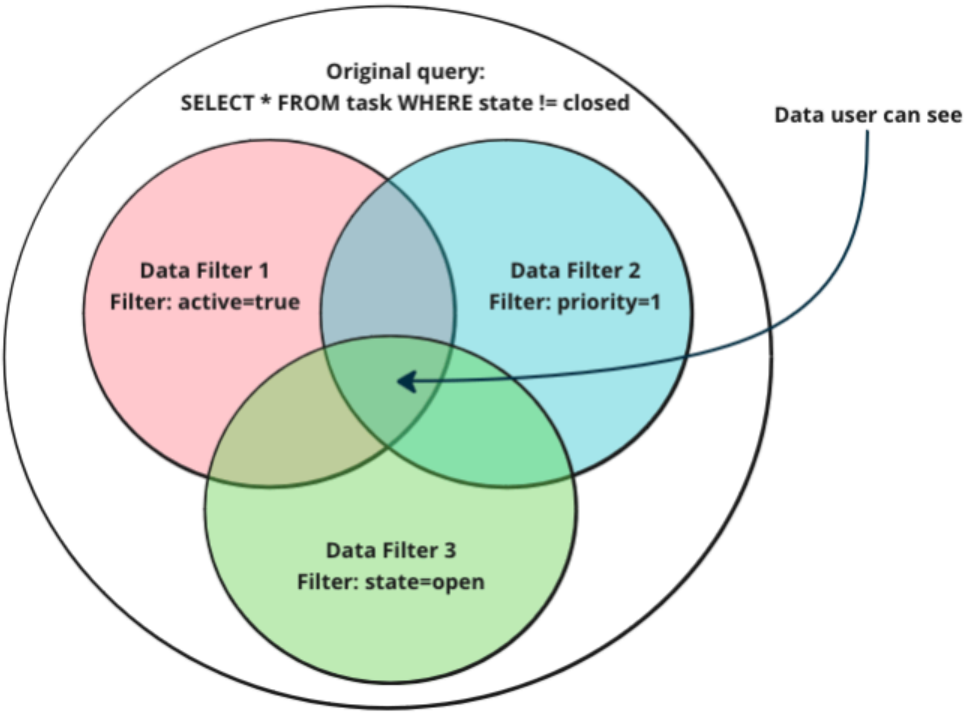
そして最初のクエリ:

```
SELECT * FROM task WHERE state != closed AND active = true AND
    priority = 1
```

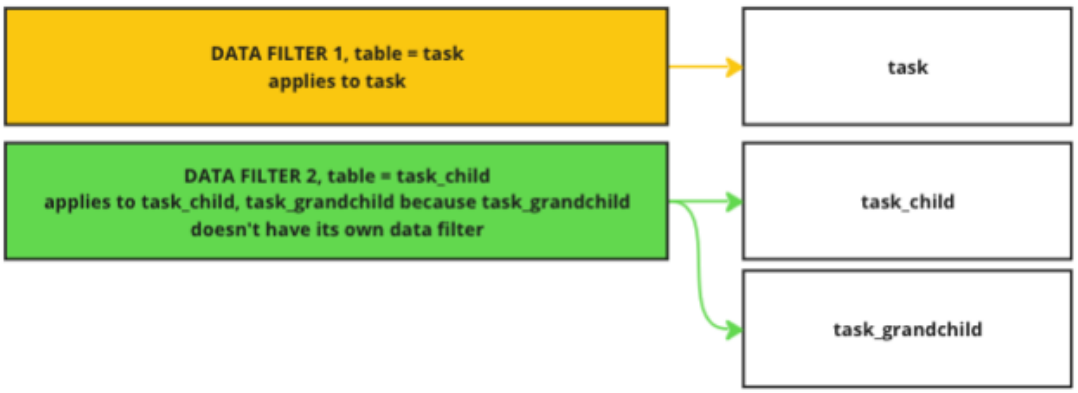
最終的なクエリは次のとおりです。

```
SELECT * FROM task WHERE state != closed
    AND active = true AND priority = 1 AND state = open
```

この例の視覚的表現については、次の図を参照してください。

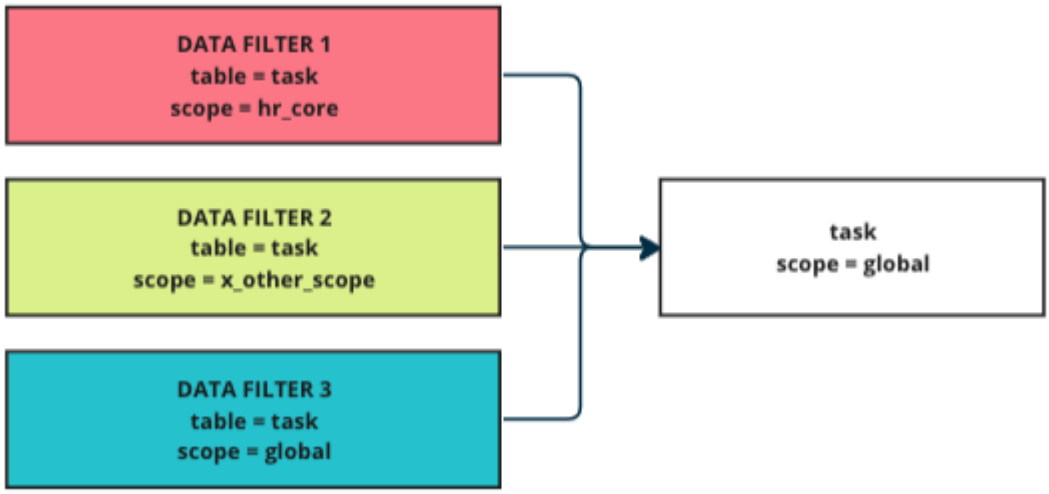


セキュリティデータフィルターと ACL の適用方法における重要な違いの 1 つは、親テーブルからデータがクエリーされる場合、子テーブルのデータフィルターは親テーブルに適用されないことです。子テーブルと親テーブルの両方にデータフィルターを追加して、親テーブルのレコードへのアクセスを制限します。次の図は、階層を示しています。



注: これに対する一般的な解決策は、親テーブルの子レコードを完全に非表示にするデータフィルターを親に追加することです。

データフィルターは ACL と同様のスコーピングルールで適用されますが、データフィルターはクエリ前に適用されるため、いくつかの重要な違いがありま



す。

セキュリティデータフィルターの作成

テーブルであるレコードへのアクセス権をユーザーに付与するセキュリティデータフィルタールールを作成する方法について説明します。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > システムセキュリティ > セキュリティデータフィルター.
2. [セキュリティデータフィルター] リストで [新規] をクリックします。
3. 必要に応じて、フォームのフィールドに入力します。

データフィルタリングフォーム

フィールド	説明
テーブル名	セキュリティデータフィルターが適用されるテーブル。
説明	セキュリティデータフィルタールールの説明。
アクティブ	セキュリティデータフィルタールールをアクティブに設定します。 注: 意図せずにユーザーをレコードからロックアウトしないように、テストの準備ができるまでセキュリティデータフィルタールールを無効にします。
UI に表示	セキュリティデータフィルターがクエリに適用される場合に、通知を UI に表示するかどうかを決定します
アプリケーション	セキュリティデータフィルターのアプリケーションスコープ。
モード	データフィルターのモード。
フィルター	データフィルターがどのレコードに適用されるかを決定するフィルター条件

フィールド	説明
セキュリティ属性	データフィルターを適用するかどうかを制御するセキュリティ属性

4. フォームメニューで、[保存] を選択します。

結果
セキュリティデータフィルタールールを保存すると、データ条件で特に指定されていない限り、このルールは選択したテーブルのすべてのレコードに自動的に適用されます。

デフォルトのセキュリティフィルター

一般に、データフィルターは、絶対 ACL (テーブルレベル ACL とも呼ばれる) の後と行レベルの ACL の後に適用されます。これらはデフォルトで適用されます。慎重に使用しないと、システムの動作に影響を与える可能性があります。

デフォルトで適用されるセキュリティデータフィルター

セキュリティデータフィルターは、デフォルトで次の場所に適用されます。

カテゴリ	アプリケーション
リストとフォーム	• UI16 • ワークスペース • サービスカタログ • サービスポータル • モバイル
レポートとダッシュボード	• レポート • データの可視化 • コア UI ダッシュボード • プラットフォームアナリティクスエクスペリエンスダッシュボード
データエクスポート	• XML のエクスポート • CSV • Excel • JSON • PDF
フローエンジン	レコードルックアップステップ

カテゴリ	アプリケーション
検索	• AI 検索 • テキスト検索
GlideRecord	• GlideRecordSecure • GlideRecordSandbox • GlideAggregateSandbox
REST および GraphQL API	• REST テーブル API • REST 統計 API • GraphQL テーブルと統計 API に相当するもの

セキュリティロール

セキュリティロールによりセキュリティを強化するためには、ユーザー全員に少なくとも 1 つのロールを割り当てて、インスタンスが内部ユーザーと外部ユーザーを区別できるようにする必要があります。

明示的なロールの詳細

明示的なロールの主な機能とビジネス価値について学びます。

昇格した権限の詳細

昇格した権限ロールでセッションベースの権限を有効にする方法について説明します。

明示的なロール

内部ユーザーと外部ユーザーの両方に、インスタンスへのアクセス権を与えることができます。ただし、両方のタイプのユーザーに同レベルのアクセス権を持たせない方が良い場合もあります。セキュリティを追加するためには、ユーザー全員に少なくとも 1 つのロールを割り当てて、インスタンスが内部ユーザーと外部ユーザーを区別できるようにする必要があります。

Parisリリース時点では、ユーザーは、明示的なロール (snc_internal および snc_external) をともに持つことはできません。グループとロールには、そのようなグループまたはロールにアサインされたグループメンバーまたはユーザーが両方のロールを自動的に持つようになるため、両方のロールを包有することはできません。ServiceNow AI Platform では、このようなシナリオを作成する操作はすべて中断されます。

注: 他のロールを含まれているロールとして「*snc_external*」または「*snc_internal*」ロールに追加することはできません。

外部ユーザーは、少なくとも snc_external ロールを取得する必要があります。snc_external ロールは、ユーザーが組織の外部にいることを示します。こうしたユーザーは、リソースへのアクセス権を持つことはできません。ただし、ACL を通じて snc_external ロールか、または snc_external ロールを継承する追加ロールに対して明示的に許可された場合は例外です。 デフォルトでは、snc_external ロールを持つユーザーは次のものにアクセスできません。

- snc_external ロールまたは公開ロールを継承するロールがないテーブル。
- snc_external ロールまたは snc_external ロールを継承するロールへのアクセスの許可がないプロセッサーや UI ページなど、レコードタイプ以外のリソース。
- プラットフォームアナリティクス ダッシュボード

snc_internal ロールを昇格したロールとしてマークしないでください。内部ユーザーがインスタンスにアクセスできなくなります。

明示的ロールプラグイン

明示的ロールプラグインが有効になっている場合：

- すべての ユーザーは、内部リソースにアクセスするための snc_internal ロール、または外部リソースにアクセスするための snc_external ロールを持つ必要があります。いずれの明示的もロールを持たないユーザーは、公開リソースにのみアクセスできます。
- 既存のユーザー全員に、snc_internal ロールが自動的にアサインされます。このロールでは、既存のアクセスレベルやシステムの動作は変更されません。その代わり、内部ユーザーと外部ユーザーを区別するためのカテゴリが提供されます。内部ユーザーは全員、プラグインが有効化される前と同レベルのアクセスを維持します。

  ヒント: ユーザーの既存の機能を変更しないようにするために、明示的なロールプラグインが有効になると、インスタンス内の既存ユーザー全員に snc_internal ユーザーロールがアサインされます。既存ユーザーには、明示的なロールプラグインが有効になる前に追加された外部ユーザーも含まれます。明示的ロールプラグインが有効になった後、明示的ロールプラグインの有効化以前に追加されたすべての外部ユーザーに対して、以下を実行します。

  - snc_internal ロールを削除します。
  - snc_external ロールを追加します。

  これにより、明示的なロールプラグインを有効化する前に追加されていた外部ユーザーは、内部ユーザーのみが使用できる内部リソースにはアクセスできなくなります。

- 新しく作成されたユーザーは、最初にインスタンスにログインしようとした時点で snc_internal ロールが自動的にアサインされますが、snc_external ロールが明示的にアサインされている場合

は例外です。新しいユーザーがインスタンスに最初にログインする前に snc_external ロールを追加すると、外部ユーザー権限を提供することができます。

> **重要:**
>
> メインテナンス期間中またはログインしているユーザーが少ないときに、このプラグインを有効にします。プラグインが有効化されているときに現在ログインしているユーザーには、snc_internal ロールは動的にアサインされません。むしろ、ユーザーは snc_internal ロールをアサインするために、ログアウトして再度ログインする必要があります。プラグインが有効化されると、snc_internal および snc_external ロールをいつでも追加または削除して、ユーザーの権限を変更できます。
>
> プラグインが有効になると、ユーザーがログインするたびに、アカウントに snc_internal ロールが付与され (ない場合)、snc_external ロールが付与されます。これには、代理操作を介してログインしたユーザーが含まれます。

- ロール要件を持たないすべての既存の ACL に、snc_internal ロールが自動的にアサインされます。既存の ACL とユーザーの両方に snc_internal ロールがアサインされているため、既存のアクセスレベルは変更されません。
- ロール要件を持たない新規作成 ACL には、snc_internal ロールが自動的にアサインされます。このロールは、新規作成 ACL にロールがアサインされている場合にはアサインされません。
- **Type=script** のすべての既存のプロセッサー [sys_processor] レコードまたは新しく作成されたプロセッサー [sys_processor] レコードについては、フィールドが空の場合、snc_internal ロールが [ロール] フィールドに自動的に追加されます。
- UI ページへのアクセスを内部ユーザーに制限する場合、プラグインによって **ui_page** タイプの* ACL に snc_internal ロールが自動的に割り当てられます。
- プロセッサーへのアクセスを内部ユーザーに制限する場合、プラグインによって **ui_page** タイプの* ACL に snc_internal ロールが自動的に割り当てられます。
- 外部ユーザーがインスタンスにアクセスするためには、少なくとも snc_external ロールを取得する必要があります。このロールは、外部ユーザーに手動で付与する必要があります。レコードへのアクセスは、ACL を通じて付与されます。

明示的なロールプラグインが有効であるかどうかにかかわらず、システムアップデートセットをインスタンス間で移動しないでください。詳細については、「システムアップデートセット」を参照してください。

> **注:** このプラグインには、Contextual Security Manager プラグインも必要です。

glide.security.explicit_roles.do_not_fix の動作

Xanadu リリースの時点で、glide.security.explicit_roles.do_not_fix は、snc_internal への変更が調整されました。snc_internal ロールは、メモリ内と sys_user_has_role 内の両方で同じになりました。glide.security.explicit_roles.do_not_fix の新しい動作は次のとおりです。

glide.security.explicit_roles.do_not_fix の新しい動作

値	結果
False	メモリと sys_user_has_role の両方で、snc_internal ロールを追加します。
True	メモリまたは sys_user_has_role で、snc_internal ロールを追加しません。

特定のユーザーの snc_internal ロールを除外するには、glide.security.explicit_roles.ignore.snc_internal.exclude_role_list プロパティを使用します。

ヒント: 以前の glide.security.explicit_roles.do_not_fix の動作に戻すには、glide.security.explicit_roles.do_not_fix_in_memory プロパティを使用します。

外部ユーザーへのテーブルアクセスの提供

snc_external ロールを継承するロールをテーブルに追加することで、外部ユーザーにテーブルへのアクセス権を付与できます。詳細については、「外部ユーザーへのテーブルアクセス権の提供」を参照してください。

hasRoles() メソッド

hasroles() メソッドはまだ利用できますが、 Geneva リリースでは廃止されます。代わりに hasRole(ロール名) メソッドを使用します。

hasRoles() メソッドを使用する場合は、次のような変更に注意してください。

- この方法では、ロールをチェックする際にデフォルトの snc_internal ロールが自動的に除外されます。これは、ユーザーが snc_internal ロールしか持っていない場合でも、hasRoles() メソッドが **false** を返すことを意味します。
- ユーザーが snc_external ロールを持っている場合、インスタンスは外部ユーザーがロールを持たないと判断するため、このメソッドは **false** を返します。

相互除外：**snc_external** 対 **snc_internal**

ServiceNow AI Platform では、snc_external ロールと snc_internal ロールの両方をユーザーに付与することはできません。ServiceNow AI Platform では、 システム内のすべての場所でこの相互除外が適用され、エラーメッセージが各競合のログに書き込まれます。

注: ACL リソースにすべてのユーザーがアクセスできるようにするには、ACL に両方のロールを含めます。

例：ユーザーに明示的なロールの両方を追加 (直接衝突)：

1. ユーザー Abel Tuter に snc_internal ロールをアサインします。
2. ユーザー Abel Tuter に snc_external ロールをアサインします。

結果：Abel Tuter は snc_internal ロールを持つため、snc_external ロールの追加に失敗しました。

例：グループに両方の明示的なロールを追加 (直接衝突)：

1. 現在グループに割り当てられている明示的なロールがないテストグループと呼ばれるグループを検討してください。
2. テストグループにアベル、担当 tuter を追加します。
3. テストグループに snc_external ロールを追加します。

結果：Abel Tuter はすでに snc_internal ロールを持っており、両方のロールを持つことはできないため、snc_external ロールの追加に失敗しました。

例：グループメンバーが競合する明示的なロールを持つグループに明示的なロールを追加します (間接衝突) ：

1. ユーザー Abel Tuter に snc_internal ロールをアサインします。
2. 現在グループに割り当てられている明示的なロールがないテストグループと呼ばれるグループを検討してください。
3. テストグループにアベル、担当 tuter を追加します。
4. snc_external ロールをテストグループに追加します。

結果：Abel Tuter はグループメンバーシップを通じて snc_external ロールを継承するため、グループに snc_external ロールを追加することはできません。両方の明示的なロールが同じユーザーに割り当てられます。これは許可されていません。

その他の例については、次のテーブルを参照してください。

ロール	試行したアクション	結果
直接衝突		
ユーザーは snc_internal ロールを持っています。	snc_external ロールを追加します。	アクションが中断されました。
ユーザーは snc_external ロールを持っています。	snc_internal ロールを追加します。	アクションが中断されました。
ユーザーに明示的なロールがありません。	snc_external と snc_internal の各ロールを追加します。	ロールが追加されました。
ユーザーは、両方の明示的なロールを持ちます (衝突が存在します) 。	ロールを持たないグループにユーザーを追加してください。	アクションが中断されました。
どのユーザーにも関連付けられていないロールに、snc_internal ロールが含まれています。	snc_external ロールを追加します。	アクションが中断されました。
どのユーザーにも関連付けられていないロールには、snc_external ロールが含まれています。	snc_internal ロールを追加します。	アクションが中断されました。
ロールには、明示的なロールが含まれています (衝突が存在します) 。	ロールをユーザー、ロール、またはグループに追加します。	アクションが中断されました。
メンバーを含まないグループに snc_internal ロールが割り当てられています。	snc_external ロールを追加します。	アクションが中断されました。
メンバーを含まないグループに snc_external ロールが割り当てられています。	snc_internal ロールを追加します。	アクションが中断されました。
メンバーを含まないグループにはロールはありません。	snc_external と snc_internal の各ロールを追加します。	ロールが追加されました。
間接衝突		

ロール	試行したアクション	結果
衝突を包有するロール	1. snc_internal ロールを持つユーザーに Test ロールと呼ばれるロールを付与します。 2. snc_external ロールの Test ロールを追加します。	アクションが中断されました。
衝突を包有しないロール	1. ロールを持たないユーザーに Test ロールと呼ばれるロールを付与します。 2. snc_external ロールに Test ロールを追加します。	このロールは、ユーザーと Test ロールの両方に追加されます。
衝突を包有するグループ	1. snc_internal ロールを持つユーザーを、Test Group 2 (Test Group 1 の子) というグループに追加します。 2. snc_external ロールを Test Group 2 に追加します。 3. snc_external ロールを Test Group 1 (Test Group 2 の親) という親グループに追加します。	アクションが中断されました。
衝突を包有しないグループ	1. ロールを持たないユーザーを、Test Group 2 (Test Group 1 の子) というグループに追加します。 2. snc_external ロールまたは snc_internal ロールを Test Group 1 (Test Group 2 の親) に追加します。	ロールが親グループ、子グループ、およびユーザーに追加されます。
グループとロールに衝突が包有されています	Test Group 1 (Test Group 2 の親) に contains_external を追加します。	Test Group 1 と Test Group 2 はともに contains_external を取得しますが、snc_external ロールを明示的に取得することはできません。
	Test Group 1 の子である Test Group 2 に snc_internal ロールを追加します。	アクションが中断されました。

自動翻訳

ロール	試行したアクション	結果
グループの親の変更とグループの包有	1. Test Group 2 の親として Test Group 1 を削除します。 2. snc_internal ロールを Test Group 1 に追加します。 3. snc_external ロールを Test Group 2 に追加します。 4. Test Group 2 では、Test Group 1を親グループとして設定して保存します。	アクションが中断されました。 同じ想定で、既にネストされたグループに対して手順を繰り返します。

アクションの中断の原因がエラーメッセージに表示され、別の試行が成功する前に対応する必要があります。

個々のユーザーに明示的ロールを追加するなど、直接的なケースでは、ユーザーがどの明示的ロールを持つ必要があるかを確認します。ユーザーが不適切な明示的なロールを持つ場合は、まず削除し、次に正しい明示的なロールを追加する必要があります。

グループに明示的なロールを追加するなど、間接ケースでは (グループメンバーに両方の明示的なロールがアサインされるようにするために)、ユーザーをグループに含める必要があるかどうかを評価します。グループ階層とロールの包含を通じた継承を含め、グループに明示的なロールを指定する必要があるかどうかも決定します。

ServiceNow AI Platform では、最初に発生した衝突の可能性のみがレポートされることに注意してください。修正後も繰り返し試行が失敗し、毎回新しい根本原因が発生する場合は、関連するユーザー/グループ/ロールの相互依存関係をより広範に再評価してください。グループとロールの包有がどのように構成されているかを再考することをお勧めします。

明示的なロールを要求する

Now Support サービスカタログから明示的なロールプラグイン (com.glide.explicit_roles) を要求して明示的なロールをアクティブ化します。

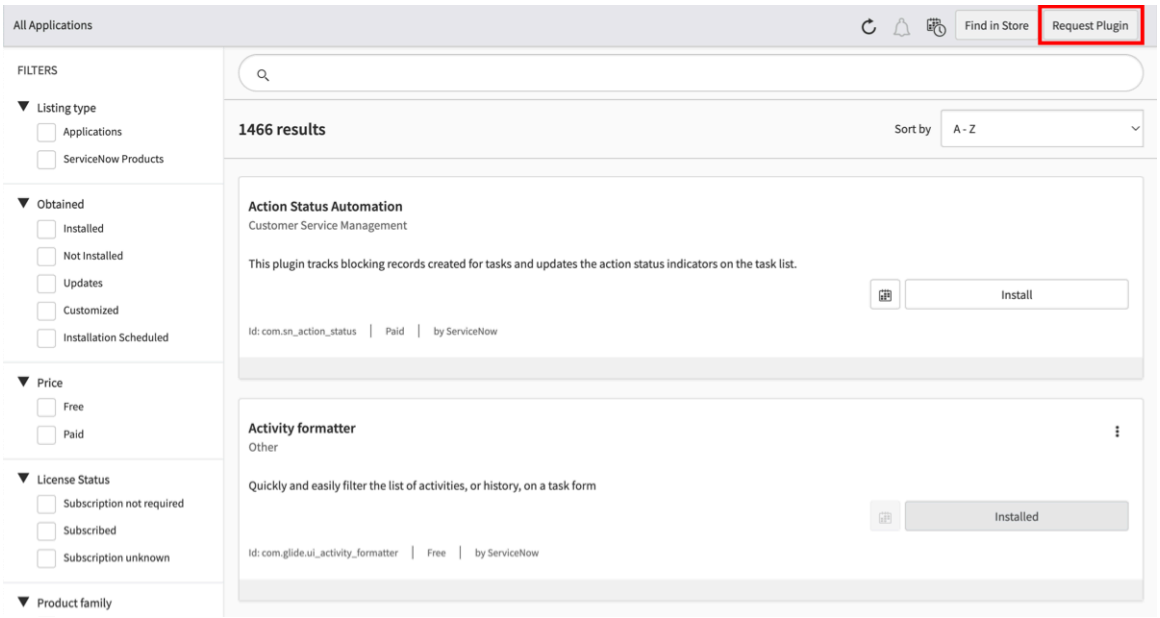
始める前に

必要なロール：admin

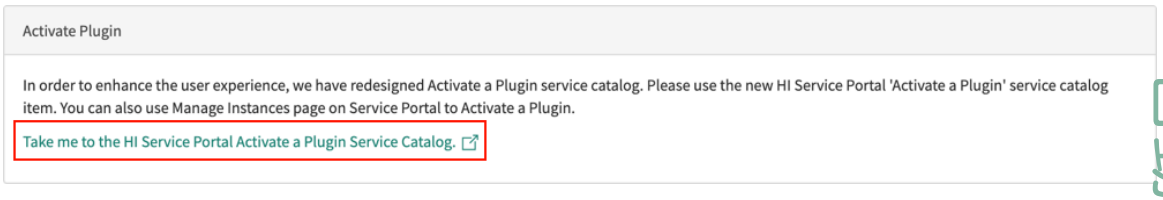
> **重要:** メインテナンス期間中またはログインしているユーザーが少ないときに、このプラグインを有効にします。プラグインが有効化されているときに現在ログインしているユーザーには、snc_internal ロールは動的にアサインされません。むしろ、ユーザーは snc_internal ロールをアサインするために、ログアウトして再度ログインする必要があります。プラグインが有効化されると、snc_internal および snc_external ロールをいつでも追加または削除して、ユーザーの権限を変更できます。

手順

1. 移動先 すべて > システムアプリケーション > 利用可能なすべてのアプリケーション > すべて.
2. [すべてのアプリケーション] ページで [プラグインの要求] をクリックして、Now Support で [プラグインをアクティブ化] フォームを開きます。



3. Now Support で、Now Support サービスポータル サービスカタログ にアクセスするリンクを選択します。



4. インスタンスを選択します。
5. [アクション] > [プラグインのアクティブ化] を選択します。
6. [プラグインのアクティブ化] フォームで、次の情報を入力します。

[プラグインのアクティブ化] フォーム

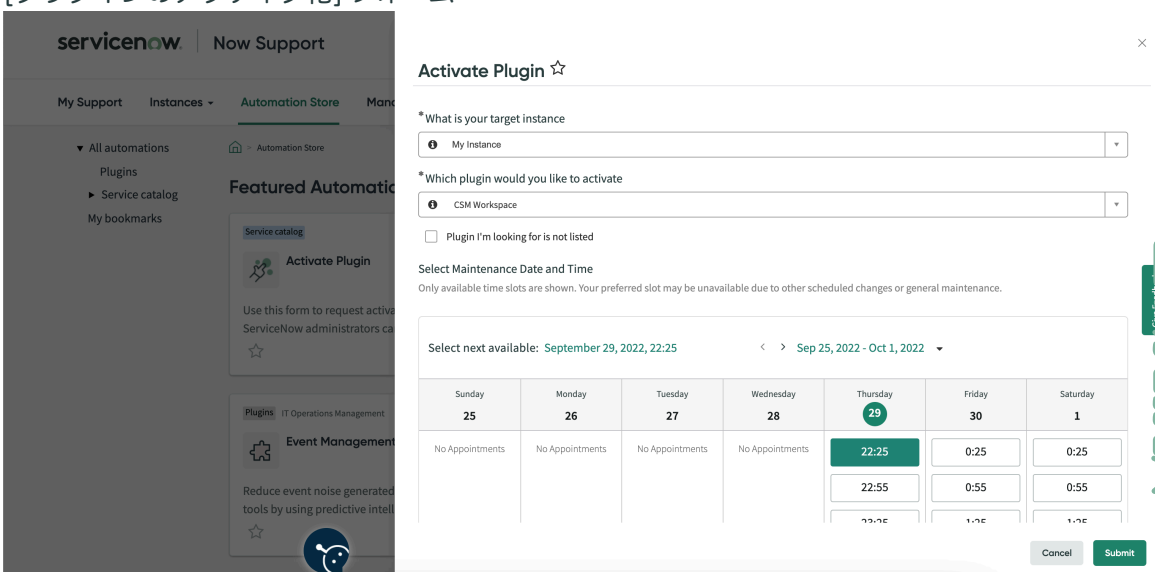
フィールド	説明
ターゲットインスタンスは何ですか	プラグインをアクティブ化するインスタンス。
どのプラグインをアクティブ化しますか	アクティブ化するプラグインの名前です。 注: 必要なプラグインが表示されない場合、または OEM またはオンプレミスのインスタンスでプラグインをアクティブ化している場合は、[探しているプラグインが表示されていません **(Plugin I'm looking for is not listed)**] チェックボックスをオンにして、プラグインの名前を入力します。
	プラグインをアクティブ化する日時。

フィールド	説明
メンテナンスの日時を選択 (Select Maintenance Date and Time)	注: プラグインは、米国太平洋標準時で、毎営業日の朝と夕方の 2 回のバッチでアクティブ化されます。特定の時刻にプラグインをアクティブ化する必要がある場合は、[理由/コメント **(Reason/Comments)]** フィールドに要求を入力します。

Example

たとえば、 [自分のインスタンス (My Instance)] という名前のインスタンスで CSM Workspace プラグインをアクティブ化するには、次のフォームを参照してください。

[プラグインのアクティブ化] フォーム



7. **[Submit** (送信)] を選択します。
   プラグインの要求の詳細については、次を参照してください。 のサービスカタログ [KB0751715] 記事からのプラグインの要求 Now Support ナレッジベース.

昇格された権限ロール

昇格された権限ロールでは、ロールの機能にアクセスする前に、そのロールを使用する責任を手動で受け入れる必要があります。

デフォルトでは、ログイン時に昇格された権限のロールはありません。ロールの権限に手動で昇格させる必要があります。昇格された権限のロールは、ユーザーセッションの間のみ有効です。セッションタイムアウトまたはログアウトでこのロールは削除されます。

任意のロールを昇格された権限のロールとして指定し、そのロールを 1 人以上のユーザーに割り当てることができます。これは、ログイン直後にロールが提供する権限にユーザーがアクセスできないようにする場合に実行します。ロールフォームで権限ロールを指定できます。手順については「ロールの作成 」を参照してください。

昇格したロールを使用するには、次の条件を満たす必要があります。

- 昇格したロールを割り当てる必要があります。
- 最初に昇格したロールを含む 2 番目に昇格したロールに既に昇格している場合でも、その権限を取得するには、特定の昇格したロールに手動で昇格させる必要があります。

  たとえば、昇格したロール A に昇格したロール B が含まれている場合、ロール A に昇格しても、その権限を取得するにはロール B に昇格する必要があります。

admin ロール

ユーザーに admin ロールを付与するには、付与するユーザーも admin ロールが必要です。たとえば、user_admin ロールのみを持つユーザーは、他のユーザーに admin ロールを付与することはできません。

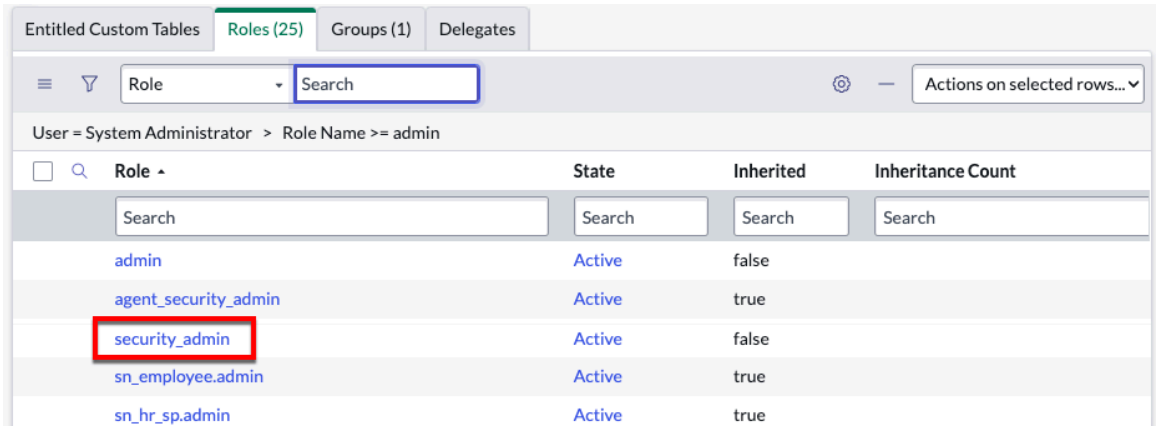
- 非アドミンユーザーは admin ロールが含まれるグループにユーザーを追加できません。
- ユーザーに security_admin ロールを付与するには、そのユーザーも admin ロールを持っている必要があり、他のユーザーに security_admin ロールを付与する前に、security_admin ロールに昇格する必要があります。admin ロールのみを持つユーザーは、他のユーザーに security_admin ロールを付与することはできません。
- security_admin ロールのないユーザーは、security_admin ロールが含まれるグループにユーザーを追加できません。

⚠ **警告:** admin ロールでの昇格された権限の使用はサポートされていません。代わりに、管理者に手動での昇格を要求します。「 アドミニストレーターに手動での昇格を強制する

security_admin ロール

ベースシステムでは、security_admin ロールのみが昇格された権限を持ちます。このロールは、デフォルトのシステムアドミニストレーター (admin) ユーザーに自動的に割り当てられます。ACL および 高セキュリティ設定へのアクセスを提供します。

システムアドミニストレーター **(admin)** ユーザーに割り当てられたロール



ℹ **注:** このロールを表示するには、まず security_admin ロールに昇格する必要があります。システムアドミニストレーター (admin) ユーザーとしてのみログインしている場合、ロールのリストに security_admin レコードは表示されません。

security_admin ロール

security_admin ロールは、ユーザーがアクセス制御の作成や変更を行ったり、高セキュリティ設定を変更したりできる、高セキュリティ設定で提供される昇格された権限ロールです。

ベースシステムでは、デフォルトのシステムアドミニストレーター (admin) ユーザーのみが security_admin ロールを持っています。昇格する権限が必要なため、admin ユーザーはログイン

時にこのロールを持っていません。権限を昇格した後、admin ユーザーはユーザーセッションの間、security_admin ロールを持ちます。詳細については、「特権ロールへの昇格」を参照してください。

高いセキュリティを維持するために、security_admin ロールには昇格する権限が必要です。このロールを割り当てるユーザーとグループを制限します。

特権ロールへの昇格

ベースシステム admin は、高セキュリティ設定の機能にアクセスするために、特権ロールに昇格できます。

始める前に
必要なロール：admin

注: 追加のユーザーに admin ロールを付与した場合、そのユーザーは特権ロールに昇格できません。ベースシステム admin のみが昇格できます。

手順

1. バナーフレームで画像をクリックします。画像をアップロードしていない場合は、自分のイニシャルをクリックします。
2. [ロールを昇格] を選択します。
   昇格に利用可能なロールを含むダイアログボックスが表示されます。
3. アサインする昇格済みロールを選択し、**[OK]** をクリックします。
   このロールは、残りのセッションでこのロールによって制御されるすべてのリソースに対する昇格された権限をユーザーに付与します。ユーザーがログアウトすると、昇格した権限はセッションで終了しますが、次回のログイン時に再確立できます。
4. 手順 2 のダイアログボックスに戻り、ロールの選択を解除して、ロールの昇格を終了します。

アドミニストレーターに手動での昇格を強制する

アドミニストレーターロールを持つすべてのユーザーに、昇格するロールを手動で選択させるプロパティがあります。

始める前に
必要なロール：security_admin

手順

1. admin としてログインします。
2. ロールを security_admin に昇格させます。
3. **sys_properties.list** に移動します。
4. *glide.security.strict_elevate_privilege* プロパティを検索して選択します。
5. [値] フィールドを **true** に設定し、[送信] をクリックします。

結果
ユーザーがログインすると、昇格できるロールを選択するダイアログウィンドウが表示されます。

接続と資格情報

ディスカバリー、サービスマッピング、およびクラウド管理用にコンピューターまたはネットワークデバイスへのアクセス権を得る、またはオーケストレーションを使用して作業を実行するには、認証

情報と接続情報が必要です。共有または AppStore にコンテンツを追加すると、作成されたコンテンツを変更することなく、環境に関連する接続と資格情報を設定することができます。

探索

認証情報について学習します。

構成

認証情報を設定します。

参照

認証情報の詳細を取得します。

トラブルシューティング

接続と資格情報の問題をトラブルシューティングする方法について説明します。

自動翻訳

資格情報、接続、およびエイリアスの詳細

ServiceNow AI Platform でのすべてのアプリケーション統合では、接続、資格情報、およびエイリアスを使用して、それぞれのアプリケーションがリソースにアクセスできるようにします。

ServiceNow AI Platform でアプリケーションデータ連携を実行する前に、接続情報および対応する認証情報を作成して、エイリアスを追加する必要があります。ServiceNow でのこれらの用語の定義を確認するには：

接続

「接続」とは、プロトコルを使用する IP アドレスやエンドポイントなどの、システムとのデータ連携です。これには、データベースと統合する際のデータベースの詳細など、特定の詳細が含まれています。

認証情報

「認証情報」とは、ID やパスワードなど、接続に必要な認証データです。

エイリアス

「エイリアス」とは、インスタンス上の一連の接続または認証情報に関連付けられている命名規則またはタグです。エイリアスには、アプリケーションのデータ連携に必要な接続および認証情報が含まれています。データ連携するたびにその情報を入力する代わりに、エイリアスを使用できます。たとえば、同じアプリケーションデータ連携の QA、開発、および本番認証情報を格納するエイリアスを指定できます。エイリアスは、各環境のアプリケーションデータ連携を解決します。

ServiceNow AI Platform は、さまざまなタイプのエイリアスを区別します。

認証情報エイリアス

このエイリアスは認証情報データにのみ関連付けられ、実行時に解決します。

接続および資格情報エイリアス

このエイリアスは、データ連携を完了するために必要な接続情報と認証情報データに関連付けられ、実行時に解決します。

接続および資格情報エイリアス内で、「子エイリアス」と呼ばれる追加のエイリアスを作成することもできます。子エイリアスを使用すると、同じアプリケーションデータ連携内で複数の接続を作成できます。子エイリアスを作成すると、作成したエイリアスが属するエイリアスが「親エイリアス」になります。子エイリアスは親エイリアスからプロパティを継承しますが、子エイリアスは独自の接続および認証情報を保持します。

接続、資格情報、およびエイリアスを使用するメリット

- 外部サービスの認証情報を一元的に保存および管理できます
- 一度定義すれば、複数のプラットフォーム機能に再利用できます
- 他のプラットフォーム機能の構成を最小限に抑えることができます
- アドミン以外のユーザーに、事前定義された接続および認証情報の使用を許可できます
- セキュリティを強化できます

接続、資格情報、およびエイリアスを使用する機能

以下の機能では、接続、資格情報、およびエイリアスを使用します。

- フローデザイナー
- 統合ハブ
- Cloud Management
- ディスカバリー
- オーケストレーション
- サービスマッピング

ServiceNow AI Platform でエイリアスを設定するには、次の 2 つの方法のいずれかを使用します。

- 接続情報と認証情報 モジュールを使用します。「接続情報および認証情報エイリアスの作成」を参照してください。
- 統合ハブ の接続ダッシュボードで設定します。「接続の追加」を参照してください。

> **注:** 統合ハブ では別のサブスクリプションが必要です。詳細については、次を参照してください。 要求 統合ハブ .

MID サーバーでの認証情報の同期

インスタンスと同期しているネットワーク内の各 *MID* サーバーは、作成したすべての認証情報のコピーを保持します。管理、計測、ディスカバリー (MID) サーバーは、ServiceNow インスタンスと外部のアプリケーション、データソース、サービスの間で行われるデータの通信や移動を支援する Java アプリケーションです。この同期により、ディスカバリー や サービスマッピング などのアプリケーションがネットワーク上の複数のデバイスにアクセスする必要がある場合に認証情報の読み取り速度が向上します。MID サーバーは、ECC キューで `credentials_reload` ジョブが見つかったときに同期します。再ロード ジョブは、すべてのフィールド値を含む認証情報 [discovery_credentials] テーブル内の認証情報のリスト全体を取得するために、MID サーバーにインスタンスへの SOAP 呼び出しの実行を指示します。詳細については、「MID サーバー」を参照してください。

インスタンスを各 MID サーバーに送信する SOAP 応答には、カスタマイズした認証情報フォームに追加したカスタム フィールドも含まれます。参照フィールドを追加すると、参照テーブルのデータも SOAP 応答の一部として送信されます。これにより、複数の MID サーバーで認証情報の同期が発生したときにパフォーマンスの問題が発生する可能性があります。これを制御するには、次のプロパティをシステム プロパティ [sys_properties] テーブルに手動で追加します。

> **注:** 次のプロパティの値を変更するには、それらの値をシステム プロパティ [sys_properties] テーブルに追加します。それらの値を追加しない場合、デフォルト値が使用されます。

プロパティ	説明
com.snc.credentials_user_fields	認証情報の同期でカスタマイズされたすべてのフィールドが含まれます。認証情報フォームに追加したフィールドを含めない場合は、このプロパティを false に設定します。 • タイプ：true | false • デフォルト値：true
com.snc.credentials_recursion_depth	認証情報の同期メカニズムによって参照テーブルからフィールドが収集されるときにトラバースするテーブルの数を定義します。パフォーマンスの問題が発生し、参照フィールド

プロパティ	説明
	も含まれているテーブルへの参照フィールドを含む認証情報フォームをカスタマイズしている場合は、この数を少なくします。 • タイプ：整数 • デフォルト値：3

認証情報および接続のスコープ保護

特定のタイプの接続および資格情報レコードをスコープに属するものとして分類し、スコープの保護をそれらに拡張することができます。 これらのスコープポリシーは、テーブルで作成したレコードを保護し、別のスコープとのプライベートなレコードのやり取りを防止します。

接続 [sys_connection] テーブルおよびディスカバリー認証情報 [discovery_credentials] テーブルの [アプリケーション] フィールドを使用して、これらのタイプのレコードを特定のスコープに関連付けることができます。Zurich の UI フォームには表示されませんが、簡単に追加できます。これらのレコードタイプの詳細と、UI フォームへのフィールドの追加方法については、以下を参照してください。

- 接続の開始
- 認証情報の使用を開始する
- ディスカバリーの認証情報エイリアス
- フォームレイアウトの構成

接続および資格情報レコードの使用を特定のスコープに制限することは、セキュリティを強化する必要があるアプリケーションを管理するために重要です。これらのアプリケーションには HR サービスデリバリ (HRSD) または セキュリティオペレーション スコープ指定の管理対象アプリケーションで作成された接続および認証情報レコードは、admin ユーザーには表示されません。 接続および認証情報レコードを特定のアプリケーションスコープに関連付けると、次の保護が得られます。

- 制限付きスコープにアクセス制御リストルール (ACL) が適用されます。スコープ指定の ACL の詳細については、「アクセス制御リストのルール」を参照してください。

  注: スコープ管理および強化セキュリティを使用する一部のアプリケーションでは、追加のセットアップが必要になる場合があります。詳細については、「人事ロールの管理 」を参照してください。

- スクリプトを使用してクエリーが実行されたときにレコードを保護します。グローバルスコープからクエリーを実行し、接続および認証情報レコードが保護されたスコープ内にある場合、アクセス権が付与されていない限り、クエリーには表示されません。

制限付きの発信者アクセスを使用して、クエリー制限付きレコードへのアクセスをカスタマイズおよび付与できます。詳細については、「制限付き発信者アクセス特権の設定 」を参照してください。スコーピング制限は、接続 [sys_connection] テーブルおよびディスカバリー認証情報 [discovery_credentials] テーブルのすべての子テーブルにも適用されます。空のフィールドおよびその他のスコープは制限されません。

注: スコープ保護は、新しいレコードを設定する際の混乱を避けるために、特定の保護されたスコープに対してのみ有効になります。誰かがスコープ対象のアプリケーションスコープで接続を行っても、自動的にスコープが制限されることはありません。

ドメインセパレーション、認証情報および接続

ドメインセパレーションは認証情報および接続でサポートされています。 ドメインセパレーションでは、データ、プロセス、および管理タスクをドメインと呼ばれる論理的なグループに分けることができます。どのユーザーがデータを表示できるか、データにアクセスできるかなど、このアプリケーションのいくつかの側面を制御できます。

サポートレベル：標準

- ベーシックレベルのサポートのすべての側面が含まれます。
- アプリケーションプロパティは、必要に応じてドメイン対応です。
- ビジネスロジック：サービスプロバイダー (SP) によって顧客ごとにプロセスを作成または変更できます。ユースケースには、単一のインスタンスでの複数のサービスプロバイダー顧客によるアプリケーションの正しい使用が反映されています。
- インスタンスのオーナーは、特定のアプリケーションに期待される通りに、テナントごとに MVP ビジネスロジックとデータパラメーターを設定できる必要があります。

サンプルユースケース：アドミンは、レコードを他のテナントに対してはクローズしないが、1 つのテナントに対してクローズする場合、コメントを必須にすることができる必要があります。

サポートレベルの詳細については、「アプリケーションでのドメインセパレーションのサポート)」を参照してください。

概要

認証情報は、インスタンス外のシステムにアクセスするさまざまなServiceNow機能に関連付けられます。認証情報は、認証情報を使用する機能に関連付けられたドメインセパレーションに従います。

接続情報は、インスタンス外のターゲット ホストを参照するプロトコル固有の情報です。接続情報には、アクティビティを実行するドメインを指定できます。

認証情報および接続情報でのドメインセパレーションの仕組み

認証情報はインスタンス外のリソースにアクセスし、 ディスカバリー 、 オーケストレーション 、 サービスマッピング 、および クラウドプロビジョニングとガバナンス 更新があるとわかりますこれらの認証情報は、特定のドメインに関連付けられるのではなく、アプリケーションにバインドされ、アプリケーションで使用されるドメインセパレーションに従います。また、認証情報を MID サーバー に割り当てて、MID サーバー 構成で指定されたドメインセパレーションに従うこともできます。

接続情報によって、*JMS*、*JDBC*、または *HTTP(S)* 接続を使用してターゲットホストにアクセスします。接続元となるグローバル ドメインまたは特定のドメインを指定できます。

関連トピック

- サービスプロバイダーのドメインセパレーション

接続と資格情報の構成テンプレート

admin および flow_designer ロールを持つユーザーは、カスタマイズ可能な単一のフォームを使用して、サードパーティシステムとのスポーク統合を設定できます。

たとえば、OAuth プロバイダーを登録し、トークンを生成し、接続レコードおよび認証情報レコードを作成する OAuth データ連携を設定できます。アクションデザイナーまたは開発者は構成テンプレートを使用して 1 つの場所でスポークを設定でき、関連付けられたレコードがシステムによって作成されます。

利点

構成テンプレートを使用することで以下のことができるようになります。

- アドミニストレーターまたはフロー設計者が、単一のフォームを使用して複雑な統合を設定できるようになります。
- 開発者がデータ連携で静的な値を設定し、アドミンおよびフロー設計者のセットアッププロセスを簡素化できるようになります。

サポートされている認証情報タイプ

次の認証情報タイプとの統合用の構成テンプレートを作成できます。

- 基本認証
- API キー
- OAuth JWT ベアラー権限許可タイプ
- OAuth 認証コード権限許可タイプ
- カスタム認証

構成テンプレートのコンポーネント

デフォルトのデータテンプレート

すべてのデータ連携に適用される静的情報を設定します。たとえば、値がすべてのデータ連携に適用される場合は、API およびトークンの URL を設定できます。

動的データテンプレート

統合を設定するためにユーザーが完了する必要のある情報を定義します。たとえば、ユーザー名とパスワードのキーと値のペアを追加して、ユーザー定義の値を収集することができます。

後処理スクリプト

統合によって必要とされる追加のレコードを作成します。たとえば、スポークにカスタムテーブルが含まれている場合は、構成テンプレートのユーザー入力に基づいてそれらのテーブルにレコードを作成できます。このスクリプトは、接続と資格情報レコードが作成された後に実行されます。

事前編集スクリプト

既存の接続を編集するときに、[追加情報] セクションのカスタムフィールドを事前に入力します。カスタムフィールドを事前入力すると、カスタムフィールドに関連付けられている現在の値を表示できます。

テストアクション

ワークフロースタジオ のフロービューで統合アクションから直接接続をテストできます。テストアクションでは、アクション定義を使用して、テンプレートが現在添付されているエイリアスをテストします。詳細については、「構成テンプレートから接続エイリアスをテストするテストアクションの作成」を参照してください。

デモデータ

接続と資格情報のテンプレート [sys_alias_templates] テーブルには、一般的な認証タイプのテンプレートを設定する方法を示すテンプレートの例が含まれています。独自の例を作成するときは、これらの例をガイドとして使用してください。

OAuth JWT ベアラー権限許可タイプのテンプレートの構成

この構成テンプレートの例では、Docusign への要求を認証するために JWT ベアラー権限許可タイプを使用して、認証情報レコードおよび接続レコードを設定します。

デフォルトのデータテンプレート

デフォルトのデータテンプレートの各トップレベルのアイテムでは、関連するレコードが作成されます。テンプレートには、次のセクションが含まれています。

- 認証情報：認証情報テーブルにレコードを作成します。
- 接続：接続 [sys_connection] テーブルおよび関連する接続レコードにレコードを作成します。
- 追加：オプションでカスタムテーブルにレコードを作成します。後処理スクリプトは、システムがこれらのレコードを処理する方法を指定します。

次の例では、OAuth JWT ベアラー権限許可タイプ認証に必要なレコードを作成します。

```
{
 "credential": {
  "oauth_entity": {
   "oauth_entity_profile": [
    {
     "grant_type": "urn:ietf:params:oauth:grant-type:jwt-bearer",
     "name": "Docusign Profile",
     "default": true,
     "oauth_entity_profile_scope": [
      "users:read.email"
     ]
    }
   ],
   "code_challenge_method": "S256",
   "type": "consumer",
   "oauth_entity_scope": [
    {
     "oauth_entity_scope": "users:read.email",
     "name": "email"
    }
   ],
   "client_id": "<provider-client-id>",
   "use_mutual_auth": false,
   "revoke_token_url": "https://<provider-domain-name>.com/oauth2/revoke",
   "default_grant_type": "urn:ietf:params:oauth:grant-type:jwt-bearer",
   "public_client": false,
   "oauth_api_script": "3e3a3a11c333210016194ffe5bba8f70",
   "name": "Docusign Spoke OAuth",
   "client_secret": "<provider-client-secret>",
   "auth_url": "https://<provider-domain-name>.com/oauth2/auth",
   "token_url": "https://<provider-domain-name>.com/oauth2/token",
   "redirect_url": "https://<instance-name>.service-now.com/oauth_redirect.do"
  },
  "jwt_provider": {
   "jwt_keystore_aliases": {
    "kid": "<provider-key-id>",
    "name": "Docusign Spoke JWT Key",
    "signing_keystore": "<signing-keystore-sys-id>",
    "signing_algorithm": "rsa_256",
```

```
      "signing_key_password": "password"
    },
    "jwt_claim_validation" : [ {
      "name" : "iss",
      "is_standard" : true,
      "data_type" : "string",
      "value":"<docusign-iss-claim>"
    }, {
      "name" : "sub",
      "is_standard" : true,
      "data_type" : "string",
      "value":"<docusign-sub-claim>"
    }, {
      "name" : "aud",
      "is_standard" : true,
      "data_type" : "string",
      "value":"<docusign-aud-claim>"
    }, {
      "name" : "scope",
      "is_standard" : false,
      "data_type" : "string",
      "value" : "signature impersonation"
    } ],
    "name": "Docusign Spoke JWT Provider",
    "jwt_api_script": "9ef6af86ff10330001d3cd6bd53bf144"
   },
   "name": "Docusign Spoke Credential",
   "table": "oauth_2_0_credentials"
  },
  "connection": {
   "use_mid": false,
   "connection_url": "https://<provider-domain-name>.com",
   "name": "Docusign Spoke Connection",
   "table": "http_connection"
  },
"additional":{
   "docusign_account_name": "<docusign-account-name>",
   "docusign_account_email": "<docusign-account-email>"
}
}
```

動的なデータスキーマ

動的なデータスキーマは、ユーザーが接続および資格情報エイリアスを作成して入力を収集するときに表示される内容を定義します。ドット連結構文を使用して、デフォルトのデータテンプレートで作成されたフィールドにユーザー入力をマッピングします。たとえば、connection_fields は、デフォルトのデータテンプレートによって作成された connection オブジェクトの connection_url フィールドにユーザー入力をマッピングします。

```
{
  "connection_fields": [
    {
      "name": "connection.connection_url",
      "label": "Connection URL",
      "type": "text",
      "defaultValue": "https://demo.docusign.net",
      "hint": "Connection URL for Docusign"
```

```
  }],
"additional_fields":[
  {
    "name": "additional.docusign_account_id",
    "label": "Docusign Account Number",
    "type": "text",
    "hint": "Docusign Account Number"
  },
  {
    "name": "additional.docusign_account_name",
    "label": "Docusign Account Name",
    "type": "text",
    "hint": "Name to identify the Docusign account"
  },
  {
    "name": "additional.docusign_account_email",
    "label": "Docusign Account Email",
    "type": "text",
    "hint": "Docusign Account Email"
  }
 ],
 "credential_fields": [
  {
    "name": "credential.oauth_entity.client_id",
    "label": "OAuth Client ID",
    "type": "text",
    "hint": "Client ID for Docusign"
  },
  {
    "name": "credential.oauth_entity.redirect_url",
    "label": "OAuth Redirect URL",
    "type": "text",
    "defaultValue": "https://<instance-name>.service-now.com/oauth_redirect.do",
    "hint": "Callback URL for Docusign"
  },
{
    "name": "credential.jwt_provider.jwt_claim_validation[0].value",
    "label": "Issuer (iss) Claim value",
    "type": "text",
    "hint": "The integrator key (also known as client ID) of the application"
  },
{
    "name": "credential.jwt_provider.jwt_claim_validation[1].value",
    "label": "Subject (sub) Claim value",
    "type": "text",
    "hint": "The user ID of the user to be impersonated"
  },
{
    "name": "credential.jwt_provider.jwt_claim_validation[2].value",
    "label": "Audience (aud) Claim value",
    "type": "text",
    "defaultValue": "account-d.docusign.com",
    "hint": "The URI of the authentication service instance to be used e.g.
account.docusign.com"
  },
  {
    "name": "credential.jwt_provider.jwt_keystore_aliases.kid",
```

```
    "label": "Key ID (kid)",
    "type": "text",
    "hint": "Indicates which key was used to secure the JWS"
   },
   {
    "name": "credential.jwt_provider.jwt_keystore_aliases.signing_keystore",
    "label": "Key Store",
    "type": "file"
   }
 ]
}
```

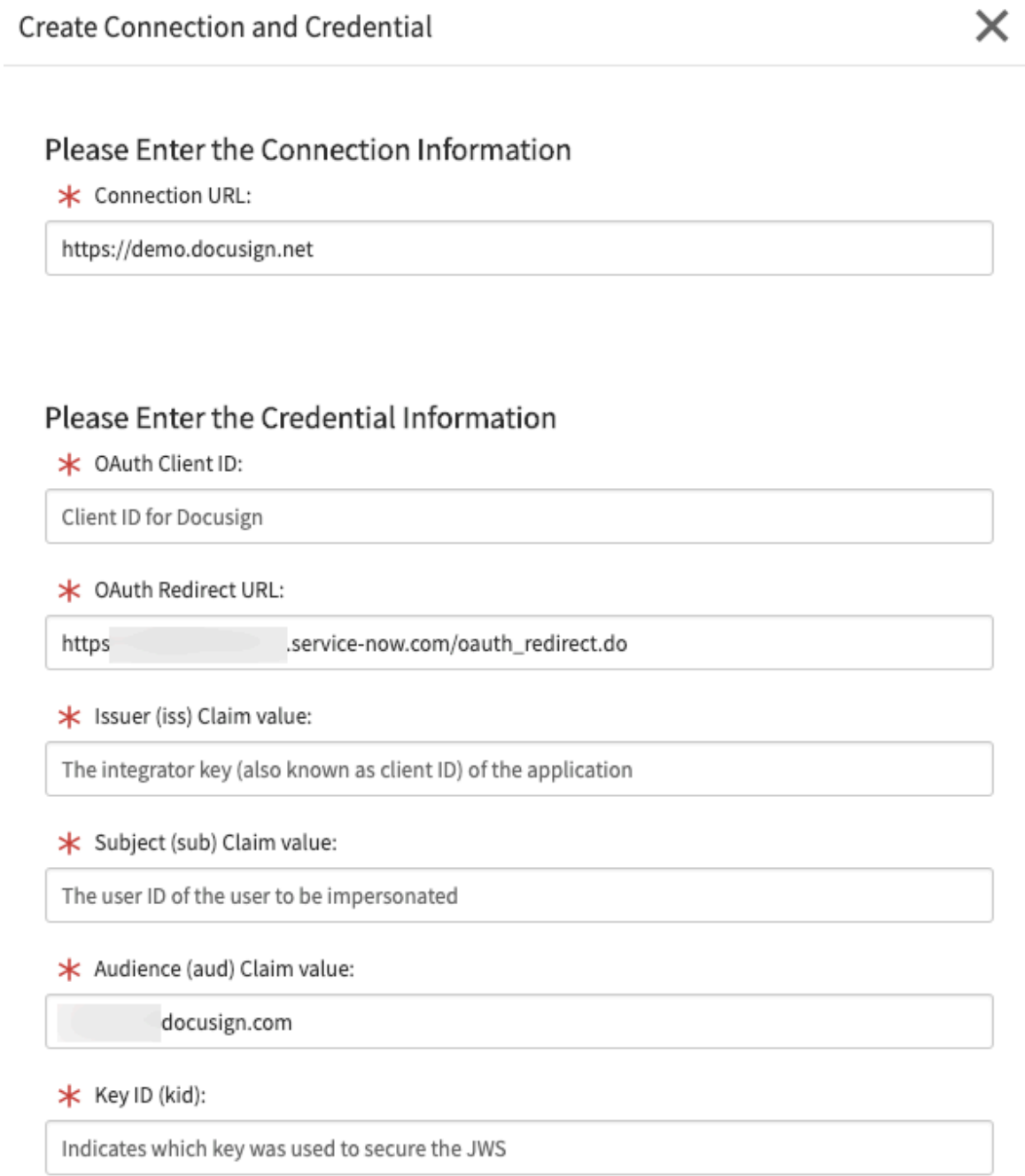
後処理スクリプト

次の後処理スクリプトは、ユーザー入力を sn_docusign_spoke_accounts テーブルのフィールドにマッピングします。

```
(function execute(aliasId, connectionSysId, jsonDefaultData, jsonDynamicData) {
 var jsonDynamicDataP = JSON.parse(jsonDynamicData);
 var accountGR = new GlideRecord("sn_docusign_spoke_accounts");
   accountGR.setValue("account_name",
 jsonDynamicDataP["additional.docusign_account_name"]);
   accountGR.setValue("alias", aliasId);
   accountGR.setValue("email", jsonDynamicDataP["additional.docusign_account_email"]);
   accountGR.setValue("id", jsonDynamicDataP["additional.docusign_account_id"]);
   accountGR.insert();
})(aliasId, connectionSysId, jsonDefaultData, jsonDynamicData);
```

結果の **Docusign** 接続および認証情報の構成フォーム

ユーザーが関連する Docusign 接続および資格情報エイリアスに移動して [新しい接続と資格情報の作成] を選択すると、次のダイアログが表示されます。



構成テンプレートの作成

スポークの設定に必要な入力を定義するテンプレートを作成します。静的なキーと値のペアを設定してレコードを作成し、すべてのデータ連携に適用される値を設定します。動的なキーと値のペアを設定してユーザー入力を収集し、さまざまなフィールド値を設定します。このテンプレートを使用すると、アドミンおよびフロー設計者は単一のフォームからスポークを設定できます。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > 統合ハブ > 接続 **&** 認証情報 > 構成テンプレート.

2. [New (新規)] を選択します。

3. 作成する構成テンプレートのタイプを選択します。

構成タイプ	説明
OAuth 認証コード権限許可タイプを使用した HTTP 接続	認証コードを使用して、サードパーティシステムを OAuth プロバイダーとして登録するためのテンプレートを作成します。
OAuth JWT ベアラー権限許可タイプを使用した HTTP 接続	JSON Web トークンを使用して、サードパーティシステムを OAuth プロバイダーとして登録するためのテンプレートを作成します。
クライアント認証情報権限許可タイプを使用した HTTP OAuth	クライアント認証情報権限許可タイプを使用する OAuth 2.0 認証を介してサードパーティアプリケーションと統合するためのテンプレートを作成します。
OAuth クライアント認証情報権限許可タイプを使用した HTTP 接続 (外部ストレージ)	外部ストレージに保存されているクライアント認証情報で構成される OAuth トークン要求を、MID サーバーを介して OAuth サーバーに送信できるようにするテンプレートを作成します。
基本認証情報を使用した HTTP 接続	ベーシック認証を使用して、サードパーティシステムとデータ連携するためのテンプレートを作成します。
API キー認証情報を使用した HTTP 接続	API キーを使用して、サードパーティシステムとデータ連携するためのテンプレートを作成します。
その他の構成	空のテンプレートを作成し、カスタム認証用のテンプレートを設定できるようにします。

4. [名前] フィールドに、テンプレートを識別するための名前を追加します。
5. [デフォルトのデータテンプレート] フィールドで必要な変更を行います。

   このフィールドで、すべてのデータ連携に適用される静的情報を設定します。たとえば、値がすべてのデータ連携に適用される場合は、API およびトークンの URL を設定できます。

   デフォルトのデータテンプレートにある次のオブジェクトは必須です。

   - `credential`：必須フィールドを持つ認証情報レコードを作成します。
   - `connection`：必須フィールドを持つ接続レコードを作成します。`extended_attributes` 子オブジェクトを使用して接続属性にアクセスします。たとえば、

```
"connection": {
        "extended_attributes": {
           "api_version": "v1"

        },
     "connection_url": "https://<provider-domain-name>.com",
     "name": "Spoke Connection",
     "table": "http_connection"
  }
```

   `additional` オブジェクトを使用してカスタムテーブルのデータを設定し、後処理スクリプトを使用してデータをテーブルに挿入できます。

注: OAuth 認証コード権限許可タイプのデフォルトのデータテンプレートでは、oauth_entity_profile_scope キーと oauth_entity_scope キーの値が一致する必要があります。次の例では、両方のキーが [Read user's email] 値を持ちます。

```
"oauth_entity_profile": [
      {
        "grant_type": "authorization_code",
        "name": "<provider-name> Profile",
        "default": true,
        "oauth_entity_profile_scope": [
          "Read user's email"
        ]
      }
    ],
    "code_challenge_method": "S256",
    "type": "consumer",
    "oauth_entity_scope": [
      {
        "oauth_entity_scope": "Read user's email",
        "name": "email"
      }
    ],
```

6. [動的なデータスキーマ] フィールドで必要な変更を行います。

このフィールドは、統合を設定するためにユーザーが完了する必要のある情報を定義します。たとえば、ユーザー名とパスワードのキーと値のペアを追加して、ユーザー定義の値を収集することができます。。

動的なデータスキーマのフィールドには、次のプロパティが含まれます。

- name：ユーザーの入力がマッピングされるフィールドです。たとえば、ユーザー入力を接続レコードの [接続 URL] フィールドにマッピングするには、「connection.connection_url」と入力します。
- label：テンプレートの完成時にユーザーに表示されるフィールドラベルです。
- type：フィールドタイプです。このデータタイプが、値をマッピングするフィールドのデータタイプと一致することを確認してください。
- defaultValue：オプションです。このフィールドのデフォルト値です。デフォルトが指定されていない場合は、ヒントが表示されます。
- hint：オプションです。デフォルト値がない場合に表示されるヒントテキストです。

注: OAuth JWT ベアラー権限許可タイプ認証用のテンプレートを設定する場合は、jwt_claim_validation アレイ内の単一のキーと値のペアに対するユーザー入力が必要になる場合があります。アレイ内のインデックスを参照することで、動的なデータスキーマ内の単一のキーと値のペアを参照できます。たとえば、デフォルトのデータテンプレートにはこのスニペットが含まれている場合があります。

```
"jwt_claim_validation" : [ {
      "name" : "iss",
      "is_standard" : true,
      "data_type" : "string",
      "value":"<docusign-iss-claim>"
    }, {
      "name" : "sub",
      "is_standard" : true,
      "data_type" : "string",
      "value":"<docusign-sub-claim>"
    }, {
      "name" : "aud",
      "is_standard" : true,
      "data_type" : "string",
      "value":"<docusign-aud-claim>"
    }, {
      "name" : "scope",
      "is_standard" : false,
      "data_type" : "string",
      "value" : "signature impersonation"
    } ],
```

アイテムのゼロベースのインデックス (credential.jwt_provider.jwt_claim_validation[0].value) を使用して、iss キーと値のペアを参照します。

7. オプション: [後処理スクリプト] フィールドに、統合によって必要とされる追加のレコードを作成します。たとえば、スポークにカスタムテーブルが含まれている場合は、構成テンプレートのユーザー入力に基づいてそれらのテーブルにレコードを作成できます。このスクリプトは、接続と資格情報レコードが作成された後に実行されます。のスクリプトを追加します
後処理スクリプトは、次のグローバルオブジェクトにアクセスできます。

グローバルオブジェクト	説明
aliasId	接続および資格情報エイリアス [sys_alias] テーブルのエイリアスレコードの sys_id
connectionSysId	テンプレートによって作成された接続レコードの sys_id
jsonDefaultData	文字列形式の [デフォルトのデータテンプレート] フィールドの JSON コンテンツ
jsonDynamicData	文字列形式の [動的データテンプレート] フィールドの JSON コンテンツ

8. [事前編集スクリプト **(Pre-Edit Script)**] フィールドで、接続を編集するときの追加のフィールドを事前入力するスクリプトを追加します。

このスクリプトではオブジェクトのアレイが返されます。オブジェクトごとに、追加フィールドを設定するための名前と値のペアがあります。たとえば、接続にカスタムテーブル内のフィールドが必要な場合は、そのフィールドをカスタムテーブルにマッピングできます。

事前編集スクリプトがアクセスできるグローバルオブジェクトは次のとおりです。

グローバルオブジェクト	説明
aliasId	接続および資格情報エイリアス [sys_alias] テーブルのエイリアスレコードの sys_id
connectionSysId	テンプレートによって作成された接続レコードの sys_id
jsonDefaultData	文字列形式の [デフォルトのデータテンプレート] フィールドの JSON コンテンツ
jsonDynamicData	文字列形式の [動的データテンプレート] フィールドの JSON コンテンツ

スクリプト内の各オブジェクトには、次のプロパティがあります。

- `name`：接続の値を指定するカスタムフィールドの名前。
- `value`：カスタムフィールドに入力するマッピング対象の値。関数または変数を使用するか、またはハードコーディングすることによって、フィールドをマッピングできます。

フィールドに対してサポートされているデータタイプは次のとおりです。

フィールドに対してサポートされるデータタイプ

タイプ	説明
テキスト	文字列値。
ブーリアン	選択ボックス。選択されている場合は true、選択されていない場合は false の値を示します。
番号	数値。
日付	yyyy-mm-dd 形式の日付の値。GlideDate オブジェクトを使用することもできます。
選択肢	[動的なデータスキーマ] フィールドで定義されている有効な選択肢のリスト。
参照	有効な GlideRecord。
ラジオ グループ	さまざまなフィールドのセットを含むグループ。これらのグループは、接続を編集するときのドロップダウンリストの選択肢として使用できます。ドロップダウンリストから必要なグループを選択すると、各グループのフィールドが表示されます。 たとえば、[動的なデータスキーマ] フィールドで定義された次の構造のラジオグループを考えてみましょう。 `{ "name": "radio_groups", "label": "Radio Groups", "type": "radio", "groups": [{ "name": "radio_group1",`

タイプ	説明

	<pre>"label": "Radio Group 1",
	"fields": [{
	"name": "radio_field1",
	"label": "Radio Field 1",
	"type": "text",
	"defaultValue": "efgh",
	"mandatory": true
	}]
	}, {
	"name": "radio_group2",
	"label": "Radio Group 2",
	"fields": [{
	"name": "radio_field2",
	"label": "Radio Field 2",
	"type": "text",
	"defaultValue": "abcd",
	"mandatory": true
	}],
	"default_group": true
	}</pre>
	この例では、次のコードスニペットを使用して、ラジオグループがどのようにドット連結を使用してスクリプト内で使用されているかを理解できます。
	<pre>{
	name: "radio_field.first_radio_group.radio_field1",
	value: "radio field 1"
	}, {
	name:
	"radio_field.second_radio_group.radio_field2",
	value: "radio field 2"
	}, {
	name: “radio_groups”,
	value: gr.getValue(‘radio_groups’)
	}</pre>
	ドット連結の使用方法の詳細については、「ドット連結」を参照してください。

ヒント: 接続の編集中に事前入力された値がフィールドに表示されない場合は、次に移動します: システム診断 > セッションのデバッグ > デバッグログ 問題を診断します。

9. オプション: [テストアクション] フィールドに、テストアクションの名前を入力します。テストアクションを作成すると、 ワークフロースタジオ のフロービューで統合アクションから直接接続をテストできます。テストアクションでは、アクション定義を使用して、テンプレートが現在添付されているエイリアスをテストします。詳細については、「構成テンプレートから接続エイリアスをテストするテストアクションの作成」を参照してください。
10. テンプレートを接続および資格情報エイリアスに追加します。
    a. 移動先 統合ハブ > 接続 & 認証情報 > 接続と認証情報エイリアス.
    b. スポークのエイリアスレコードを開きます。
    c. [構成テンプレート] フィールドで、ルックアップアイコンをクリックします。

d. 作成したテンプレートをリストから選択します。

e. [更新] をクリックします。

結果
ユーザーが関連する接続および資格情報エイリアスに移動して [新しい接続および資格情報の作成] を選択すると、ユーザー入力を取得するダイアログが表示されます。OAuth 認証コード権限許可タイプのテンプレートを作成した場合は、このダイアログから OAuth トークンを取得することもできます。

次のタスク
関連する接続および資格情報エイリアスに移動し、[新しい接続および資格情報の作成] を選択して、テンプレートをテストします。ダイアログが必要なデータを収集し、システムに必要なレコードを作成することを確認します。

接続の開始

接続テーブルを使用して、ターゲットホストへの基本接続、JMS 接続、JDBC 接続、または HTTP(s) 接続を設定します。

接続テーブル

接続テーブル (sys_connection) は、すべての接続テーブルの基本テーブルです。次のプロトコルの接続を設定できます。

- PowerShell および SSH 用の基本接続
- JDBC
- JMS
- HTTP(S)

接続テーブルは、接続エイリアステーブルを参照します。これにより、接続エイリアスが接続情報に結合されます。接続ごとに次の情報が記録されます。

基本接続のプロパティ

フィールド	説明
名前	接続の名前。このフィールドは、テーブル上で一意でなければなりません。
認証情報	この接続で使用する認証情報を指定します。これはオプションです。
接続エイリアス	接続エイリアスにより、実行時に接続と資格情報が解決されます。接続エイリアスごとに一度に 1 つの接続のみアクティブになります。
アクティブ	この接続をアクティブにする場合はオンにします。
ドメイン	接続が属するドメイン。

空でない場合、認証情報は有効な接続全体で一意です。

接続情報のアップグレード

- JDBC 接続 [jdbc_connection] テーブルと JMS 接続[orch_jms_ds] テーブルは、既存のオーケストレーション接続テーブルであり、接続 [sys_connection] テーブルから拡張されています。テーブルは本来、sys_metadata から拡張されています。sys_metadata 関連データは削除されます。

- テーブルは、オーケストレーション ランタイム プラグイン [com.snc.runbook_automation.runtime] から [認証情報 & 接続] プラグインに移動します。
- アップグレード プロセスでは、JDBC および JMS 接続情報が取得され、対応する接続エイリアスが作成され、対応する接続にそのエイリアスが割り当てられます。
- JDBC フィールド名の変更：
  - JDBC サーバーの名前が host に変更されました
  - データベース ポートの名前が port に変更されました
  - アップグレード時に JDBC サーバーおよびデータベースのデータが host および port に移行されました

PowerShell および SSH 用の基本接続の作成

PowerShell または Secure Shell (SSH) プロトコルを使用するカスタムアクティビティまたはアクションで使用する接続情報を設定します。

始める前に
必要なロール：admin または connection_admin

手順

1. 移動先 すべて > 認証情報 **&** 接続 > 接続.
2. [**New**] をクリックします。
3. **[PowerShell** および **SSH** 用の基本接続] を選択します。
4. フォームに入力します。

フィールド	説明
名前	接続レコードの一意の名前。
認証情報	接続を許可するために使用する認証情報レコードを選択します。
接続エイリアス	この接続に関連付けるエイリアス レコードを選択します。エイリアスを使用すると、エイリアスを使用するアクションまたはアクティビティを再設定することなく、接続レコードを更新できます。
ホスト	アクティビティまたはアクションが実行されるターゲット ホストの完全修飾ドメイン名。たとえば、host.domain.com です。
有効	この接続を有効にする場合に選択します。
ドメイン	アクティビティが実行されるドメインを決定します。フローデザイナーでは、ドメインセパレーションがサポートされていないため、このフィールドは無視されます。
Override default port (デフォルトポートに上書きする)	接続で使用されるターゲット ポート。このフィールドを空白のままにすると、デフォルトのポート値が使用されます。
MID サーバーを使用する	MID サーバー を介してターゲット ホストに接続する場合に選択します。選択した場合、[MID サーバー詳細設定] セクションでフィールドを定義します。 **注:** PowerShell には MID サーバー が必要です。

フィールド	説明
MID 選択	特定の MID サーバーまたは MID クラスターを選択するオプション。次のオプションのいずれかを選択します。 ○ **MID** サーバーの自動選択: ServiceNow インスタンスは手動入力なしで MID サーバー を選択します。 ○ 特定の **MID** サーバー: ServiceNow インスタンスは、指定した MID サーバー を使用します。 ○ 特定の **MID** クラスター: ServiceNow インスタンスは、指定した MID クラスターを使用します。 MID クラスターは、 ServiceNow インスタンスが複数の統合を処理し、統合速度を向上させることを可能にする MID サーバーのグループです。詳細については、「MID サーバークラスターの設定」を参照してください。 このフィールドは、**[MID** サーバーを使用] のチェックがオンの場合に利用可能です。 注: オーケストレーション 接続レコードではなく、統合ハブ 接続レコードが参照されていることを確認してください。
できること	MID サーバーが選択対象となるためにサポートする必要がある機能です。選択した機能をサポートする MID サーバーからアクションまたはアクティビティが実行されます。**[MID** サーバーを使用] が選択されている場合のみ表示されます。 必要な機能によって、実行時にどのMID サーバーが選択されるかが決定されます。実行時に MID サーバー が選択される方法の詳細については、「MID サーバーの選択」を参照してください。このフィールドは、MID 選択リストから **[MID** サーバーの自動選択] が選択されている場合にのみ表示されます。
MID アプリケーション	MID サーバーが選択対象となるためにサポートする必要があるアプリケーションです。選択したアプリケーションをサポートする MID サーバーからアクションが実行されます。**[MID** サーバーを使用] が選択されている場合のみ表示されます。 実行時にMID サーバーがどのように選択されるかについての詳細は、「MID サーバーの選択」を参照してください。
MID サーバー	ステップが実行される特定の MID サーバー です。このフィールドは、MID 選択リストから [特定の **MID** サーバー] が選択されている場合にのみ表示されます。
MID クラスター	使用する特定の MID クラスター。このフィールドは、**[MID** サーバーを使用] チェックボックスがオンになっていて、MID 選択リストから [特定の **MID** クラスター] が選択されている場合に利用可能です。

5. [送信] をクリックします。

HTTP(S) 接続を作成

HTTP(S) 接続は、カスタム HTTP(S) アクションまたはアクティビティが接続に使用する情報を提供します。

始める前に

必要なロール：connection_admin

手順

1. 移動先 すべて > 認証情報 **&** 接続 > 接続をクリックし、[新規] をクリックして **[HTTP(s)** 接続] を選択します。
2. 次の接続情報を追加して [送信] をクリックします。

フィールド	説明
名前	この HTTP(S) 接続の一意の名前。
認証情報	接続を許可するために使用する認証情報レコードを選択します。
接続エイリアス	この接続に関連付けるエイリアス レコードを選択します。エイリアスを使用すると、エイリアスを使用するアクションまたはアクティビティを再設定することなく、接続レコードを更新できます。
URL ビルダー	手動で接続 URL を入力するか、システムを使用して入力に基づいて URL を作成します。デフォルトはオフです。オンにすると、次のフィールドから接続 URL が計算されます。 ○ [相互認証] - 相互認証を使用する場合はオンにします。 ○ [プロトコル] - 相互認証を使用しない場合は、プロトコルを入力します。デフォルトは HTTPs です。 ○ [プロトコル プロファイル] - 相互認証を使用する場合は、sys_protocol_profile からプロトコル プロファイルを入力します。 ○ ホスト ○ ポート ○ ベースパス - 接続文字列のパス。 **注:** 相互認証をオンにした場合は、接続 URL が作成されます (プロトコル + :// + host:port +URL)。相互認証をオフにした場合は、接続 URL が作成されます (プロトコル プロファイル + :// + host:port +URL)。
接続 URL	URL ビルダーをオフにした場合は、このフィールドに接続 URL を入力します。 **注:** 相互認証をオンにした場合は、接続 URL が作成されます (プロトコル + :// + host:port +URL)。相互認証をオフにした場合は、接続 URL が作成されます (プロトコル プロファイル + :// + host:port +URL)。
アクティブ	この接続を有効にする場合にオンにします。
ドメイン	アクションまたはアクティビティが実行されるドメインを決定します。

自動翻訳

フィールド	説明
MID サーバーを使用	このアクションまたはアクティビティで MID サーバーを使用するには、オンにします。選択した場合、[MID サーバー詳細設定] セクションでフィールドを定義します。
MID 選択	特定の MID サーバーまたは MID クラスターを選択するオプション。次のオプションのいずれかを選択します。 ○ **MID** サーバーの自動選択: ServiceNow インスタンスは手動入力なしで MID サーバー を選択します。 ○ 特定の **MID** サーバー: ServiceNow インスタンスは、指定した MID サーバー を使用します。 ○ 特定の **MID** クラスター: ServiceNow インスタンスは、指定した MID クラスターを使用します。 MID クラスターは、 ServiceNow インスタンスが複数の統合を処理し、統合速度を向上させることを可能にする MID サーバーのグループです。詳細については、「MID サーバークラスターの設定」を参照してください。 このフィールドは、**[MID** サーバーを使用] のチェックがオンの場合に利用可能です。 注: オーケストレーション 接続レコードではなく、統合ハブ接続レコードが参照されていることを確認してください。
できること	MID サーバーが選択対象となるためにサポートする必要がある機能です。選択した機能をサポートする MID サーバーからアクションまたはアクティビティが実行されます。**[MID** サーバーを使用] が選択されている場合のみ表示されます。 必要な機能によって、実行時にどのMID サーバーが選択されるかが決定されます。実行時に MID サーバー が選択される方法の詳細については、「MID サーバーの選択」を参照してください。このフィールドは、MID 選択リストから **[MID** サーバーの自動選択] が選択されている場合にのみ表示されます。
MID アプリケーション	MID サーバーが選択対象となるためにサポートする必要があるアプリケーションです。選択したアプリケーションをサポートする MID サーバーからアクションが実行されます。**[MID** サーバーを使用] が選択されている場合のみ表示されます。 実行時にMID サーバーがどのように選択されるかについての詳細は、「MID サーバーの選択」を参照してください。
MID サーバー	ステップが実行される特定の MID サーバー です。このフィールドは、MID 選択リストから [特定の **MID** サーバー] が選択されている場合にのみ表示されます。
MID クラスター	使用する特定の MID クラスター。このフィールドは、**[MID** サーバーを使用] チェックボックスがオンになっていて、MID 選択リストから [特定の **MID** クラスター] が選択されている場合に利用可能です。
接続タイムアウト	システムがホスト接続の成功を待機するミリ秒数。この間に接続が成功しない場合、接続要求はタイムアウトします。システムのデフォ

自動翻訳

フィールド	説明
	ルトの接続タイムアウト値を使用するには、このフィールドを空のままにします。 注: 接続タイムアウト値をゼロに設定しないでください。接続が古くなる可能性があるためです。
additional_http_headers ([属性] タブ)	必要な追加のヘッダー (sap-client=100 など) を含めます。複数のヘッダーを追加する場合は、セミコロンで区切ります。
odata_ping_url ([属性] タブ)	デフォルトの OData ping URL は /sap/bc/ping です。OData ハートビートの ping URL を編集するには、ロック アイコンを選択し、URL を編集して、ロック アイコンをもう一度選択します。

3. [**Submit** (送信)] を選択します。
これで、カスタム HTTP(S) アクションまたはアクティビティを作成する準備ができました。

JDBC 接続を作成

JDBC 接続は、カスタム JDBC アクションまたはアクティビティがさまざまなターゲット データベースへの接続に使用する情報を提供します。

始める前に

インスタンスに付属するかどうかに関係なく、適切な JAR ファイル、またはカスタム JAR ファイルが必要です。

注: ServiceNow インスタンスでは、mysql-connector-java-5.1.21.jar、sql-server-jdbc-4.0.jar、および ojdbc6.jar の各ファイルが現在のリリースの一部として提供され、MySQL、SQLServer、および Oracle データベースをサポートしています。Sybase や DB2 Universal などの他のデータベースでは、カスタム JAR ファイルを使用する必要があります。このファイルは、JDBC 接続を設定する前にインスタンスにアップロードする必要があります。

必要なロール：connection_admin

このタスクについて

JDBC 認証情報は、アクティビティデザイナーテンプレートによって別々に取得され、CyberArk などの外部認証情報ストレージをサポートします。

手順

1. 移動先 すべて > 認証情報 **&** 接続 > 接続をクリックし、[新規] をクリックして **[JDBC** 接続] を選択します。
2. テーブル内のフィールド使用して、フォームに値を入力します。
[フォーマット] フィールドのデータベース選択により、利用可能なフィールドが決定されます。

JDBC 接続フィールド

フィールド	データベース形式	説明
名前	すべて	この JDBC 接続の一意の名前。たとえば、**JDBC MySQLProd** を入力します。
認証情報	すべて	JDBC プロバイダーの認証情報を追加します。
接続エイリアス	すべて	この接続に関連付けるエイリアス レコードを選択します。エイリアスを使用すると、エイリアスを使用するア

フィールド	データベース形式	説明
		クションまたはアクティビティを再設定することなく、接続レコードを更新できます。
クエリー タイムアウト	すべて	JDBC クエリが応答なしで実行される最大経過時間。
接続タイムアウト	すべて	JDBC 接続または接続要求がクローズされるまでシステムが待機する秒数。 たとえば、[接続タイムアウト] の値が 10 秒の場合、システムは接続が成功するまで 10 秒間待機します。この間に接続が成功しない場合、接続要求はタイムアウトします。接続が成功した場合、非アクティブ状態が 10 秒間続くまで、接続は開いたままになります。接続が 10 秒間非アクティブになると、接続はクローズされます。 **注:** 接続タイムアウト値をゼロに設定しないでください。接続が古くなる可能性があるためです。
アクティブ	すべて	これを有効な接続にするには、オンにします。
ドメイン	すべて	このテーブルのドメイン。デフォルトでは、グローバルドメインで JDBC 接続 [jdbc_connection] テーブルが実行されます。
フォーマット	すべて	この接続のデータベース タイプ。デフォルトの選択肢は次のとおりです。 ○ **MySQL** ○ **Oracle** ○ **SQLServer** ○ なし 適切な JDBC ドライバー JAR ファイルをインスタンスにアップロードすることで、**[Sybase]** または **[DB2 Universal]** を選択リストに追加できます。オーケストレーション は、これらのドライバーがシステムにロードされると自動的に認識し、このリストに追加します。
ホスト	Oracle、MySQL、SQLServer	データベース サーバーのホスト名または IP アドレス。
Oracle SID	Oracle	Oracle データベース サイトの識別子。デフォルト値は **orcl** です。
Oracle ポート	Oracle	Oracle データベースが使用しているポート。デフォルト値は **1521** です。
データベース名	MySQL、SQLServer	データベースの名前。
ポート	MySQL、SQLServer	選択したデータベースが使用しているポート。
インスタンス名	SQLServer	選択した SQLServer のインスタンス名
接続 URL	すべて	指定したデータベースへの接続に MID サーバーが使用する URL。この URL は、フォームを保存したときに自動的に作成され、デフォルトのデータベースでは読み取り専用になります。

フィールド	データベース形式	説明
		注: 選択した形式がデフォルトのデータベースのものでない場合は、接続 URL を手動で作成して、MID サーバーが接続の作成方法を認識できるようにする必要があります。
JDBC ドライバー	なし、DB2 Universal、Sybase	デフォルトのデータベースでない場合にこの接続に使用する JDBC ドライバー。 注: Sybase または DB2 Universal Database を追加する場合、このフィールドにドライバー名を入力して、そのドライバー JAR ファイルをインスタンスにアップロードする必要があります。
MID サーバーを使用	すべて	このアクションまたはアクティビティで MID サーバーを使用するには、オンにします。選択した場合、[MID サーバー詳細設定] セクションでフィールドを定義します。
MID 選択	すべて	特定の MID サーバーまたは MID クラスターを選択するオプション。次のオプションのいずれかを選択します。 ○ **MID** サーバーの自動選択: ServiceNow インスタンスは手動入力なしで MID サーバー を選択します。 ○ 特定の **MID** サーバー: ServiceNow インスタンスは、指定した MID サーバー を使用します。 ○ 特定の **MID** クラスター: ServiceNow インスタンスは、指定した MID クラスターを使用します。 MID クラスターは、 ServiceNow インスタンスが複数の統合を処理し、統合速度を向上させることを可能にする MID サーバーのグループです。詳細については、「MID サーバークラスターの設定」を参照してください。 このフィールドは、**[MID** サーバーを使用] のチェックがオンの場合に利用可能です。 注: オーケストレーション 接続レコードではなく、統合ハブ 接続レコードが参照されていることを確認してください。
できること	すべて	MID サーバーが選択対象となるためにサポートする必要がある機能です。選択した機能をサポートする MID サーバーからアクションまたはアクティビティが実行されます。**[MID** サーバーを使用] が選択されている場合のみ表示されます。 必要な機能によって、実行時にどのMID サーバーが選択されるかが決定されます。実行時に MID サーバー が選択される方法の詳細については、「MID サーバーの選択」を参照してください。このフィールドは、MID 選択リストから **[MID** サーバーの自動選択] が選択されている場合にのみ表示されます。

自動翻訳

フィールド	データベース形式	説明
MID アプリケーション	すべて	MID サーバーが選択対象となるためにサポートする必要があるアプリケーションです。選択したアプリケーションをサポートする MID サーバーからアクションが実行されます。**[MID** サーバーを使用] が選択されている場合のみ表示されます。 実行時にMID サーバーがどのように選択されるかについての詳細は、「MID サーバーの選択」を参照してください。
MID サーバー	すべて	ステップが実行される特定の MID サーバー です。このフィールドは、MID 選択リストから [特定の **MID** サーバー] が選択されている場合にのみ表示されます。
MID クラスター	すべて	使用する特定の MID クラスター。このフィールドは、**[MID** サーバーを使用] チェックボックスがオンになっていて、MID 選択リストから [特定の **MID** クラスター] が選択されている場合に利用可能です。

3. [送信] をクリックします。

関連トピック

JDBC 認証情報

JMS 接続の作成

カスタム JMS アクティビティまたはアクションで Java Messaging Service (JMS) を使用するようにシステムを設定します。

始める前に

必要なロール：connection_admin

このタスクについて

MID サーバーには、組織に適切な JMS 接続ファクトリーが必要です。これらの値を、次にある *mid.property.jms.command.allowed_factory_names* プロパティで構成します。 **MID** サーバー > プロパティ. このプロパティのデフォルト値は、サードパーティの JMS プロバイダーが通知する任意の値またはカンマ区切りの値リストに変更できます。

手順

1. 移動先 認証情報 **&** 接続 > 接続.
2. [新規]をクリックして **[JMS** 接続] を選択し、フォームに入力して [送信] をクリックします。

オプション	説明
名前	この接続ファクトリーの一意の名前。
認証情報	JMS プロバイダーの認証情報を追加します。
接続エイリアス	この接続に関連付けるエイリアス レコードを選択します。エイリアスを使用すると、エイリアスを使用するアクションまたはアクティビティ

オプション	説明
	を再設定することなく、接続レコードを更新できます。
初期コンテキスト ファクトリー	InitialContext の作成に使用する JNDI クラスの名前。 **注:** たとえば、ActiveMQ V5.10 (JMS プロバイダー) に接続する場合、値は org.apache.activemq.jndi.ActiveMQInitialContextFactory です。
プロバイダー URL	実行中の JMS プロバイダーのインストールの場所。 **注:** たとえば、ActiveMQ V5.1 に接続するには、tcp：// ipAddressOrHostName：61616になります。
有効	これを有効な接続にするには、オンにします。
ドメイン	アクションまたはアクティビティが実行されるドメインを決定します。
MID サーバーを使用	このアクションまたはアクティビティで MID サーバーを使用するには、オンにします。選択した場合、[MID サーバー詳細設定] セクションでフィールドを定義します。
MID 選択	特定の MID サーバーまたは MID クラスターを選択するオプション。次のオプションのいずれかを選択します。 ○ **MID** サーバーの自動選択: ServiceNow インスタンスは手動入力なしで MID サーバー を選択します。 ○ 特定の **MID** サーバー: ServiceNow インスタンスは、指定した MID サーバー を使用します。 ○ 特定の **MID** クラスター: ServiceNow インスタンスは、指定した MID クラスターを使用します。 MID クラスターは、 ServiceNow インスタンスが複数の統合を処理し、統合速度を向上させることを可能にする MID サーバーのグループです。詳細については、「MID サーバークラスターの設定」を参照してください。 このフィールドは、**[MID** サーバーを使用] のチェックがオンの場合に利用可能です。
MID クラスター	使用する特定の MID クラスター。このフィールドは、**[MID** サーバーを使用] チェックボックスがオンになっていて、MID 選択リストから

オプション	説明
	[特定の **MID** クラスター] が選択されている場合に利用可能です。
機能	MID サーバーが選択対象となるためにサポートする必要がある機能です。選択した機能をサポートする MID サーバーからアクションまたはアクティビティが実行されます。**[MID** サーバーを使用] が選択されている場合のみ表示されます。 実行時にMID サーバーがどのように選択されるかについての詳細は、「MID サーバーの選択」を参照してください。
MID アプリケーション	MID サーバーが選択対象となるためにサポートする必要があるアプリケーションです。選択したアプリケーションをサポートする MID サーバーからアクションが実行されます。**[MID** サーバーを使用] が選択されている場合のみ表示されます。 実行時にMID サーバーがどのように選択されるかについての詳細は、「MID サーバーの選択」を参照してください。

3. 移動先 接続 **&** 認証情報 > 認証情報.
4. [新規]をクリックして **[JMS** 認証情報] を選択し、 MID で JMS プロバイダーとの通信に使用されるユーザー名とパスワードを入力します。
   詳細については、「JMS 認証情報」を参照してください。
5. [送信] をクリックします。
   これで、カスタム JMS アクションまたはアクティビティを作成する準備ができました。

統合ハブの接続属性を作成

統合ハブ統合ステップで使用できる接続固有の変数を定義します。

始める前に

必要なロール：

- 接続属性を作成するには、admin ロールが必要です。
- 属性値を割り当てるには、connection_admin または admin ロールが必要です。
- カスタム アクションで属性を使用するには、action_designer または admin ロールが必要です。

接続属性は、統合ステップでのみ使用され、統合ハブ に対するサブスクリプションが必要です。統合ハブのアクティブ化の詳細については、次を参照してください。 の要求 統合ハブ プラグイン .

このタスクについて

統合ステップを使用する場合は、外部システムとの接続を確立する必要があります。接続をインラインで定義する代わりに、接続および資格情報エイリアスを使用します。エイリアスを使用すると、すべてのアクションを再設定することなく、接続の詳細を 1 回で更新できます。エイリアスを使用するアクション ステップは、エイリアスに関連付けられた属性を継承します。フローデザイ

ナーには、アクション ステップにドラッグできるデータ ピルとして属性が表示されます。たとえば、REST ステップのクエリー パラメーターになるページ サイズ属性を作成できます。

カスタム フローデザイナー アクションのビルドの詳細については、以下を参照してください。 アクションデザイナー .

手順

1. 移動先 すべて > 認証情報 & 接続 > 接続と認証情報エイリアス.
2. エイリアス レコードを作成または選択します。
3. 接続属性関連リストから、[新規] をクリックします。
4. 属性ラベルとフィールド タイプを定義します。
   フィールドタイプのリストについては、「フィールドタイプ 」を参照してください。
5. [詳細ビュー] 関連リンクをクリックして、属性のディクショナリー詳細設定を設定します。
   たとえば、動的に計算された値を持つ属性を作成するには、「辞書エントリフォーム 」を参照してください。
6. [送信] をクリックします。
7. 接続レコードで属性値を定義します。

   a. 移動先 認証情報 & 接続 > 接続.

   b. エイリアスと同じ接続タイプの接続レコードを作成または選択します。

   c. [接続エイリアス] で、接続属性を持つエイリアスを選択します。

   d. レコードを保存します。
   [属性] タブに、エイリアス レコードで定義された接続属性が入力されます。

   e. 属性の値を設定します。
   エイリアスで [複数のアクティブな接続をサポート] が有効になっている場合、複数の接続レコードをエイリアスに関連付けて、接続レコードごとに属性値を定義することができます。同じエイリアスの属性値を持つ接続レコードが複数ある場合は、 フローが実行されるときに使用される接続によって属性値が決まります。たとえば、production と test という 2 つのアクティブ接続エンドポイントを持つエイリアスを使用するアクションが 1 つあるとします。この属性は、実行時に使用される接続によって定義された値に解決されます。

8. アクションデザイナー の統合ステップにエイリアスを追加します。

   a. フローデザイナー に移動して、アクションを作成または選択します。

   b. <アクションに統合ステップを追加します。

   c. [接続の詳細] で、属性を作成したエイリアスを追加します。

   エイリアスに関連付けられている接続属性が、[データ] ペインにデータ ピルとして表示されます。

   注: エイリアスをステップに関連付けると、接続属性ラベルおよびデータタイプの変更は追跡されません。接続属性ラベルまたはデータタイプをリフレッシュするには、ステップからエイリアスを削除して再度追加します。

認証情報の使用を開始する

MID サーバーでは、認証情報 [discovery_credentials] テーブルで作成した認証情報を使用して、ディスカバリー、オーケストレーション、サービスマッピング、およびクラウド管理のリソースにアクセスします。

MID サーバーが認証情報を使用する仕組み

デフォルトでは、Windows MID サーバーは、ホストマシン上の MID サーバーサービスのログイン認証情報を使用して、ネットワーク内の Windows デバイスを検出します。Windows MID サーバーサービス認証情報を設定 して、最低限ローカルアドミニストレーター権限を持つようにする必要があります。Linux および UNIX のマシンとネットワークデバイスの場合、MID サーバーは次のインスタンスで設定された SSH および SNMP 認証情報を使用します。 ディスカバリー > 認証情報.

オーケストレーション で使用される MID サーバーには、[ワークフローアクティビティ] で指定されているとおり、ネットワーク内のコンピューター上でコマンドを実行するために必要な認証情報へのアクセス権が必要です。オーケストレーションでは、ディスカバリー と同じ SSH および SNMP 認証情報を使用できますが、特定のワークフローアクティビティ用に設計された 2 つの追加認証情報：Windows (PowerShell アクティビティ 用) と VMware があります。

暗号化と復号化

プラットフォームでは、認証情報 [discovery_credentials] テーブルの暗号化フィールドに認証情報を保存します。認証情報が入力されると、認証情報は表示できません。

MID サーバーによって認証情報が要求されると、ServiceNow AI Platform で次のプロセスを使用して認証情報が復号化されます。

1. password2 固定キーを使用してインスタンスに対して認証情報が復号化されます。
2. 認証情報は、MID サーバーの公開鍵を使用してインスタンス上で再暗号化されます。
3. 認証情報は、SSL を使用してロード バランサー上で暗号化されます。
4. 認証情報は、SSL を使用して MID サーバー上で復号化されます。
5. 認証情報は、MID サーバーの秘密鍵を使用して MID サーバー上で復号化されます。

注: プラットフォームには、マルチテナント インスタンス用の個別の暗号鍵はありません。

認証情報の順序

認証情報には、 [Credentials Form] (認証情報フォーム) の順序値を割り当てることができます。これにより、特定の順序ですべての認証情報をアプリケーションで自由に試行させることができます。順序値を指定しない場合、アプリケーションは、有効な認証情報が見つかるまで、認証情報 [discovery_credential] テーブルの認証情報をランダムに試行します。たとえば、次の場合です。

- オーケストレーションは、Linux マシンや UNIX マシンなどの SSH サーバーでコマンドを実行しようとします。
- ディスカバリーは、プリンター、ルーター、UPS などの SNMP デバイスのクエリーを試行します。

デバイスの認証情報を特定した後、ディスカバリー とオーケストレーションでは、認証情報親和性 [dscy_credentials_affinity] テーブルを使用して認証情報とデバイス間の親和性が作成されます。後続のすべての検出またはオーケストレーション アクティビティで、このテーブルの認証情報が、親和性が存在するデバイスと照合されます。デバイスの認証情報が変更された場合、 ディスカバリー とオーケストレーションでは、新しい親和性を作成するまで利用可能な認証情報をすべて再試行します。

注: オーケストレーションと ディスカバリー がインストールされ、認証情報エイリアスが有効になっている場合、複数の親和性が存在する可能性があります。この場合、プラットフォームでは、親和性ごとに認証情報が検索され、順序が最も低い親和性の認証情報がプローブに挿入されます。

認証情報の順序付けは、次の場合に役立ちます。

- 認証情報テーブルには多くの認証情報が含まれており、一部の認証情報が他の認証情報よりも頻繁に使用される場合があります。たとえば、テーブルに 150 個の SSH 認証情報が含まれていて、そのうちの 5 個がデバイスの 90% へのログインに使用されている場合は、その 5 個の認証情報の順序値を小さくすることをお勧めします。これにより、その 5 個が実行リストの最上部に配置されます。このようにして、よく使用する認証情報が最初に試行されると、ディスカバリー およびオーケストレーションの動作がより迅速になります。最初の接続に成功した後、ServiceNow AI Platform によりデバイスごとに次回使用する認証情報が認識されます。
- ServiceNow AI Platform には、積極的なログインセキュリティがあります。たとえば、ネットワーク内の Solaris データベースサーバーが、ログイン試行を 3 回失敗すると MID サーバーからロックアウトされる場合は、データベース認証情報の順序値を小さくします。

認証情報エイリアス

認証情報エイリアスは ディスカバリーとオーケストレーションで使用できます。

ディスカバリーにエイリアスを使用すると、アドミニストレーターは以下の操作が可能です。

- 設定可能なコンプライアンスレベルで認証情報フィルタリング動作を使用します。
- 複数の認証情報エイリアスをディスカバリースケジュールに割り当てます。
- 不適切な認証情報や機密の認証情報を使用する認証情報親和性の作成を防止します。詳細については、「認証情報親和性」を参照してください。

オーケストレーションにエイリアスを使用すると、ワークフロー作成者は以下の操作が可能です。

- 個々の認証情報をオーケストレーション ワークフローの任意のアクティビティに割り当てる。
- 個々の認証情報をフローデザイナーの任意のアクションに割り当てる。
- オーケストレーション ワークフローで同じアクティビティ タイプが発生するたびに異なる認証情報を割り当てる。
- デザイナー フローで同じアクションが発生するたびに異なる認証情報を割り当てる。

外部の認証情報ストア

インスタンスに認証情報を保存しない場合は、外部の認証情報リポジトリを使用できます。外部の認証情報ストアにより、インスタンスによってアクセスできる外部サイトに認証情報が保存されます。CyberArk は、サポートされている唯一の外部認証情報ストアです。ただし、ServiceNow API を使用して他の外部ストアを構成することもできます。

接続情報および認証情報エイリアスの作成

認証情報または接続情報レコードにラベルを付けるエイリアスを定義します。

始める前に

必要なロール：

- エイリアスを作成するには、admin ロールが必要です。
- credential_admin ロールと connection_admin ロールには、エイリアス レコードへの読み取りアクセス権があります。

このタスクについて
接続および資格情報エイリアスは、認証情報または接続レコードにラベルを付けるエイリアスを定義します。エイリアスには次のフィールドが含まれています。

手順

1. 移動先 すべて > 接続 & 認証情報 > 接続と認証情報エイリアス.
2. [**New**] をクリックします。
3. フォームのフィールドに入力します。

接続および資格情報エイリアス

フィールド	説明
名前	エイリアスの名前。エイリアスには、英数字とアンダースコアだけを使用できます。 アップグレード時に、認証情報レコードのタグが接続および資格情報エイリアスに移行します。認証情報タグに英数字とアンダースコア以外の特殊文字が含まれている場合、タグ名はアップグレード後も保持されます。この移行されたエイリアスは引き続き使用できますが、名前を変更して命名の制限を満たすまでエイリアスを更新することはできません。
ID	形式 scope_name.alias_name に基づく、接続および資格情報エイリアスの一意の識別子です。 ○ スコープがグローバルである場合、ID はエイリアス名です。たとえば、グローバルスコープで作業日エイリアスを作成すると、ID は workday に設定されます。 ○ HR アプリケーションスコープで作業日エイリアスを作成すると、ID は x_hr_app.workday に設定されます。
タイプ	[認証情報] または [接続と資格情報] を選択します。デフォルトは [接続と資格情報] です。
アプリケーション	接続および資格情報エイリアスの割り当て対象のアプリケーションスコープです。アプリケーションピッカーで最後に選択した現在のセッションスコープが表示されます。 ○ たとえば、[グローバル] がこのセッションの現在のスコープである場合はそのように表示されます。 ○ エイリアスを作成する前に、アプリケーションピッカーでスコープを変更できます。アプリケーションスコープおよびその選択方法の詳細については、以下を参照してください。 ▪ アプリケーションスコープ ▪ アプリケーション ピッカーからのアプリケーションの選択
接続タイプ	接続タイプの名前 (基本、HTTP、JDBC、JMS、Kafka のいずれか) です。デフォルトは [HTTP] です。
複数の有効な接続をサポート	エイリアスが複数の有効な接続をサポートするかどうかを指定する指定子です。接続テーブルを使用して接続を追加し、接続関連リストを使用してその接続をエイリアスに関連付けます。
デフォルトの再試行ポリシー	エイリアスの再試行ポリシーです。詳細については、「再試行ポリシー」を参照してください。
構成テンプレート	接続および認証情報レコードの作成に使用する構成テンプレートです。

4. [保存] をクリックします。
   接続と接続属性関連リストが表示されます。

関連リスト	説明
接続	このエイリアスに関連付けられた関連接続レコード。エイリアスを作成した後、接続レコードを定義してエイリアスに関連付けることができます。[複数の有効な接続をサポート] を選択した場合、複数の接続をエイリアスに関連付けることができます。
接続属性	接続の属性です。接続に固有のデータを定義し、 統合ハブ統合ステップでそのデータを使用します。詳細については、「統合ハブの接続属性を作成」を参照してください。
子エイリアス	親エイリアスに関連付けられた子エイリアスです。接続および資格情報エイリアスを作成した後、子エイリアスを作成して、同じアプリケーションデータ連携のために複数の接続を構成できます。

関連リスト	説明
接続	このエイリアスに関連付けられた関連接続レコード。エイリアスを作成した後、接続レコードを定義してエイリアスに関連付けることができます。[複数の有効な接続をサポート] を選択した場合、複数の接続をエイリアスに関連付けることができます。
接続属性	接続の属性です。接続に固有のデータを定義し、 統合ハブ統合ステップでそのデータを使用します。詳細については、「統合ハブの接続属性を作成」を参照してください。

5. オプション: 認証情報エイリアスに関連付けられた新しい認証情報および接続を作成する場合は、*Related Links* で [新しい接続と資格情報の作成] をクリックします。
   結果の接続および認証情報レコードは、事前定義された構成テンプレートに基づいています。「接続と資格情報の構成テンプレート」を参照してください。
6. オプション: 接続および資格情報エイリアスの子エイリアスを作成する場合は、[子エイリアス] 関連リストで [新規] を選択します。

   a. 子エイリアスの名前を入力し、[送信] を選択します。
      子エイリアスは、親エイリアスからプロパティを継承します。その後、子エイリアスを設定して、独自の接続および認証情報のセットを含めることができます。

次のタスク

エイリアスまたは子エイリアスに関連付ける 1 つ以上の接続レコードを作成します。接続の作成に関する詳細については、「接続の開始」を参照してください。接続属性をエイリアスに追加して、接続メタデータを フローデザイナー のフローで利用できるようにします。

MID サーバーを介した OAuth 統合の設定

OAuth トークン要求を MID サーバー経由でサードパーティサーバーに送信できるようにする接続レコードを作成します。

始める前に
ServiceNow 統合ハブ標準パックインストーラーに登録していることを確認します。詳細については、「https://www.servicenow.com/content/dam/servicenow-assets/public/en-us/doc-type/legal/snc-addendum-integrationhub.pdf 」を参照してください。

必要なロール：admin

手順

構成テンプレートを使用して接続を構成します。

a. 移動先 すべて > 統合ハブ > 構成テンプレート.

b. **[New** (新規)] を選択します。

c. **[OAuth** クライアント認証情報権限許可タイプを使用した **HTTP** 接続] を選択します。

d. 必要に応じて、フォームを更新します。
たとえば、[デフォルトのデータテンプレート] セクションの oauth_entity_profile_scope フィールドと oauth_entity_scope フィールドにスコープを指定できます。構成テンプレートの作成の詳細については、「構成テンプレートの作成」を参照してください。

e. [送信] を選択します。
構成テンプレートが作成されました。

f. 移動先 すべて > 統合ハブ > 接続と認証情報エイリアス.

g. フォームを更新します。

接続および資格情報エイリアス

フィールド	説明
名前	エイリアスの名前。エイリアスには、英数字とアンダースコアだけを使用できます。 アップグレード時に、認証情報レコードのタグが接続および資格情報エイリアスに移行します。認証情報タグに英数字とアンダースコア以外の特殊文字が含まれている場合、タグ名はアップグレード後も保持されます。この移行されたエイリアスは引き続き使用できますが、名前を変更して命名の制限を満たすまでエイリアスを更新することはできません。
アプリケーション	形式 scope_name.alias_name に基づく、接続および資格情報エイリアスの一意の識別子です。

フィールド	説明
	○ スコープがグローバルである場合、ID はエイリアス名です。たとえば、グローバルスコープで作業日エイリアスを作成すると、ID は workday に設定されます。 ○ HR アプリケーションスコープで作業日エイリアスを作成すると、ID は x_hr_app.workday に設定されます。
親エイリアス	親エイリアスを選択するオプション。選択したエイリアスの下に、この接続および資格情報エイリアスを作成します。作成している接続および資格情報エイリアスは、子エイリアスになります。子エイリアスは、親「接続および資格情報エイリアス」ページの [子エイリアス] タブに一覧表示されます。 
タイプ	作成しているエイリアスのタイプを示すオプション。次のオプションから選択します。 ○ 認証情報：認証情報レコードを含むエイリアス。 ○ 接続と資格情報：接続および資格情報レコードの両方を含むエイリアス。このオプションはデフォルトで選択されています。 [接続と資格情報] が選択されていることを確認します。
複数の有効な接続をサポート	エイリアスが複数の有効な接続をサポートするかどうかを指定する指定子です。接続テーブルを使用して接続を追加し、接続関連リストを使用してその接続をエイリアスに関連付けます。
デフォルトの再試行ポリシー	エイリアスの再試行ポリシーです。詳細については、「再試行ポリシー」を参照してください。
構成テンプレート	構成テンプレートを選択するオプション。このテンプレートに基づいて接続および資格情報エイリアスを作成します。前の手順で作成した **[OAuth** クライアント認証情報権限許可タイプを使用した **HTTP** 接続] タイプのテンプレートを選択します。

h. [送信] を選択します。
接続および資格情報エイリアスレコードが作成されました。

i. 移動先 すべて > 統合ハブ > 接続ダッシュボード.

j. [すべての接続を検索] フィールドに、作成した「接続および資格情報エイリアス」レコードの名前を入力します。

k. 接続および資格情報エイリアスレコードで、[詳細を表示] を選択します。

l. [構成] を選択します。

m. フォームに入力します。

接続レコード

フィールド	説明
接続名	接続レコードの名前。名前を更新することはできません。
接続 URL	サードパーティサーバーに接続する URL を指定するオプション。
MID を使用	MID サーバー経由で OAuth トークン要求を送信することを指定するオプション。 **重要:** このオプションが選択されていることを確認します。
MID 選択	特定の MID サーバーを使用するか、MID サーバーの自動選択を有効にするか、MID クラスターを使用するかを指定するオプション。次のいずれかのオプションを選択します。 ○ MID サーバーの自動選択 ○ 特定の MID サーバー ○ 特定の MID クラスター
機能	1 つ以上の MID サーバー機能を選択するオプション。機能は、IP アドレス範囲内の MID サーバーの特定の機能を定義し、アプリケーションが最適な MID サーバーを選択できるようにします。MID 機能 + を選択して、1 つ以上の機能を選択します。 **注:** このオプションは、[MID 選択] フィールドで **[MID** サーバーの自動選択] を選択した場合に表示されます。
MID アプリケーション	MID アプリケーションを指定するか、デフォルトのアプリケーション選択を承認するオプション。 **注:** このオプションは、[MID 選択] フィールドで **[MID** サーバーの自動選択] を選択した場合に表示されます。デフォルトでは、[すべて] オプションが選択されています。

フィールド	説明
MID サーバー	MID サーバーを選択するオプション。 **注:** このオプションは、[MID 選択] フィールドで [特定の **MID** サーバー] を選択した場合に表示されます。
MID クラスター	MID クラスターを選択するオプション。 **注:** このオプションは、[MID 選択] フィールドで[特定の **MID** クラスター] を選択した場合に表示されます。
OAuth クライアント ID	クライアント ID を指定するオプション。
OAuth クライアントシークレット	クライアントシークレットを指定するオプション。
MID サーバー経由で認証サーバーに接続	ServiceNow インスタンスと認証サーバー間の接続が MID サーバー経由で行われるように指定するオプション。 **重要:** このオプションが選択されていることを確認します。
OAuth トークン URL	OAuth トークンを要求するために使用される OAuth トークン URL を指定するオプション。

n. **[OAuth** トークンを設定して取得] を選択します。
接続および認証情報レコードが作成されます。

ディスカバリーの認証情報エイリアス

ディスカバリー の認証情報エイリアスを使用すると、アドミニストレーターはディスカバリースケジュールで特定の認証情報を使用できます。システムで使用方法を厳密に適用する方法を決定する、エイリアスの動作を設定できます。

認証情報エイリアスがない場合、ディスカバリースケジュールは、インスタンスで定義されているすべての認証情報にアクセスできます。状況によっては、特に昇格した権限を持つ認証情報の場合には、この動作は望ましくありません。認証情報エイリアスを使用すると、ディスカバリースケジュールで使用できる認証情報を細かく制御し、昇格した権限を持つ認証情報が不必要に公開されないようにすることができます。

認証情報エイリアスの仕組み

ディスカバリー親和性および認証情報エイリアスの挿入 (以前の名前はディスカバリー親和性の挿入) と呼ばれるビジネスルールは、レコード (ディスカバリーを実行するためのタスク) が ECC キューに挿入されたときに実行されます。

ビジネスルールは、ディスカバリースケジュールで定義された認証情報エイリアスをプローブに添付するため、プローブがディスカバリーを実行する途中で MID サーバーに到達したときに、MID サーバーは、プローブがスキャンのために送信されたデバイスへのアクセスを試みるために使用できる認証情報を正確に認識します。

MID サーバーは、 認証情報を親和性 、次にタグ (存在する場合) でフィルタリングします。認証情報はすべての認証情報タグと一致する必要があります。MID サーバーは、有効な認証情報が見つかるまで反復処理します。

デバイスの親和性が存在するとビジネスルールが判断した場合、ルールは使用する適切な *credential_id* を識別します。これは、認証情報 *[discovery_credentials]* テーブル内のレコードの sys_id です。

認証情報エイリアス値 (ビジネスルールの *credential_alias* で定義) との親和性がプラットフォームで見つかると、ビジネスルールにより、親和性によって参照される認証情報に指定されたエイリアスがあるかどうかが判断されます。エイリアスがある場合、ビジネスルールにより、エイリアスの *credential_id* が選択され、その値が MID サーバーに渡されます。

スケジュールに資格情報エイリアスが定義され、そのエイリアスを使用するようにスケジュールが構成されている場合、スケジュールは、資格情報自体が他の資格情報エイリアスに関連付けられていない場合のみ、既存の資格情報とターゲットの親和性を無視します。認証情報に認証情報エイリアスがない場合は、ターゲット システムに存在するその他の親和性がチェックされます。

ディスカバリー認証情報エイリアスの作成

エイリアスを作成し、認証情報レコードで認証情報にそのエイリアスを追加します。複数のエイリアスに 1 つの認証情報を追加することも、1 つのエイリアスに複数の認証情報を追加することもできます。

始める前に

必要なロール：admin、credential_admin (読み込みアクセスのみ)、connection_admin (読み込みアクセスのみ)

このタスクについて

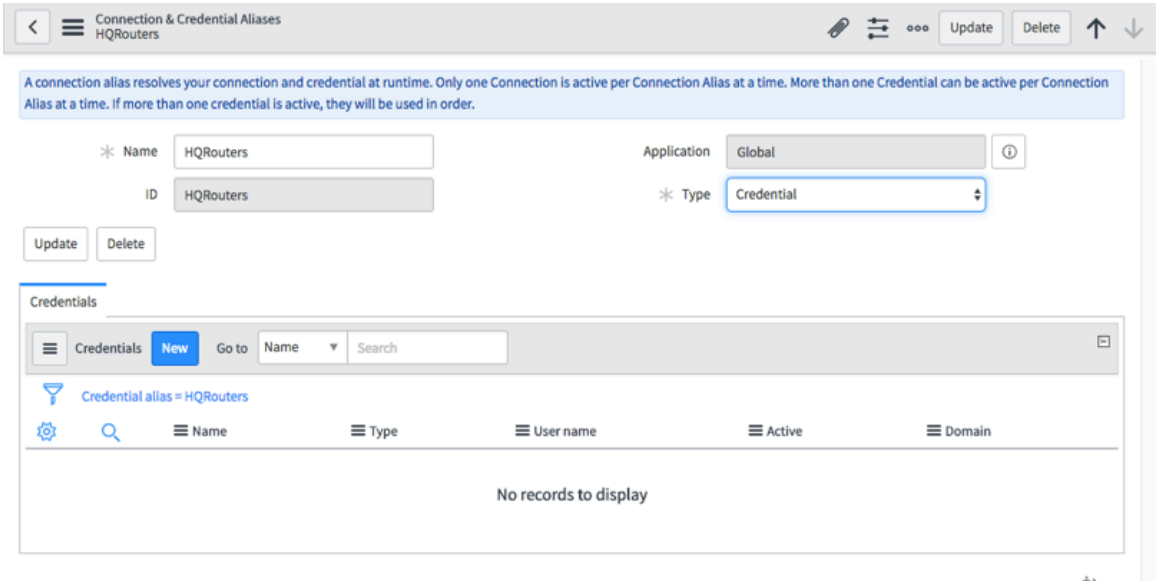
ディスカバリースケジュールは、そのスケジュール用に定義されたエイリアスに含まれている認証情報のみを使用します。

注: 認証情報エイリアスがスケジュールに定義されている場合、その認証情報エイリアスを使用するようにセットアップされたスケジュールで検出されたターゲットと認証情報間の既存の認証情報親和性は無視されます。

手順

1. エイリアスを作成します。
   a. 移動先 接続 **&** 認証情報 > 接続と認証情報エイリアス.
   b. [**New**] をクリックします。
   c. エイリアスの一意の名前を入力し、エイリアスの [タイプ] の [認証情報] を選択します。
   d. [送信] をクリックします。

[認証情報] 関連リストが表示されます。このリストでは、このエイリアスの新しい認証情報を追加できますが、既存の認証情報は追加できません。

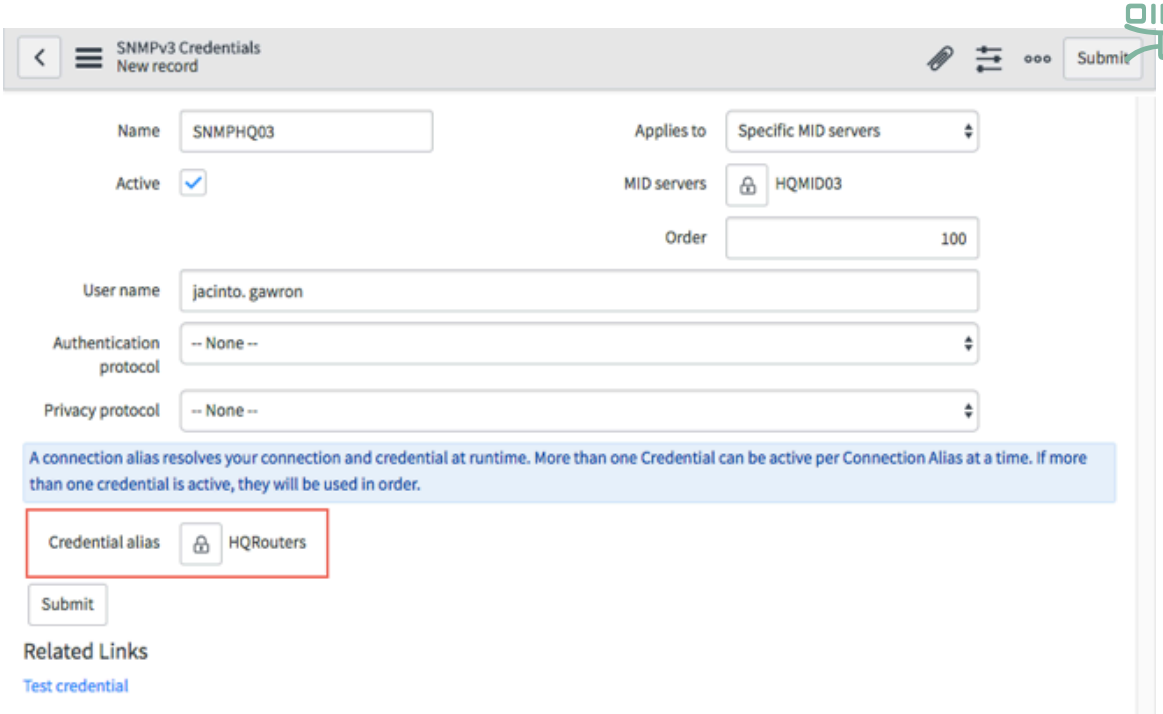


2. 新しいエイリアスの認証情報を設定します。

a. 移動先 接続 **&** 認証情報 > 認証情報.

b. リストから既存の認証情報を選択するか、[新規] をクリックして新しい認証情報を作成します。

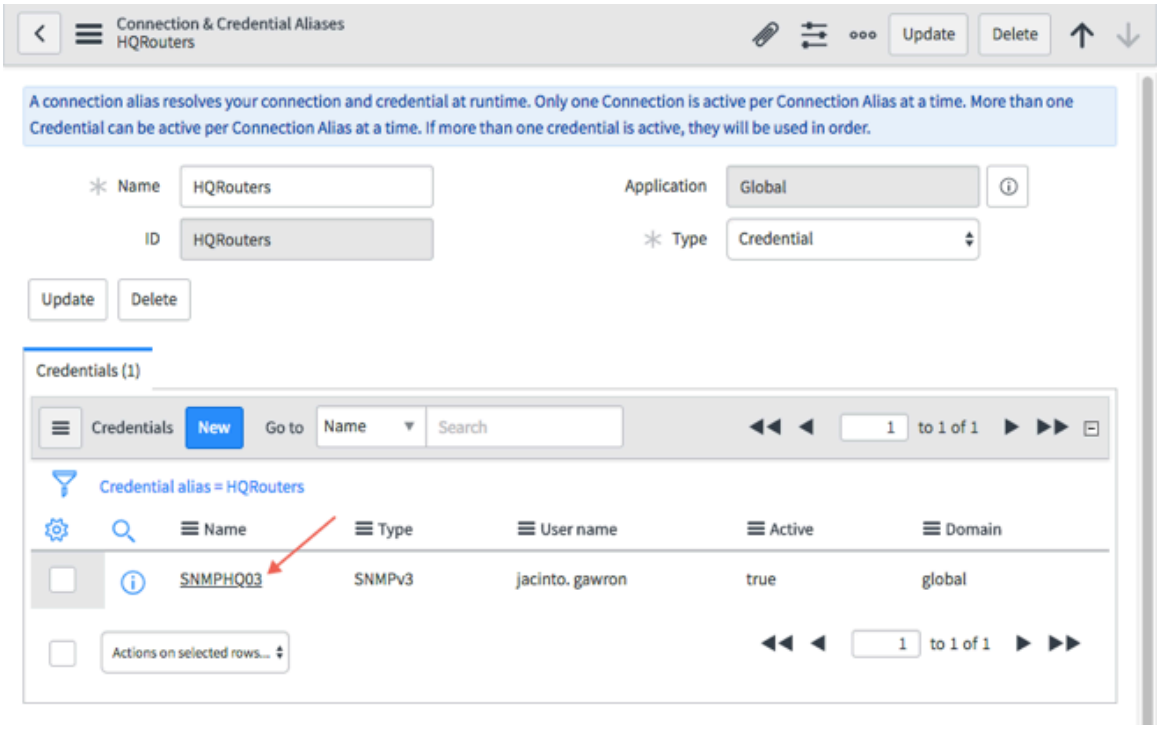
c. 認証情報レコードで、[認証情報エイリアス] フィールドのロックを解除し、作成したエイリアスを選択します。



d. レコードを保存または送信します。

3. 戻る 接続 **&** 認証情報 > 接続と認証情報エイリアス をクリックし、新しいエイリアスを開きます。

エイリアスに添付した認証情報が関連リストに表示されます。



4. このエイリアスに追加の認証情報を作成するには、関連リストで [新規] をクリックし、認証情報タイプを選択します。
   エイリアス名は認証情報レコードの [認証情報エイリアス] フィールドに事前に設定されています。
5. フォーム内のフィールドに入力し、レコードを送信します。

オーケストレーション アクティビティの認証情報エイリアス

認証情報エイリアスを使用すると、アドミニストレーターはオーケストレーション アクティビティで使用される認証情報をより詳細に制御できます。

これは、アクティビティがタスクを実行するために特定の信任状を必要とする場合に便利です。[認証情報] タグを使用して、個々の認証情報をオーケストレーション ワークフロー内の任意のアクティビティに割り当てたり、オーケストレーション ワークフロー内の同じアクティビティタイプが発生するたびに異なる認証情報を割り当てたりすることができます。

認証情報エイリアスは、認証情報親和性とやり取りして、どの認証情報をオーケストレーションアクティビティに使用するかを決定します。

認証情報エイリアスの仕組み

ディスカバリー親和性の挿入 (Geneva リリースの「認証情報親和性の挿入」から改名) と呼ばれるビジネス ルール は、レコードが ECC キューに挿入されたときに実行されます。このルールにより、デバイスに認証情報親和性が存在するかどうかが判断され、使用する適切な *credential_id* (認証情報 [discovery_credentials] テーブル内のレコードの *sys_id*) が特定されます。認証情報エイリアス値が定義された (ビジネス ルールの *credential_alias*) 親和性がプラットフォームで見つかると、ビジネス ルールにより、親和性によって参照される認証情報に指定されたエイリアスがあるかどうかが判断されます。エイリアスがある場合、ビジネス ルールにより、認証情報エイリアスの *credential_id* が選択され、その値が MID サーバーに渡されます。認証情報に指定された認証情報エイリアスがない場合は、ターゲット システムに存在するその他の親和性が確認されます。親和性が適切にタグ付けされた認証情報を参照していない場合、MID サーバーで認証情報

[discovery_credentials] テーブルが繰り返し処理され、適切なタグを含む認証情報が選択されます。その後、MID サーバーでこの認証情報に対して新しい親和性が作成されます。

認証情報の作成およびテスト

ディスカバリー、サービスマッピング、クラウド管理、およびオーケストレーションでネットワーク内のハードウェアとソフトウェアへのアクセスに必要な認証情報を作成およびテストします。

始める前に

必要なロール：admin
組織内のセキュリティチームとともにセキュリティポリシーおよびオプションを確認してください。

このタスクについて

このタスクには、認証情報を作成するための一般的な手順が含まれています。特定のフィールドおよび要件の詳細については、「認証情報タイプのドキュメント」を参照してください。

サポートされている認証情報タイプ

適用可能な認証情報	ベーシック認証情報	Chef サーバー認証情報
CIM 認証情報	クラウド認証情報	コンテナイメージリポジトリ認証情報
Infoblox 認証情報	JDBC 認証情報	JMS 認証情報
OAuth 2.0 認証情報	SAP 認証情報	SNMP 認証情報
SSH 認証情報	VMware 認証情報	Windows 資格情報

注: セキュリティを強化するには、認証情報のスコープを特定の MID サーバーまたはスケジュールに制限して、不要な認証情報を回避します。

手順

1. 次のいずれかのモジュールに移動します。
   - ディスカバリー > 認証情報
   - サービスマッピング > 認証情報
   - オーケストレーション > 認証情報
2. [**New**] をクリックします。
3. [認証情報] ページで、認証情報タイプのリンクをクリックしてフォームに入力します。

   詳細については、選択した認証情報タイプのドキュメントを参照してください。

   最初に認証情報レコードを送信してから後でテストするか、認証情報を保存する直前にテストすることができます。

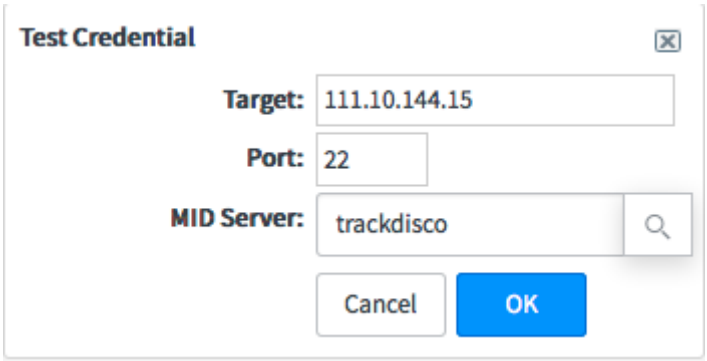
   次の認証情報タイプで認証情報のテストがサポートされています。
   - SSH 秘密鍵
   - Windows
   - SNMP v3
   - VMware
   - JDBC
   - JMS

4. [関連リンク] で、[認証情報をテスト] をクリックします。

   注: 認証情報は、テスト時に常に暗号化されます。

5. [認証情報をテスト] ダイアログボックスのフィールドに入力します。

[認証情報をテスト] ダイアログ ボックス



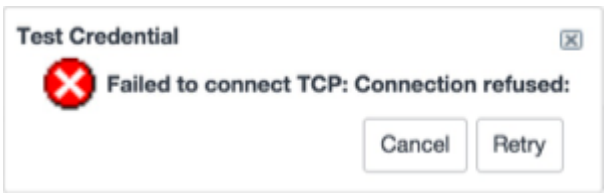
認証情報テスト フィールド

フィールド	説明	認証情報タイプ
ターゲット	これらの認証情報が実行されるターゲット ホスト。この値は、VMware 以外のすべての認証情報タイプの IP アドレスである必要があります。これはホスト URL にすることができます。MID サーバーをターゲットにすることはできません。 注: JMS の場合、これはプロバイダー URL です。この URL の情報により、JMS プロバイダーを検索してアクセスする方法が JNDI に通知されます。ActiveMQ V5.1 に接続するための値の例は、tcp：// ipAddressOrHostName：61616 です。	すべて
ポート	このテストに使用するターゲット上のポート。このフィールドには、選択した認証情報タイプのデフォルト ポートが事前入力されています。	すべて
MID サーバー	このテストに使用する MID サーバー。Windows 資格情報をテストするには、Windows MID サーバーを使用する必要があります。起動中および検証	すべて

フィールド	説明	認証情報タイプ
	済みの MID サーバーのみを利用できます。	
DB タイプ	これらの認証情報をテストするデータベースのタイプ。	JDBC
DB 名	これらの認証情報をテストするデータベースの名前。	JDBC
初期コンテキスト ファクトリー	InitialContext の作成に使用する JNDI クラスの名前。この [初期コンテキスト ファクトリー] を使用すると、JMS 接続などのさまざまな JMS オブジェクトが作成されます。たとえば、ActiveMQ V5.10 (JMS プロバイダー) に接続する場合、このフィールドの値は org.apache.activemq.jndi.ActiveMQInitialContextFactory です。	JMS

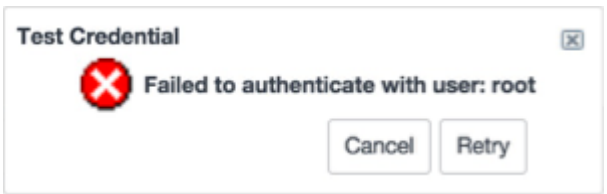
6. **[OK]** をクリックしてテストを開始します。
インジケーターが表示され、入力した認証情報を使用したターゲットへの接続の試行が示されます。インスタンスがターゲットに接続すると、成功メッセージが表示されます。入力したテスト入力に関する問題がインスタンスで発生すると、該当するエラーメッセージが表示されます。よくあるエラーメッセージは次のとおりです。
   - 不適切なターゲットまたはポート番号：

     TCP 接続エラー

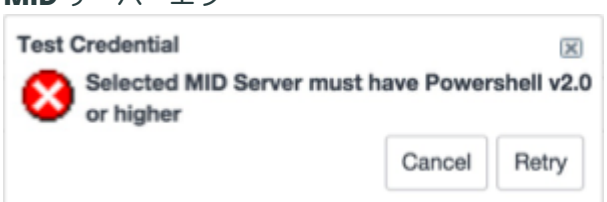
   - ユーザー名またはパスワードが間違っています。

     認証失敗

   - Windows 資格情報の MID サーバーが正しくありません。

     MID サーバーエラー

7. [再試行] をクリックして [認証情報をテスト] ダイアログ ボックスを開き、入力エラーを修正します。
8. 認証情報テストが成功したら、[送信] をクリックしてレコードを保存します。

> **重要:** 認証情報のテストでは、目的のディスカバリーまたはオーケストレーションのワークフロー タスクに必要な権限が認証情報にあるとは保証されません。

Ansible Tower 認証情報

Ansible 構成管理アカウントにアクセスするには、Ansible Tower 認証情報が必要です。これらの認証情報を使用して、クラウド管理アプリケーションで Ansible リソースを管理します。

クラウドプロビジョニングとガバナンス と Ansible 構成管理アカウントを統合するには、Ansible でアドミニストレーターアカウントのユーザー名およびパスワードを設定する必要があります。

Ansible Tower 認証情報のフォームフィールド

フィールド	説明
名前	わかりやすい名前を入力します。
ユーザー名/パスワード	アドミニストレーター権限を持つ Ansible Tower ユーザーの認証情報を入力します。

> **注:** その他のフィールドを設定する必要はありません。

API キー認証情報

API キーは、コール元アプリケーションまたはユーザーを識別するために API に渡される一意のコードです。

API キー認証情報

API キー認証情報フォームのフィールド

フィールド	入力値
名前	この認証情報のわかりやすい一意の名前を入力します。
アクティブ	これらの認証情報の使用を有効または無効にします。
API キー	API キーを入力します。
認証情報エイリアス	フローおよびワークフローの作成者は、個々の認証情報をフローまたはワークフロー内の任意のアクティビティに割り当てたり、フローまたはワークフロー内の同じアクティビティタイプが発生するたびに異なる認証情報を割り当てたりすることができます。
適用先	これらの認証情報をネットワーク内の [すべての **MID** サーバー] または 1 つ以上の [特定の **MID** サーバー] に適用するかどうかを選択します。**[MID** サーバー] フィールドでこれらの認証情報を使用する MID サーバー を指定します。

API キー認証情報フォームのフィールド (続く)

フィールド	入力値
順番	ディスカバリー がデバイスにログオンしようとするときに、この認証情報を試行する順序 (シーケンス)。番号が小さいほど、この認証情報がリストの上位に表示されます。多くの認証情報を使用する場合、またはログイン試行が 3 回失敗してセキュリティによってユーザーがロックアウトされた場合、認証情報の順序を確立します。すべての認証情報の順序番号が同じ (または順序番号がない) 場合、インスタンスは認証情報をランダムな順序で試行します。

適用可能な認証情報

ホストマシンに必要な認証情報に加え、一部のアプリケーションでは認証情報が必要です。これらのアプリケーションにアクセスするために必要な認証情報は、適用可能な認証情報と呼ばれています。

一般的な認証情報には、デバイスまたはアプリケーションにログインするためのユーザー名とパスワードが含まれています。ほとんどのアプリケーションでは、アクセスに必要な認証情報は 1 つだけですが、ホストとアプリケーションによっては、セキュリティを強化するために別の認証情報を持つ場合もあります。たとえば、ABAP SAP Central Services (ASCS) では、ASCS をホストするサーバーの SSH または Windows ホスト認証情報に加えて、適用可能な認証情報が必要です。

注: ServiceNow アプリケーションは、構成アイテム (CI) としてサービスインスタンスを構成するデバイスおよびアプリケーションを参照します。

ホスト認証情報と同様に、適用可能な認証情報を MID サーバー に割り当てます。

CI タイプごとに適用可能な認証情報を作成します。たとえば、ASCS の CI タイプは SAP ASCS アプリケーション [cmdb_ci_appl_sap_ascs] です。この CI タイプに属する CI を検出するための事前設定されたパターンには、MID サーバーにこの CI タイプに適用可能な認証情報を使用することを求めるコマンドが含まれています。この CI タイプに対して複数の認証情報が設定されている場合、MID サーバーは適合する認証情報が見つかるまで、これらの認証情報を定義された順序で試行します。

特定のアプリケーション CI に適用可能な認証情報が必要かどうかを判断するには、ServiceNow のドキュメントの ディスカバリー 要件に関する情報を確認してください。ディスカバリー の前提条件に言及されていなければ、適用可能な認証情報を設定する必要はありません。

適用可能な認証情報フォームのフィールド

フィールド	説明
名前	認証情報の名前です。Oracle DB または London Oracle DB (Oracle データベースの場合) などのわかりやすい名前を使用します。認証情報名にスペースまたは特殊文字を使用することはできません。
アクティブ	認証情報を使用するには、このチェックボックスをオンにします。
ユーザー名	適用可能な認証情報の実際のユーザー名を入力します。

適用可能な認証情報フォームのフィールド (続く)

フィールド	説明
パスワード	適用可能な認証情報の実際のパスワードを入力します。認証情報名にスペースまたは特殊文字を使用することはできません。
CI タイプ	CI が属する CI タイプを選択します。
認証情報エイリアス	特定のディスカバリースケジュールに対して特定の認証情報をアサインするエイリアスを作成します。エイリアスをアサインする際、アプリケーションが使用する適用可能な認証情報を持つ CI タイプのテーブル名を識別する必要があります。アプリケーションは、それ自身とは異なる CI タイプの適用可能な認証情報を使用する場合があります。特定のアプリケーションについては、次の該当するテーブルのリストを参照してください。 • ABAP SAP Central Services (ASCS): cmdb_ci_appl_sap_ascs • IBM Security Access Manager アプライアンス：cmdb_ci_app_server_webseal • SAP Central インスタンス: cmdb_ci_appl_sap_ascs • SAP Central Services (SCS): cmdb_ci_appl_sap_ascs • SAP Evaluated Receipt Settlement (ERS): cmdb_ci_appl_sap_ascs • SAP Java クラスター: cmdb_ci_appl_sap_ascs • SAP NetWeaver Dialog インスタンス: cmdb_ci_appl_sap_ascs • Microsoft Exchange Mailbox (Microsoft Exchange の場合): cmdb_ci_exchange_mailbox • Microsoft SQL Database: cmdb_ci_db_mssql_instance • MySQL Server: cmdb_ci_db_mysql_instance • Oracle Advanced Queue Queue: cmdb_ci_db_ora_instance • Oracle Database: cmdb_ci_db_ora_instance • Oracle E-Business Suite: cmdb_ci_db_ora_instance • Oracle WebLogic モジュール: cmdb_ci_app_server_weblogic • Tibco Enterprise Message Service (EMS): cmdb_ci_appl_tibco_message
適用先	これらの認証情報をネットワーク内の [すべての **MID** サーバー] または 1 つ以上の [特定の **MID** サーバー] に適用するかどうかを選択します。**[MID** サーバー] フィールドでこれらの認証情報を使用する MID サーバー を指定します。
順序	ディスカバリー がデバイスにログオンしようとするときに、この認証情報を試行する順序 (シーケンス)。番号が小さいほど、この認証情報がリストの上位に表示されます。多くの認証情報を使用する場合、またはログイン試行が 3 回失敗してセキュリティによってユーザーがロックアウトされた場合、認証情報の順序を確立します。すべての認証情報の順序番号が同じ (または順序番号がない) 場合、インスタンスは認証情報をランダムな順序で試行します。

ベーシック認証情報

ベーシック認証情報タイプにより、ベーシック認証情報を保存するためのアクセス権が管理されます。

次のフィールドは、ベーシック認証の認証情報フォームで使用できます。

基本認証情報フォーム

フィールド	入力値
名前	この認証情報のわかりやすい一意の名前を入力します。たとえば、**Basic Authentication** と入力します。
ユーザー名	ユーザー名を入力します。
パスワード	パスワードを入力します。
認証情報 ID	CyberArk の外部認証情報ストレージ システムでこれらの認証情報に設定されている一意のキーを入力します。認証情報 ID は、複数のセーフが使用されている場合に安全な上書きとして使用できます。デフォルトで、[認証情報 **ID]** フィールドの構文は *<safe name>:<Credential ID>* です。セーフ名を省略する場合、config.xml ファイルで定義されているセーフ名が必要です。区切り文字をデフォルトのコロンから別の文字に変更するには、値をオプションの *ext.cred.safe_name* パラメーターで上書きします。[認証情報 **ID]** フィールドには 40 文字の制限があります。 このフィールドは、**[External storage]** (外部ストレージ) チェック ボックスがオンになっている場合にのみ表示されます。
外部の認証情報ストア	外部認証情報ストレージシステムを使用するには、このチェックボックスをオンにします。このオプションを選択すると、[ユーザー名] フィールドと [パスワード] フィールドが [認証情報 **ID]** フィールドに置き換えられます。現在サポートされている外部ストレージ システムは CyberArk だけです。
順番	ディスカバリー がデバイスにログオンしようとするときに、この認証情報を試行する順序 (シーケンス)。番号が小さいほど、この認証情報がリストの上位に表示されます。多くの認証情報を使用する場合、またはログイン試行が 3 回失敗してセキュリティによってユーザーがロックアウトされた場合、認証情報の順序を確立します。すべての認証情報の順序番号が同じ (または順序番号がない) 場合、インスタンスは認証情報をランダムな順序で試行します。

Chef サーバー認証情報

Chef サーバー認証情報は、インスタンスとの Chef 統合にアクセスします。

次のフィールドは、Chef サーバー タイプの認証情報の認証情報フォームで利用可能です。この情報は、[Chef server installation] (Chef サーバー インストール) の実行時に設定した設定から取得されます。

フィールド	説明
名前	この認証情報のわかりやすい一意の名前を入力します。

フィールド	説明
アクティブ	これらの認証情報の使用を有効または無効にします。
アドミン名	Chef サーバーのインストール時に作成したアドミン名を入力します。
管理キー	アドミンの作成時に Chef サーバーによって生成された RSA 秘密鍵を入力します。
検証ツール名	検証ツールを入力します。
検証ツールキー	組織の作成時に Chef サーバーによって生成された RSA 秘密鍵を入力します。
証明書名	認定名を入力します。
証明書キー	認定キーを入力します。

CIM 認証情報

CIM 認証情報タイプは、VMware ESX サーバーについての情報を得るための CIM サーバー (別名 CIMOM - 共通情報モデル オブジェクト マネージャー) へのアクセス権を管理します。 この認証情報タイプはディスカバリーで使用できます。

次のフィールドは、CIM の認証情報フォームで使用できます。

フィールド	説明
名前	この認証情報のわかりやすい一意の名前を入力します。
アクティブ	これらの認証情報の使用を有効または無効にします。
ユーザー名	[認証情報] テーブルに、作成するユーザー名を入力します。ユーザー名の先頭または末尾にはスペースを使用しないでください。プラットフォームでユーザー名の先頭または末尾のスペースが検出されると、警告が表示されます。CIM ディスカバリーの場合、ユーザーには admin ロールが必要です。
パスワード	パスワードを入力します。
認証情報 ID	外部認証情報システム用に MID サーバーにアップロードされた JAR ファイルに外部認証情報用に設定された一意のキーを入力します。[認証情報 **ID**] フィールドには 40 文字の制限があります。 このフィールドは、[外部の認証情報ストア] チェックボックスがオンになっている場合にのみ表示されます。
認証情報エイリアス	ワークフロー作成者は、個々の認証情報をオーケストレーションワークフロー内の任意のアクティビティに割り当てたり、オーケストレーションワークフロー内で同じアクティビティタイプが発生するたびに異なる認証情報を割り当てたりすることができます。 サービスマッピングおよびディスカバリーパターンを使用してこの CI タイプに属していない CI を検出するために認証情報を使用するには、CI が属する CI タイプのテーブル名 (cmdb_ci_apache_web_server など) を入力します。

フィールド	説明
外部の認証情報ストア	外部認証情報ストレージシステムを使用するには、このチェックボックスをオンにします。このオプションを選択すると、[ユーザー名] フィールドと [パスワード] フィールドが [認証情報 **ID**] フィールドに置き換えられます。外部の認証情報ストレージは、外部認証情報ストレージ プラグインをアクティブ化した場合にのみ使用できます。 注: 現在サポートされている外部ストレージ システムは CyberArk だけです。
適用先	これらの認証情報をネットワーク内の [すべての **MID** サーバー] または 1 つ以上の [特定の **MID** サーバー] に適用するかどうかを選択します。**[MID** サーバー] フィールドでこれらの認証情報を使用する MID サーバー を指定します。
MID サーバーを使用	使用可能な MID サーバーのリストから 1 つ以上の MID サーバーを選択します。このレコードで設定された認証情報は、このリストの MID サーバーで使用できます。このフィールドは、[適用先] フィールドで [特定の **MID** サーバー] を選択した場合のみ使用できます。 注: [特定の **MID** サーバー] を選択しても、MID サーバーの選択には影響しません。これは、認証情報を表示する必要がある MID サーバーを決定するためにのみ使用されます。[特定の **MID** サーバー] はオーケストレーションアクティビティではサポートされていません。
順序	ディスカバリー がデバイスにログオンしようとするときに、この認証情報を試行する順序 (シーケンス)。番号が小さいほど、この認証情報がリストの上位に表示されます。多くの認証情報を使用する場合、またはログイン試行が 3 回失敗してセキュリティによってユーザーがロックアウトされた場合、認証情報の順序を確立します。すべての認証情報の順序番号が同じ (または順序番号がない) 場合、インスタンスは認証情報をランダムな順序で試行します。
Windows MID サーバー サービスアカウント	有効な場合、定義された認証情報は MID サーバー サービスアカウントを表します。

NetApp ストレージデバイスでの **CIM** 認証情報の設定

ディスカバリー で NetApp ストレージデバイスを検出するには、追加の設定が必要です。

始める前に
必要なロール：admin

手順

1. ストレージ デバイス ホスト上に SMI-S エージェント をインストールします。

   手順および要件については、「Data ONTAP SMI-S Agent 5.2 のインストールおよび設定ガイド 」を参照してください。

   注:

   ServiceNow では、このサイトでこのドキュメントを管理していません。このドキュメントは、予告なしに変更される可能性があります。

2. SMI-S エージェントのユーザーアカウントとパスワードを作成します。
3. SMI-S エージェント認証情報の認証情報レコードを作成します。
   認証情報タイプを **[CIM]** に設定します。

クラウド認証情報

クラウド認証情報タイプにより、Amazon Web Services や Microsoft Azure クラウドなどのクラウドベースのアプリケーションへのアクセス権が管理されます。

AWS Identity and Access Management (IAM) ロール

AWS クラウドの Amazon EC2 に MID サーバーがインストールされていて、その MID サーバーがクラウド内のリソースを検出するように設定されている場合は、インスタンスで管理されている認証情報ではなく、AWS Identity and Access Management (IAM) ロールによって提供されるセキュリティ認証情報を使用できます。これらの AWS 認証情報は、ロールに基づいて、インスタンスプロファイルを通じてクラウドでの権限を付与します。これらの認証情報は一時的なものであり、構成可能な間隔で自動的にローテーションされます。MID サーバーで IAM ロールが定義されている場合。詳細については、「AWS IAM ロール用の MID サーバーの構成」を参照してください。

ディスカバリーはインスタンスに格納されているすべての認証情報を無視し、インスタンスプロファイルのロールによって付与された認証情報を優先します。AWS インスタンスプロファイルの詳細については、「Amazon EC2 の IAM ロール (IAM Roles for Amazon EC2)」を参照してください。

AWS 認証情報

AWS 認証情報フォームのフィールド

フィールド	入力値
名前	AWS 認証情報の分かりやすい一意の名前。
アクティブ	認証情報を使用するオプション。
アクセスキー ID	AWS 管理コンソールで生成された [アクセスキー **ID]** (例：APIAIOSFODNN7EXAMPLE)。
秘密アクセスキー	AWS 管理コンソールで生成された [秘密アクセスキー] (例：wPalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY)。

Azure サービスプリンシパル認証情報フォームフィールド

フィールド	値
名前	インスタンスに登録するサービスプリンシパルの名前を入力します。
テナント ID および	Azure ポータルの Azure ディレクトリ **ID** の値を クラウド管理 [テナント **ID]** フィールドに貼り付けます。
クライアント ID	Azure に登録したアプリケーションの Azure アプリケーション **ID** の値を クラウド管理 [クライアント **ID]** フィールドに貼り付けます。
認証方法	[クライアントシークレット] を選択します。

フィールド	値
	注: [クライアントアサーション] はサポートされていません。
秘密キー	Azure サービスプリンシパルの作成中に生成された秘密キーを貼り付けます。 このフィールドは、[認証手法] が [クライアントシークレット] のときに表示されます。

Azure エンタープライズ契約認証情報

Azure エンタープライズ契約認証情報は、クラウド管理アプリケーションで提供される請求機能に必要です。

Azure エンタープライズ契約認証情報フォームのフィールド

フィールド	説明
名前	内容を端的に表す名前を入力します。
登録数	Azure からの登録番号を入力します。
アクセス キー	Azure が提供するアクセス キーを貼り付けます。

クラウド管理認証情報

この認証情報タイプはオーケストレーションで使用できます。

クラウド管理認証情報フォームのフィールド

フィールド	入力値
名前	この認証情報のわかりやすい一意の名前を入力します。たとえば、**Cloud Atlanta** と入力します。
有効	これらの認証情報の使用を有効または無効にします。
タイプ	**[AWS]** を指定します。
ユーザー名	認証情報テーブルに作成する CIM ユーザー名を入力します。ユーザー名の先頭または末尾にはスペースを使用しないでください。プラットフォームでユーザー名の先頭または末尾のスペースが検出されると、警告が表示されます。
パスワード	CIM パスワードを入力します。
SSH パスフレーズ	キーの生成用に覚えやすいフレーズを入力します。たとえば、**Friday is a good day** と入力します。
SSH 秘密鍵	SSH 秘密鍵を入力します。

クラウド管理認証情報フォームのフィールド (続く)

フィールド	入力値
認証プロトコル	[認証キー] の生成に使用した **[MD5]** または **[SHA]** の認証プロトコルを選択します。
認証キー	SSH で生成された認証キーを入力します。
プライバシープロトコル	[プライバシー キー] の暗号化を説明する次のいずれかのプライバシー プロトコルを入力します。 • **3DES** (Triple Data Encryption Standard (DES) の場合) • **AES128** (128 ビット暗号化を使用する Advanced Encryption Standard (AES) の場合) • **AES192** (192 ビット暗号化を使用する AES の場合) • **AES256** (256 ビット暗号化を使用する AES の場合) • **DES** (レガシー DES 暗号化の場合)
追加のプライバシー キーを入力します。	
認証情報エイリアス	ワークフロー作成者は、個々の認証情報をオーケストレーションワークフロー内の任意のアクティビティに割り当てたり、オーケストレーションワークフロー内で同じアクティビティタイプが発生するたびに異なる認証情報を割り当てたりすることができます。
外部の認証情報ストア	外部認証情報ストレージシステムを使用するには、このチェックボックスをオンにします。このオプションを選択すると、[ユーザー名] フィールドと [パスワード] フィールドが [認証情報 **ID]** フィールドに置き換えられます。現在サポートされている外部ストレージ システムは CyberArk だけです。
適用先	これらの認証情報をネットワーク内の [すべての **MID** サーバー] または 1 つ以上の [特定の **MID** サーバー] に適用するかどうかを選択します。**[MID** サーバー] フィールドでこれらの認証情報を使用する MID サーバー を指定します。
分類	CI ディスカバリーのアプリケーション分類を入力します。
順番	ディスカバリー がデバイスにログオンしようとするときに、この認証情報を試行する順序 (シーケンス)。番号が小さいほど、この認証情報がリストの上位に表示されます。多くの認証情報を使用する場合、またはログイン試行が 3 回失敗してセキュリティによってユーザーがロックアウトされた場合、認証情報の順序を確立します。すべての認証情報の順序番号が同じ (または順序番号がない) 場合、インスタンスは認証情報をランダムな順序で試行します。

クラウド管理 (CMP) ノード認証情報

クラウド管理 (CMP) ノード認証情報は、クラウド管理によってプロビジョニングされた仮想サーバーの認証情報を関連付けます。クラウド管理アプリケーションにより、これらの認証情報が自動的に作成されます。

注: これらの認証情報を使用しない場合の非アクティブ化、順序の優先順位の変更、またはこれらの認証情報へのアクセスが許可されている MID サーバーの選択が必要な場合があります。それ以外の場合は、このタイプの認証情報を手動で作成または変更する必要はありません。

CMP ノード認証情報フォームのフィールド

フィールド	説明
名前	仮想マシンが配置されているデータセンターに基づいて自動的に生成される名前。
有効	認証情報がアクティブな場合。
適用先	この認証情報を特定の MID サーバーまたはすべての MID サーバーで利用できるかどうかを選択します。
順番	ディスカバリー がデバイスにログオンしようとするときに、この認証情報を試行する順序 (シーケンス)。番号が小さいほど、この認証情報がリストの上位に表示されます。多くの認証情報を使用する場合、またはログイン試行が 3 回失敗してセキュリティによってユーザーがロックアウトされた場合、認証情報の順序を確立します。すべての認証情報の順序番号が同じ (または順序番号がない) 場合、インスタンスは認証情報をランダムな順序で試行します。
ユーザー名とパスワード	仮想サーバーのユーザー名とパスワード。
SSH パスフレーズと SSH 秘密鍵	仮想サーバーでキーが必要な場合にそのキーを保護する秘密鍵とパスフレーズ。
認証プロトコルと認証キー	仮想サーバーでキーが必要な場合にそのキーを保護する秘密鍵とパスフレーズ。
プライバシープロトコルとプライバシーキー	仮想サーバーで使用される暗号化プロトコル。プライバシー キーを入力します。
認証情報エイリアス	ワークフロー作成者は、個々の認証情報をオーケストレーションワークフロー内の任意のアクティビティに割り当てたり、オーケストレーションワークフロー内で同じアクティビティタイプが発生するたびに異なる認証情報を割り当てたりすることができます。

クラウド管理 (CMP) SSH キー ペア認証情報

ユーザーがスタック リソースをプロビジョニングしたときに、クラウド管理 (CMP) SSH キー ペアにより、クラウド管理アプリケーションで自動的に生成されるキーが保存されます。

注: これらの認証情報を使用しない場合は、これらの認証情報を非アクティブ化する必要があります。それ以外の場合は、このタイプの認証情報を手動で作成または変更する必要はありません。

CMP SSH キー ペア認証情報フォームのフィールド

フィールド	説明
名前	自動的に生成される名前。
有効	認証情報がアクティブな場合。
SSH 公開鍵	公開鍵。
SSH 秘密鍵	SSH ログインのパスワードの代わりに使用できる安全な秘密鍵。

コンテナイメージリポジトリ認証情報

コンテナイメージリポジトリ認証情報は、コンテナイメージスキャン用のプライベートリポジトリへのアクセスを管理します。この認証情報タイプは、ディスカバリー で使用できます。

次のフィールドは、認証情報イメージリポジトリタイプの認証情報の認証情報フォームで使用できます。

コンテナイメージリポジトリ認証情報フォーム

フィールド	説明
名前	ユーザーが割り当てた認証情報名。
ユーザー名	リポジトリの読み取り権限を持つユーザー名。
パスワード	リポジトリの読み込み権限を持つパスワード。
リポジトリ	リポジトリの完全修飾ドメイン名。例：docker.io/snow_images。

Infoblox 認証情報

Infoblox 認証情報は、クラウド管理アプリケーションで IP プール (IPAM) を設定するために必要です。

次のフィールドは、Infoblox タイプの認証情報の認証情報フォームで使用できます。

フィールド	説明
名前	この認証情報のわかりやすい一意の名前を入力します。
アクティブ	これらの認証情報の使用を有効または無効にします。
適用先	この認証情報を特定の MID サーバーまたはすべての MID サーバーで利用できるかどうかを選択します。

フィールド	説明
順番	ディスカバリー がデバイスにログオンしようとするときに、この認証情報を試行する順序 (シーケンス)。番号が小さいほど、この認証情報がリストの上位に表示されます。多くの認証情報を使用する場合、またはログイン試行が 3 回失敗してセキュリティによってユーザーがロックアウトされた場合、認証情報の順序を確立します。すべての認証情報の順序番号が同じ (または順序番号がない) 場合、インスタンスは認証情報をランダムな順序で試行します。 ディスカバリー がデバイスにログオンしようとするときに、この認証情報を試行する順序 (シーケンス)。番号が小さいほど、この認証情報がリストの上位に表示されます。多くの認証情報を使用する場合、またはログイン試行が 3 回失敗してセキュリティによってユーザーがロックアウトされた場合、認証情報の順序を確立します。すべての認証情報の順序番号が同じ (または順序番号がない) 場合、インスタンスは認証情報をランダムな順序で試行します。
WAPI バージョン	使用している WAPI のバージョン を入力します。
ユーザー名とパスワード	InfoBlox のユーザー名とパスワードを入力します。

JDBC 認証情報

JDBC 認証情報タイプにより、Java Database Connectivity (JDBC) 接続へのアクセスが管理されます。この認証情報タイプはディスカバリーとオーケストレーションで使用できます。

次のフィールドは、JDBC タイプの認証情報の認証情報フォームで使用できます。

フィールド	説明
名前	この認証情報のわかりやすい一意の名前を入力します。
アクティブ	これらの認証情報の使用を有効または無効にします。
ユーザー名	[認証情報] テーブルに、作成するユーザー名を入力します。ユーザー名の先頭または末尾にはスペースを使用しないでください。プラットフォームでユーザー名の先頭または末尾のスペースが検出されると、警告が表示されます。CIM ディスカバリーの場合、ユーザーには admin ロールが必要です。
パスワード	パスワードを入力します。
認証情報 ID	外部認証情報システム用に MID サーバーにアップロードされた JAR ファイルに外部認証情報用に設定された一意のキーを入力します。[認証情報 **ID**] フィールドには 40 文字の制限があります。

フィールド	説明
	このフィールドは、[外部の認証情報ストア] チェックボックスがオンになっている場合にのみ表示されます。
認証情報エイリアス	ワークフロー作成者は、個々の認証情報をオーケストレーションワークフロー内の任意のアクティビティに割り当てたり、オーケストレーションワークフロー内で同じアクティビティタイプが発生するたびに異なる認証情報を割り当てたりすることができます。 サービスマッピングおよびディスカバリーパターンを使用してこの CI タイプに属していない CI を検出するために認証情報を使用するには、CI が属する CI タイプのテーブル名 (cmdb_ci_apache_web_server など) を入力します。
外部の認証情報ストア	外部認証情報ストレージシステムを使用するには、このチェックボックスをオンにします。このオプションを選択すると、[ユーザー名] フィールドと [パスワード] フィールドが [認証情報 **ID]** フィールドに置き換えられます。外部の認証情報ストレージは、外部認証情報ストレージ プラグインをアクティブ化した場合にのみ使用できます。 **注:** 現在サポートされている外部ストレージ システムは CyberArk だけです。
適用先	これらの認証情報をネットワーク内の [すべての **MID** サーバー] または 1 つ以上の [特定の **MID** サーバー] に適用するかどうかを選択します。**[MID** サーバー] フィールドでこれらの認証情報を使用する MID サーバー を指定します。
MID サーバーを使用	使用可能な MID サーバーのリストから 1 つ以上の MID サーバーを選択します。このレコードで設定された認証情報は、このリストの MID サーバーで使用できます。このフィールドは、[適用先] フィールドで [特定の **MID** サーバー] を選択した場合のみ使用できます。 **注:** [特定の **MID** サーバー] を選択しても、MID サーバーの選択には影響しません。これは、認証情報を表示する必要がある MID サーバーを決定するためにのみ使用されます。[特定の **MID** サーバー] はオーケストレーションアクティビティではサポートされていません。
順序	ディスカバリー がデバイスにログオンしようとするときに、この認証情報を試行する順序 (シーケンス)。番号が小さいほど、この認証情報がリストの上位に表示されます。多くの認証情報を使用する場合、またはログイン試行が 3 回失敗してセキュリティによってユーザーがロックアウトされた場合、認証情報の順序を確立します。すべての認証情報の順序番号が同じ (または順序番号がない) 場合、インスタンスは認証情報をランダムな順序で試行します。
Windows MID サーバー サービスアカウント	有効な場合、定義された認証情報は MID サーバー サービスアカウントを表します。

JMS 認証情報

JMS 認証情報タイプにより、Java Message Service (JMS) へのアクセスが管理されます。この認証情報タイプはディスカバリーとオーケストレーションで使用できます。

次のフィールドは、JMS 認証情報フォームで使用できます。

フィールド	説明
名前	この認証情報のわかりやすい一意の名前を入力します。
アクティブ	これらの認証情報の使用を有効または無効にします。
ユーザー名	[認証情報] テーブルに、作成するユーザー名を入力します。ユーザー名の先頭または末尾にはスペースを使用しないでください。プラットフォームでユーザー名の先頭または末尾のスペースが検出されると、警告が表示されます。CIM ディスカバリーの場合、ユーザーには admin ロールが必要です。
パスワード	パスワードを入力します。
認証情報 ID	外部認証情報システム用に MID サーバーにアップロードされた JAR ファイルに外部認証情報用に設定された一意のキーを入力します。[認証情報 **ID]** フィールドには 40 文字の制限があります。 このフィールドは、[外部の認証情報ストア] チェックボックスがオンになっている場合にのみ表示されます。
認証情報エイリアス	ワークフロー作成者は、個々の認証情報をオーケストレーションワークフロー内の任意のアクティビティに割り当てたり、オーケストレーションワークフロー内で同じアクティビティタイプが発生するたびに異なる認証情報を割り当てたりすることができます。 サービスマッピングおよびディスカバリーパターンを使用してこの CI タイプに属していない CI を検出するために認証情報を使用するには、CI が属する CI タイプのテーブル名 (cmdb_ci_apache_web_server など) を入力します。
外部の認証情報ストア	外部認証情報ストレージシステムを使用するには、このチェックボックスをオンにします。このオプションを選択すると、[ユーザー名] フィールドと [パスワード] フィールドが [認証情報 **ID]** フィールドに置き換えられます。外部の認証情報ストレージは、外部認証情報ストレージ プラグインをアクティブ化した場合にのみ使用できます。 **注:** 現在サポートされている外部ストレージ システムは CyberArk だけです。
適用先	これらの認証情報をネットワーク内の [すべての **MID** サーバー] または 1 つ以上の [特定の **MID** サーバー] に適用するかどうかを選択します。**[MID** サーバー] フィールドでこれらの認証情報を使用する MID サーバー を指定します。
MID サーバーを使用	使用可能な MID サーバーのリストから 1 つ以上の MID サーバーを選択します。このレコードで設定された認証情報は、このリストの MID サーバーで使用できます。このフィールドは、[適用先] フィールドで [特定の **MID** サーバー] を選択した場合のみ使用できます。 **注:** [特定の **MID** サーバー] を選択しても、MID サーバーの選択には影響しません。これは、認証情報を表示する必要がある MID サーバーを決定するためにのみ使用されます。[特定の **MID** サーバー] はオーケストレーションアクティビティではサポートされていません。
順序	ディスカバリー がデバイスにログオンしようとするときに、この認証情報を試行する順序 (シーケンス)。番号が小さいほど、この認証情報がリストの上位に表示されます。

フィールド	説明
	多くの認証情報を使用する場合、またはログイン試行が 3 回失敗してセキュリティによってユーザーがロックアウトされた場合、認証情報の順序を確立します。すべての認証情報の順序番号が同じ (または順序番号がない) 場合、インスタンスは認証情報をランダムな順序で試行します。
Windows MID サーバー サービスアカウント	有効な場合、定義された認証情報は MID サーバー サービスアカウントを表します。

OAuth 2.0 認証情報

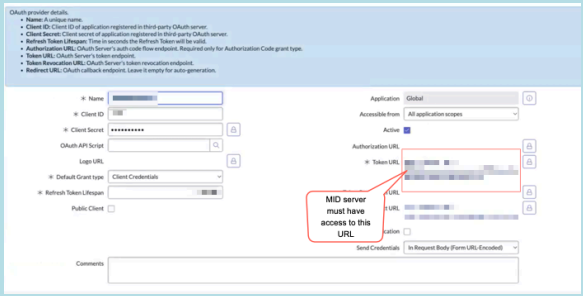
OAuth 2.0 の認証情報により、ServiceNow は HTTP サービス上のユーザーアカウントにアクセスできるようになります。

次のフィールドは、OAuth 2.0 の認証情報フォームで使用できます。

OAuth 2.0 認証情報フォーム

フィールド	入力値
名前	この認証情報のわかりやすい一意の名前を入力します。たとえば、 **OAuth2 credential** と入力します。
有効	この認証情報がアクティブかどうかを指定します。
OAuth エンティティプロファイル	OAuth プロファイルは、権限付与タイプと少なくとも 1 つのスコープを組み合わせたものです。
MID サーバー経由で認証サーバーに接続	MID サーバーを介して、ファイアウォールの背後に存在するオンプレミスの OAuth サーバーに ServiceNow インスタンスを接続します。また、MID サーバーを介して ServiceNow インスタンスをクラウドベースの OAuth サーバーに接続することもできます。このオプションを有効にすると、OAuth トークンの要求が MID サーバーを介して送信されます。 **重要:** • このオプションは、OAuth エンティティプロファイルの [権限許可タイプ] フィールドの値が[クライアント認証情報]、[認証コード]、または [リソース所有者のパスワード認証情報] に設定されている場合に表示されます。サードパーティ OAuth プロバイダーの OAuth エンティティプロファイルを設定する方法については、「サードパーティ OAuth プロバイダーへの接続」を参照してください。 • **[MID** サーバー経由で認証サーバーに接続] チェックボックスをオンにした場合は、[適用先] リストから、必要な 1 つの MID サーバーまたは複数の MID サーバーを特定する必要があります。

OAuth 2.0 認証情報フォーム (続く)

フィールド	入力値
適用先	認証情報をすべての MID サーバーに適用するか、特定の MID サーバーに適用するかを指定します。特定の MID サーバーに適用する場合は、必要に応じてその MID サーバーを追加します。 **重要:** **[MID** サーバー経由で認証サーバーに接続] チェックボックスをオンにした場合は、これらの考慮事項に注意してください。 • [適用先] で選択したすべての MID サーバーが、認証サーバーと通信できることを確認します。これは、OAuth プロバイダーレコードに記載されている (OAuth 2.0 認証情報レコードで指定された OAuth エンティティプロファイルにリンクされている) トークン **URL** に対してトークン要求を実行するうえで必要です。  • 次の構成を持つ MID サーバー ([適用先] で選択済みの MID サーバー) が少なくとも 1 つあることを確認します。 ◦ [ステータス] フィールドの値は [稼働中 **(Up)]** です。 ◦ [検証済み] フィールドの値は [はい] です。 ◦ MID サーバーの機能は **[REST]** または [すべて] に設定されています。MID サーバーの構成方法については、「https://docs.servicenow.com/csh?topicname=t_ConfigureCapabilities&version=zurich&pubname=zurich-integrate-applications」を参照してください。  これらのステータスの詳細については、「MID サーバーダッシュボード」を参照してください。
順番	ディスカバリー がデバイスにログオンしようとするときに、この認証情報を試行する順序 (シーケンス)。番号が小さいほど、この認証情報がリストの上位に表示されます。多くの認証情報を使用する場合、またはログイン試行が 3 回失敗してセキュリティによってユーザーがロックアウトされた場合、認証情報の順序を確立します。すべての認証情報の順序番号が同じ (または順序番号がない) 場合、インスタンスは認証情報をランダムな順序で試行します。
認証情報エイリアス	OAuth 2.0 認証情報に関連付ける認証情報エイリアスを指定します。

OAuth 2.0 認証情報フォーム (続く)

フィールド	入力値
統合タイプ	認証情報の統合タイプを示します。システムまたはユーザー固有の OAuth トークンを生成する OAuth 要求を使用して、サードパーティの API を呼び出します。統合タイプは次のとおりです。 • システム：要求者プロファイルに基づいてトークン情報をプルします。システム統合タイプは、次の認証メカニズムをサポートしています。 **1.** セキュリティアサーションマークアップ言語 (SAML) **2.** JSON Web トークン (JWT) • 個人：ユーザー固有のトークン情報をプルします。MID サーバーユーザーには、oauth_admin ロールが必要です。パーソナルおよびシステムの統合タイプでは、次の権限許可タイプがサポートされています。 **1.** 認証コード **2.** リソース所有者のパスワード認証情報 [個人] を選択すると、[OAuth 要求者プロファイル] ページに [個人] というフラグが追加で表示されます。 **注:** • ユーザーに関連する情報には、[データ連携タイプ] が [個人] のユーザー固有の OAuth トークンでのみアクセスできます。 • セッションユーザー関連のトークンを使用するには、[フロー] プロパティの [実行方法] フィールドで [セッションを開始するユーザー] を選択する必要があります。

SAP 認証情報

SAP 認証情報タイプにより、SAP JCo システムへのアクセスが管理されます。この認証情報タイプはディスカバリーとオーケストレーションで使用できます。

次のフィールドは、SAP タイプの認証情報の認証情報フォームで使用できます。

フィールド	説明
名前	この認証情報のわかりやすい一意の名前を入力します。
アクティブ	これらの認証情報の使用を有効または無効にします。
ユーザー名	[認証情報] テーブルに、作成するユーザー名を入力します。ユーザー名の先頭または末尾にはスペースを使用しないでください。プラットフォームでユーザー名の先頭または末尾のスペースが検出されると、警告が表示されます。CIM ディスカバリーの場合、ユーザーには admin ロールが必要です。
パスワード	パスワードを入力します。

フィールド	説明
認証情報 ID	外部認証情報システム用に MID サーバーにアップロードされた JAR ファイルに外部認証情報用に設定された一意のキーを入力します。[認証情報 **ID]** フィールドには 40 文字の制限があります。 このフィールドは、[外部の認証情報ストア] チェックボックスがオンになっている場合にのみ表示されます。
認証情報エイリアス	ワークフロー作成者は、個々の認証情報をオーケストレーションワークフロー内の任意のアクティビティに割り当てたり、オーケストレーションワークフロー内で同じアクティビティタイプが発生するたびに異なる認証情報を割り当てたりすることができます。 サービスマッピングおよびディスカバリーパターンを使用してこの CI タイプに属していない CI を検出するために認証情報を使用するには、CI が属する CI タイプのテーブル名 (cmdb_ci_apache_web_server など) を入力します。
外部の認証情報ストア	外部認証情報ストレージシステムを使用するには、このチェックボックスをオンにします。このオプションを選択すると、[ユーザー名] フィールドと [パスワード] フィールドが [認証情報 **ID]** フィールドに置き換えられます。外部の認証情報ストレージは、外部認証情報ストレージ プラグインをアクティブ化した場合にのみ使用できます。 **注:** 現在サポートされている外部ストレージ システムは CyberArk だけです。
適用先	これらの認証情報をネットワーク内の [すべての **MID** サーバー] または 1 つ以上の [特定の **MID** サーバー] に適用するかどうかを選択します。**[MID** サーバー] フィールドでこれらの認証情報を使用する MID サーバー を指定します。
MID サーバーを使用	使用可能な MID サーバーのリストから 1 つ以上の MID サーバーを選択します。このレコードで設定された認証情報は、このリストの MID サーバーで使用できます。このフィールドは、[適用先] フィールドで [特定の **MID** サーバー] を選択した場合のみ使用できます。 **注:** [特定の **MID** サーバー] を選択しても、MID サーバーの選択には影響しません。これは、認証情報を表示する必要がある MID サーバーを決定するためにのみ使用されます。[特定の **MID** サーバー] はオーケストレーションアクティビティではサポートされていません。
順序	ディスカバリー がデバイスにログオンしようとするときに、この認証情報を試行する順序 (シーケンス)。番号が小さいほど、この認証情報がリストの上位に表示されます。多くの認証情報を使用する場合、またはログイン試行が 3 回失敗してセキュリティによってユーザーがロックアウトされた場合、認証情報の順序を確立します。すべての認証情報の順序番号が同じ (または順序番号がない) 場合、インスタンスは認証情報をランダムな順序で試行します。
Windows MID サーバー サービスアカウント	有効な場合、定義された認証情報は MID サーバー サービスアカウントを表します。

SNMP 認証情報

ディスカバリー では、SNMP プロトコルを使用して、多くの種類のデバイス (スイッチ、ルーター、プリンターなど) を調べます。SNMP の認証情報にはユーザー名が含まれておらず、コミュニティ文字列と呼ばれるパスワードのみが含まれています。

多くの SNMP デバイス向けのデフォルトの読み取り専用コミュニティ文字列はパブリックであり、ディスカバリーは自動的にその文字列を試行します。*public* コミュニティ文字列ではない場合は、適切な SNMP 認証情報を入力してください。

SNMP を検出すると、設定されているすべてのコミュニティ文字列が使用されます。この動作は、SNMPv3 の検出には適用されません。

デフォルトのオーケストレーション アクティビティの SNMP クエリーは、デバイスのオブジェクト識別子 (OID) を返し、SNMP 認証情報を求めます。

SNMP コミュニティ認証情報

SNMP コミュニティ認証情報タイプにより、SNMP プロトコルを使用して多くの種類のデバイス (スイッチ、ルーター、プリンターなど) を検出するためのアクセスが管理されます。この認証情報タイプは ディスカバリー、 サービスマッピング、および オーケストレーション で使用できます。

SNMP の認証情報にはユーザー名は含まれておらず、パスワード (コミュニティ文字列) のみが含まれます。多くの SNMP デバイス向けのデフォルトの読み取り専用コミュニティ文字列はパブリックであり、自動的にその文字列が試行されます。パブリックのコミュニティ文字列ではない場合は、適切な SNMP 認証情報を入力してください。

次のフィールドは、SNMP コミュニティの認証情報フォームで使用できます。

フィールド	説明
名前	この認証情報のわかりやすい一意の名前を入力します。
アクティブ	これらの認証情報の使用を有効または無効にします。
ユーザー名	[認証情報] テーブルに、作成するユーザー名を入力します。ユーザー名の先頭または末尾にはスペースを使用しないでください。プラットフォームでユーザー名の先頭または末尾のスペースが検出されると、警告が表示されます。CIM ディスカバリーの場合、ユーザーには admin ロールが必要です。
パスワード	パスワードを入力します。
認証情報 ID	外部認証情報システム用に MID サーバーにアップロードされた JAR ファイルに外部認証情報用に設定された一意のキーを入力します。[認証情報 **ID]** フィールドには 40 文字の制限があります。 このフィールドは、[外部の認証情報ストア] チェックボックスがオンになっている場合にのみ表示されます。
認証情報エイリアス	ワークフロー作成者は、個々の認証情報をオーケストレーションワークフロー内の任意のアクティビティに割り当てたり、オーケストレーションワークフロー内で同じアクティビティタイプが発生するたびに異なる認証情報を割り当てたりすることができます。

フィールド	説明
	サービスマッピングおよびディスカバリーパターンを使用してこの CI タイプに属していない CI を検出するために認証情報を使用するには、CI が属する CI タイプのテーブル名 (cmdb_ci_apache_web_server など) を入力します。
外部の認証情報ストア	外部認証情報ストレージシステムを使用するには、このチェックボックスをオンにします。このオプションを選択すると、[ユーザー名] フィールドと [パスワード] フィールドが [認証情報 **ID]** フィールドに置き換えられます。外部の認証情報ストレージは、外部認証情報ストレージ プラグインをアクティブ化した場合にのみ使用できます。 注: 現在サポートされている外部ストレージ システムは CyberArk だけです。
適用先	これらの認証情報をネットワーク内の [すべての **MID** サーバー] または 1 つ以上の [特定の **MID** サーバー] に適用するかどうかを選択します。**[MID** サーバー] フィールドでこれらの認証情報を使用する MID サーバー を指定します。
MID サーバーを使用	使用可能な MID サーバーのリストから 1 つ以上の MID サーバーを選択します。このレコードで設定された認証情報は、このリストの MID サーバーで使用できます。このフィールドは、[適用先] フィールドで [特定の **MID** サーバー] を選択した場合のみ使用できます。 注: [特定の **MID** サーバー] を選択しても、MID サーバーの選択には影響しません。これは、認証情報を表示する必要がある MID サーバーを決定するためにのみ使用されます。[特定の **MID** サーバー] はオーケストレーションアクティビティではサポートされていません。
順序	ディスカバリー がデバイスにログオンしようとするときに、この認証情報を試行する順序 (シーケンス)。番号が小さいほど、この認証情報がリストの上位に表示されます。多くの認証情報を使用する場合、またはログイン試行が 3 回失敗してセキュリティによってユーザーがロックアウトされた場合、認証情報の順序を確立します。すべての認証情報の順序番号が同じ (または順序番号がない) 場合、インスタンスは認証情報をランダムな順序で試行します。
Windows MID サーバー サービスアカウント	有効な場合、定義された認証情報は MID サーバー サービスアカウントを表します。

SNMPv3 認証情報

SNMPv3 認証情報は、プライバシープロトコルと追加のプライバシーキーを受け入れ、ディスカバリーとオーケストレーションで使用できます。CyberArk の外部ストレージでは、プライバシーアカウントキーを選択できます。

次のフィールドは、SNMPv3 の認証情報フォームで使用できます。

SNMPv3 認証情報フィールド

フィールド	入力値
名前	この認証情報にわかりやすい一意の名前。たとえば、**SNMP Community Atlanta** と入力します。

SNMPv3 認証情報フィールド (続く)

フィールド	入力値
有効	これらの認証情報の使用を有効または無効にします。
適用先	これらの認証情報をネットワーク内の [すべての **MID** サーバー] または 1 つ以上の [特定の **MID** サーバー] に適用するかどうかを選択します。**[MID** サーバー] フィールドでこれらの認証情報を使用する MID サーバーを指定します。
MID サーバーを使用	使用可能な MID サーバーのリストから 1 つ以上の MID サーバーを選択します。このレコードで設定された認証情報は、このリストの MID サーバーで使用できます。このフィールドは、[適用先] フィールドで [特定の **MID** サーバー] を選択した場合のみ使用できます。
順序	ディスカバリー がデバイスにログオンしようとするときに、この認証情報を試行する順序 (シーケンス)。番号が小さいほど、この認証情報がリストの上位に表示されます。多くの認証情報を使用する場合、またはログイン試行が 3 回失敗してセキュリティによってユーザーがロックアウトされた場合、認証情報の順序を確立します。すべての認証情報の順序番号が同じ (または順序番号がない) 場合、インスタンスは認証情報をランダムな順序で試行します。
ユーザー名	SNMP ユーザー名を入力します。ユーザー名の先頭または末尾にはスペースを使用しないでください。プラットフォームでユーザー名の先頭または末尾のスペースが検出されると、警告が表示されます。
認証プロトコル	この認証情報に使用する認証タイプを選択します。以下から選択可能です。 • MD5 • SHA-1 • SHA-224 • SHA-256 • SHA-384 • SHA-512
認証キー	この認証情報に使用する認証キーを入力します。
プライバシー プロトコル	この認証情報の暗号化プロトコルを選択します。以下から選択可能です。 • 3DES • AES128 • AES192 • AES256 • DES
プライバシー キー	選択したプライバシー プロトコルに関連付けられたキーを入力します。

SNMPv3 認証情報フィールド (続く)

フィールド	入力値
認証情報 ID	外部認証情報プロバイダーに対して MID サーバーにアップロードされた JAR ファイルの外部認証情報用に設定された一意のキーを入力します。[認証情報 **ID]** フィールドには 40 文字の制限があります。 このフィールドは、[外部の認証情報ストア] チェックボックスがオンになっている場合にのみ表示されます。
プライバシー認証情報 ID (Privacy Credential ID)	CyberArk で SNMPv3 の認証情報用に設定されたプライバシーアカウントキーを入力します。CyberArk でプライバシープロトコルを使用している場合、このフィールドに CyberArk SNMPv3 プライバシーアカウントの [名前] フィールドと同じ値を指定する必要があります。このフィールドは、[認証情報ストアタイプ] フィールドから [**CyberArk**] を選択した場合に、SNMPv3 に対してのみ表示されます。CyberArk のプライバシーキーを使用していない場合は、このフィールドを空白のままにします。
認証情報エイリアス	ワークフロー作成者は、個々の認証情報をオーケストレーションワークフロー内の任意のアクティビティに割り当てたり、オーケストレーションワークフロー内で同じアクティビティタイプが発生するたびに異なる認証情報を割り当てたりすることができます。
外部の認証情報ストア	外部認証情報ストレージシステムを使用するには、このチェックボックスをオンにします。外部ストレージが有効になっている場合は、[認証情報 **ID**] フィールドが表示されます。外部の認証情報ストレージは、 外部認証情報ストレージプラグインをアクティブ化し、[外部ストレージ **(External Storage)**] ビューが選択されている場合にのみ使用できます。
認証情報ストアタイプ	外部ストレージプロバイダーを選択します。CyberArk SNMPv3 プライバシーキーを使用している場合のみ、[**CyberArk**]を選択します。[プライバシー認証情報 **ID (Privacy Credential ID)**] フィールドが表示され、キーのエントリーが許可されます。
コンテキストを使用	この認証情報のコンテキスト値を追加するには、このチェックボックスをオンにします。このフィールドは [ディスカバリー] ビューに表示されます。現在、外部認証情報ストレージではコンテキストはサポートされていません。 **注:** コンテキストは、特定の OID を参照する SNMPv3 認証情報によってアクセスされる管理情報の集合です。通常の認証情報ではアクセスできないデバイスに関する情報を収集するために、コンテキストが参照されることがあります。コンテキストは、メーカーが提供することも、個別に設定することもできます。同じユーザー名とキーを持つ複数の SNMPv3 認証情報がある場合、一部のデバイスにコンテキストが必要であっても、その他のデバイスに必要なければ、デバイスごとに個別のレコードを作成する必要があります。
コンテキスト名	この認証情報のコンテキスト名の値を入力します。これは、フルアクセスでこの値が必要なデバイスがある場合のみ使用する必要があります。このフィールドは、[コンテキストを使用 **(Use Context)**] チェックボックスがオンになっている場合にのみ表示されます。

自動翻訳

SSH 認証情報

ディスカバリー、オーケストレーション、統合ハブ では、SSH 認証情報を使用して UNIX デバイスと Linux デバイスを探索し、Secure Shell (SSH) を介してコマンドを実行します。SSH コマンドは、ルート認証情報または sudo を使用してルート権限で実行する必要があります。SSH 秘密鍵認証情報はセキュリティを強化します。

ルート権限の付与

ルート権限を付与する前に、組織内のセキュリティチームとともにセキュリティポリシーとオプションを確認してください。

次のいずれかの方法を使用すると、ユーザーはルート権限で SSH コマンドを実行することができます。

- ディスカバリー、オーケストレーション、統合ハブ の他の認証情報を指定しますが、これらの認証情報のユーザーには sudo を使用してルート権限で特定のコマンドを実行する権限を付与します。これによって、限られた特権を安全に付与できます。ディスカバリー、オーケストレーション、統合ハブ では、`must_sudo` パラメーターを [**true**] (デフォルトは [**false**]) に設定した任意のプローブで sudo を使用します。ただし、sudo が動作するように各システムを設定する必要があります。これを行うには、**visudo** コマンドを使用して **/etc/sudoers** ファイルを編集します。
- ルート認証情報を指定します。これらは明らかに最も強力な認証情報ですが、セキュリティの観点からは望ましくない場合があります。ディスカバリー、オーケストレーション、統合ハブ に任意の UNIX システムまたは Linux システムに対するルート認証情報がある場合は、それ以上構成する必要はありません。

特権コマンド

プラットフォームでは、MID サーバーで使用されるデフォルトの特権コマンドと、システムに他のコマンドを追加する機能が用意されています。sudo およびその他の特権コマンドの使用の詳細については、「MID サーバーの特権コマンド」を参照してください。

SSH 秘密鍵認証情報タイプ

注: ほとんどの場合、SSH 秘密鍵認証情報を使用する必要があります。これらは、2 者間の通信が傍受される MitM (中間者) 攻撃に対するものを含め、SSH パスワード認証情報よりも優れたセキュリティを提供します。

フィールド	入力値
名前	この認証情報にわかりやすい一意の名前。たとえば、**SSH Atlanta** と入力します。
アクティブ	これらの認証情報の使用を有効または無効にします。
ユーザー名	UNIX または Linux のユーザー名を入力します。ユーザー名の先頭または末尾にはスペースを使用しないでください。プラットフォームでユーザー名の先頭または末尾のスペースが検出されると、警告が表示されます。
パスワード	UNIX または Linux のパスワードを入力します。**[SSH** 秘密鍵] タイプの認証情報でユーザー名にパスワードが必要な場合は、sudo パスワードを入力します。
SSH パスフ	セキュアな SSH パスフレーズを入力します。このフィールドは、**[SSH** 秘密鍵] の認証情報にのみ使用できます。

フィールド	入力値
レーズ	

SSH 秘密鍵	セキュアな RSA、DSA、ECDSA、または ED25519 秘密キーを入力します。
	
	秘密鍵は、正しく暗号化されるように、適切な形式で入力する必要があります。秘密鍵は、文字列 `-----BEGIN` で始める必要があります。
	
	正しい形式の RSA 秘密鍵の例を次に示します。
	
	`-----BEGIN RSA PRIVATE KEY-----`
	`MIIEogIBAAKCAQEAsEK65scPssPSobpDFMpR+Btv3MS4Q7NP8ERaStRZsh3IWz+x...`
	`...7hrxV2dbSug60FahyupGWBGtPnXm5PaE2X5WPLuUj94ue48i1Fs`
	`-----END RSA PRIVATE KEY-----`
	
	DSA キーの例を次に示します。
	
	`-----BEGIN DSA PRIVATE KEY-----`
	`MIIEogIBAAKCAQEAsEK65scPssPSobpDFMpR+Btv3MS4Q7NP8ERaStRZsh3IWz+x...`
	`...7hrxV2dbSug60FahyupGWBGtPnXm5PaE2X5WPLuUj94ue48i1Fs`
	`-----END DSA PRIVATE KEY-----`
	
	ECDSA キーの例を次に示します。
	
	`-----BEGIN EC PRIVATE KEY-----`
	`MIIEogIBAAKCAQEAsEK65scPssPSobpDFMpR+Btv3MS4Q7NP8ERaStRZsh3IWz+x...`
	`...7hrxV2dbSug60FahyupGWBGtPnXm5PaE2X5WPLuUj94ue48i1Fs`
	`-----END EC PRIVATE KEY-----`
	
	ED25519 秘密鍵の例を次に示します。
	
	`-----BEGIN OPENSSH PRIVATE KEY-----`
	`b3BlbnNzaC1rZXktdjEAAAAABG5vbmUAAAAEbm9uZQAAAAAAAAABAAAAMwAAAAtzc2gtZW`
	`QyNTUxOQAAACAIYIqhcdwx8VQzZ5XaIC5ltQpjRr3lIIq/aE66mufmiwAAAKDQUtxZ0FLcWQAAAAtzc2gtZWQyNTUxOQAAACAIYIqhcdwx8VQzZ5XaIC5ltQpjRr3lIIq/aE66mufmiwAAAECuvsTkFUPdpTh0kw23i8TYx19qsFOZ3TRgowkkHBh6wSViWqFx3DHxVDNnIdogLmW1`
	`CmNGveUiWr9oTrqa5+aLAAAAGmFiaGluYXYuc3V0YXJATVJFTUE3OTAzMkI3AQID`
	`-----END OPENSSH PRIVATE KEY-----`
	
	注: ED25519 秘密鍵の場合は、OpenSSH SSH-keygen ユーティリティを使用して生成される OpenSSH キー形式のみがサポートされます。
	
	ServiceNow AI Platform では、OpenSSH ssh-keygen ユーティリティで生成された PEM 形式の秘密鍵をサポートしています。PuTTY によって生成された PPK キーを変換するには、次の手順に従います。
	
	• PuTTYGen で秘密鍵を開きます。
	• メニューから OpenSSH 形式でエクスポートします 変換 > **OpenSSH** キーをエクスポート.
	• 新しい OpenSSH キーを保存します。

自動翻訳

フィールド	入力値
SSH 証明書	RSA または ED25519 ベースの OpenSSH 証明書を入力します。証明書を入力すると、証明書ベースの認証に秘密キーが使用されます。この認証は OpenSSH 7.8 以降でサポートされています。
認証情報エイリアス	• フローデザイナーが別名を使用して接続と資格情報を管理できるようにします。エイリアスを使用することで、複数の環境を使用する場合に、複数の認証情報と接続情報プロファイルを構成する必要がなくなります。接続または認証情報が変更された場合、接続を使用するどのアクションも更新する必要はありません。詳細については、「接続と資格情報」を参照してください。 • ワークフロー作成者は、個々の認証情報をオーケストレーションワークフロー内の任意のアクティビティに割り当てたり、オーケストレーションワークフロー内で同じアクティビティタイプが発生するたびに異なる認証情報を割り当てたりすることができます。
外部の認証情報ストア	外部認証情報ストレージシステムを使用するには、このチェックボックスをオンにします。このオプションを選択すると、[ユーザー名] フィールドと [パスワード] フィールドが [認証情報 **ID]** フィールドに置き換えられます。現在サポートされている外部ストレージ システムは CyberArk だけです。
MID サーバーを使用	使用可能な MID サーバーのリストから 1 つ以上の MID サーバーを選択します。このレコードで設定された認証情報は、このリストの MID サーバーで使用できます。このフィールドは、[適用先] フィールドで [特定の **MID** サーバー] を選択した場合のみ使用できます。
適用先	これらの認証情報をネットワーク内の [すべての **MID** サーバー] または 1 つ以上の [特定の **MID** サーバー] に適用するかどうかを選択します。**[MID** サーバー] フィールドでこれらの認証情報を使用する MID サーバーを指定します。
順序	プラットフォームがデバイスにログオンしようとするときにこの認証情報を試行する順序 (シーケンス)。番号が小さいほど、この認証情報がリストの上位に表示されます。多くの認証情報を使用する場合、またはログイン試行が 3 回失敗してセキュリティによってユーザーがロックアウトされた場合、認証情報の順序を確立します。すべての認証情報に同じ順序番号を設定した場合、または何も設定しない場合、ディスカバリーまたはオーケストレーションでは、認証情報をランダムな順序で試行します。

SSH 認証情報タイプ

次のフィールドは、SSH 認証情報フォームで使用できます。

フィールド	説明
名前	この認証情報のわかりやすい一意の名前を入力します。
アクティブ	これらの認証情報の使用を有効または無効にします。
ユーザー名	[認証情報] テーブルに、作成するユーザー名を入力します。ユーザー名の先頭または末尾にはスペースを使用しないでください。プラットフォームでユーザー名の先頭または末尾のスペースが検出されると、警告が表示されます。CIM ディスカバリーの場合、ユーザーには admin ロールが必要です。

フィールド	説明
パスワード	パスワードを入力します。
認証情報 ID	外部認証情報システム用に MID サーバーにアップロードされた JAR ファイルに外部認証情報用に設定された一意のキーを入力します。[認証情報 **ID]** フィールドには 40 文字の制限があります。 このフィールドは、[外部の認証情報ストア] チェックボックスがオンになっている場合にのみ表示されます。
認証情報エイリアス	• フローデザイナーが別名を使用して接続と資格情報を管理できるようにします。エイリアスを使用することで、複数の環境を使用する場合に、複数の認証情報と接続情報プロファイルを構成する必要がなくなります。接続または認証情報が変更された場合、接続を使用するどのアクションも更新する必要はありません。詳細については、「接続と資格情報」を参照してください。 • ワークフロー作成者は、個々の認証情報をオーケストレーションワークフロー内の任意のアクティビティに割り当てたり、オーケストレーションワークフロー内で同じアクティビティタイプが発生するたびに異なる認証情報を割り当てたりすることができます。サービスマッピングおよびディスカバリーパターンを使用してこの CI タイプに属していない CI を検出するために認証情報を使用するには、CI が属する CI タイプのテーブル名 (cmdb_ci_apache_web_server など) を入力します。
外部の認証情報ストア	外部認証情報ストレージシステムを使用するには、このチェックボックスをオンにします。このオプションを選択すると、[ユーザー名] フィールドと [パスワード] フィールドが [認証情報 **ID]** フィールドに置き換えられます。外部の認証情報ストレージは、外部認証情報ストレージ プラグインをアクティブ化した場合にのみ使用できます。 **注:** 現在サポートされている外部ストレージ システムは CyberArk だけです。
適用先	これらの認証情報をネットワーク内の [すべての **MID** サーバー] または 1 つ以上の [特定の **MID** サーバー] に適用するかどうかを選択します。**[MID** サーバー] フィールドでこれらの認証情報を使用する MID サーバー を指定します。
MID サーバーを使用	使用可能な MID サーバーのリストから 1 つ以上の MID サーバーを選択します。このレコードで設定された認証情報は、このリストの MID サーバーで使用できます。このフィールドは、[適用先] フィールドで [特定の **MID** サーバー] を選択した場合のみ使用できます。
順序	ディスカバリー がデバイスにログオンしようとするときに、この認証情報を試行する順序 (シーケンス)。番号が小さいほど、この認証情報がリストの上位に表示されます。多くの認証情報を使用する場合、またはログイン試行が 3 回失敗してセキュリティによってユーザーがロックアウトされた場合、認証情報の順序を確立します。すべての認証情報の順序番号が同じ (または順序番号がない) 場合、インスタンスは認証情報をランダムな順序で試行します。

ディスカバリー、オーケストレーション、統合ハブ のルート権限が必要なコマンド

次の例では、ユーザー名が **Disco** であることを前提とします。実際のユーザー名に置き換えて、コマンドのパスがシステム上のパスと一致することを確認してください。

注: sudo コマンドに入力するパスワードがないため、sudo コマンドは秘密鍵認証情報と連携しません。解決策は、*NOPASSWD* オプションを sudo 構成に追加することです。たとえば、disco ALL=(root) NOPASSWD:/usr/sbin/dmidecode,/usr/sbin/lsof,/sbin/ifconfig と入力します。

ルート権限が必要な **UNIX** コマンドおよび **Linux** コマンド

コマンド	目的
HP-UX	
adb	CPU の速度とメモリーを収集します。 • [**/etc/sudoers** 行の例]：Disco ALL=(root) /usr/bin/adb • [使用者]：ディスカバリー
Linux および **UNIX** のすべてのバージョン	
chage	前回のパスワード変更日からパスワードを変更するまでの日数を変更します。 • [**/etc/sudoers** 行の例]：Disco ALL=(root) /usr/bin/chage • [使用者]：オーケストレーションと 統合ハブ
chpasswd	ユーザーパスワードを変更します。 • [**/etc/sudoers** 行の例]：Disco ALL=(root) /etc/chpasswd • [使用者]：オーケストレーションと 統合ハブ
すべての **Linux**	
dmidecode	マザーボード内に埋め込まれたシリアル番号など、ハードウェアに関するいくつかの情報を収集します。 • [**/etc/sudoers** 行の例]：Disco ALL=(root) /sbin/dmidecode • [使用者]：ディスカバリー
fdisk	システム上のディスクおよびサイズ情報を収集します。 • [**/etc/sudoers** 行の例]：Disco ALL=(root) /usr/bin/fdisk -l • [使用者]：ディスカバリー
multipath	MPIO のデバイス マッピングを収集します。 • [**/etc/sudoers** 行の例]：Disco ALL=(root) /usr/bin/multipath -ll • [使用者]：ディスカバリー
ls	ディレクトリーの内容を収集します。 • **/etc/sudoers** 行の例:Disco ALL=(root) /usr/bin/ls, /bin/ls • [使用者]：ディスカバリー

ルート権限が必要な **UNIX** コマンドおよび **Linux** コマンド (続く)

コマンド	目的
Linux および **Solaris**	
dmsetup	低レベルのボリュームを調べます。 • [**/etc/sudoers** 行の例]： ◦ Disco ALL=(root) /usr/bin/dmsetup table * ◦ Disco ALL=(root) /usr/bin/dmsetup ls • [使用者]：ディスカバリー
UNIX のすべてのバージョン	
lsof	プロセスとシステムへの接続との関係を決定します。 • [**/etc/sudoers** 行の例]：Disco ALL=(root) /sbin/lsof • [使用者]：ディスカバリー
oratab	Oracle ホームおよび pfile を検索するために oratab ファイルへの読み取りアクセス権を付与します。 • [**/etc/sudoers** 行の例]：N/A • [使用者]：ディスカバリー
Solaris	
iscsiadm	iSCSI IQNs を取得 • [**/etc/sudoers** 行の例]：${sudo:iscsiadm list target -S} • [使用者]：ディスカバリー
fcinfo	ポートの WWPN を取得します。 • [**/etc/sudoers** 行の例]：${sudo:fcinfo remote-port -sl -p $port} • [使用者]：ディスカバリー
prtvtoc	ディスク パーティションに関する情報をレポートします。 • [**/etc/sudoers** 行の例]：Disco ALL=(root) /usr/bin/prtvtoc • [使用者]：ディスカバリー
pfiles	TCP 接続情報を収集する目的で使用されます。 • **/etc/sudoers** 行の例：Disco ALL=(root) /usr/bin/pfiles • [使用者]：ディスカバリー

ルート権限が必要な **UNIX** コマンドおよび **Linux** コマンド (続く)

コマンド	目的
pgrep	pfiles を実行する特定の領域のプロセス ID をリスト化する目的で使用されます。 • **/etc/sudoers** 行の例：Disco ALL=(root) /usr/bin/pgrep • [使用者]：ディスカバリー
/usr/bin/ps	実行中のプロセスを一覧表示します。ルート アクセスで実行する代わりに、proc_owner ロールを追加します。 • [**/etc/sudoers** 行の例]：Disco ALL=(root) /usr/bin/ps • [使用者]：ディスカバリー
/usr/ucb/ps	実行中のプロセスを一覧表示します。ルート アクセスで実行する代わりに、proc_owner ロールを追加します。 /usr/ucb/ps コマンドの使用は、Solaris 11 の時点で廃止となっています。ディスカバリー、オーケストレーション、統合ハブ では、Solaris のすべてのバージョンに対してこのコマンドを使用しなければならないため、Solaris 11 システムに ucb ユーティリティを手動でインストールする必要があります。手順については、「KB0564262 」を参照してください。 • [**/etc/sudoers** 行の例]：Disco ALL=(root) /usr/ucb/ps • [使用者]：ディスカバリー

ディスカバリーとサービスマッピングに必要な特権コマンドのリストについては、「 特権ユーザーを必要とする サービスマッピング コマンド 」を参照してください。このリストには、組織内の Unix ベースのホストを検出およびマッピングするために昇格された権限が必要なコマンドが含まれています。

ルート以外の認証情報のアクセス要件

ディスカバリー にルート アクセスの認証情報を指定しない場合は、次のアクセス要件で認証情報を指定する必要があります。

アプリケーション	ファイルまたはディレクトリー	必要なアクセス権
Apache	httpd.conf	読み取り
Hbase	hbase-site.xml	読み取り
JBoss	jboss-service.xml	読み取り
	JBoss ホーム ディレクトリー	読み取り
	web.xml	読み取り
MySQL	my.cnf	読み取り
NGINX	nginx.conf	読み取り

アプリケーション	ファイルまたはディレクトリー	必要なアクセス権
Oracle	oratab	読み取り
	関連する pfile	読み取り
Oracle リスナー	lsnrctl	実行
	listener.ora	読み取り
Tomcat	catalina.jar	読み取り
	server.xml	読み取り
	web.xml	読み取り
Unix	/etc/*release	読み取り
	/etc/bashrc	読み取り
	/etc/profile	読み取り
	/proc/cpuinfo	読み取り
	/proc/vmware/sched/ncpus	読み取り
	/var/log/dmesg	読み取り
	APD ディレクトリー	読み取り
WebSphere	cell.xml	読み取り
	server.xml	読み取り
	serverindex.xml	読み取り

VMware 認証情報

VMware 認証情報タイプは、vCenter 認証情報へのアクセスを管理します。

VMware クラウドリソースにアクセスするアプリケーションは、VMware 認証情報にアクセスする必要があります。たとえば、ディスカバリーでは、VMware 認証情報タイプを使用すると、Windows マシン上で実行されている VMware の vCenter を検索して、ESX マシン、仮想マシン、およびリソースプールを検出できます。VMware ディスカバリー および自動化 API (vCenter API) により、コンピューター CI に対してグローバルに一意なシリアル番号が提供されます。各 VMware ホストへのアクセスを有効にするために、CIM 認証情報は必要ありません。

注: 有効な VMware 認証情報を使用している場合、vCenter ディスカバリー に Windows 認証情報は不要です。

重要: vCenter によってクローンされた個々の仮想マシン上で作業 (Linux VM の再起動など) を実行するオーケストレーション アクティビティには、**[VMware]** タイプの認証情報を使用しないでください。これらのアクティビティの場合、認証情報の [種類] は仮想マシンのオペレーティング システム (**[SSH]** または **[Windows]**) によって決まります。

[VMWare 認証情報] フォーム

フィールド	説明
名前	VMware 認証情報のわかりやすい一意の名前を入力します。
アクティブ	これらの認証情報の使用を有効または無効にします。

[VMWare 認証情報] フォーム (続く)

フィールド	説明
ユーザー名	VMware アカウントに使用するユーザー名を入力します。ユーザー名の先頭または末尾にはスペースを使用しないでください。プラットフォームでユーザー名の先頭または末尾のスペースが検出されると、警告が表示されます。 VMware 認証情報は、vCenter で読み取り専用ロールを持っている必要があります。
パスワード	VMware アカウントのパスワードを入力します。
適用先	使用可能な MID サーバーのリストから 1 つ以上 MID サーバーを選択します。このレコードで設定された認証情報は、このリストの MID サーバーで使用できます。このフィールドは、[適用先] フィールドで [特定の **MID** サーバー] を選択した場合のみ使用できます。
順序	ディスカバリー がデバイスにログオンしようとするときに、この認証情報を試行する順序 (シーケンス)。番号が小さいほど、この認証情報がリストの上位に表示されます。多くの認証情報を使用する場合、またはログイン試行が 3 回失敗してセキュリティによってユーザーがロックアウトされた場合、認証情報の順序を確立します。すべての認証情報の順序番号が同じ (または順序番号がない) 場合、インスタンスは認証情報をランダムな順序で試行します。

Windows 資格情報

Windows 資格情報により、Windows コンピューターへのアクセス権が付与されます。この認証情報タイプはディスカバリーとオーケストレーションで使用できます。

認証情報の要件

ディスカバリーとオーケストレーションには、Windows 認証情報に関して次の要件があります。

- MID サーバーを Windows ホストにサービスとしてインストールする。
- 次のいずれかの場所に Windows 認証情報を追加する。
  - 認証情報 [windows_credentials] テーブルのエントリー
  - 特定の Windows ユーザーまたはドメイン アカウントとして実行する MID サーバーサービスアカウント

適切な権限の付与

十分な権限を与えるには、 Windows 認証情報は次のいずれかでなければなりません。

- ターゲット の Windows ホストでローカルアドミニストレーターアクセス権を持つドメイン ユーザー。
- 同じターゲット ホストでアドミニストレーター権限とユーザー アクセス制御 (UAC) が無効になっているローカル アカウント。
- Windows プローブとパーミッション の要件を満たすユーザー (ディスカバリー のみ)。
- 実行するオーケストレーションアクティビティの要件を満たすユーザー (オーケストレーションのみ)。

注: ログオン権限は必要ありません。アカウントはインタラクティブである必要はありません。

特権アクセスの付与に関するセキュリティは、JEA プロファイルを使用して ディスカバリー を実行することで強化できます。詳細については、「ディスカバリー用 Microsoft Just Enough Administration (JEA)」を参照してください。

ワークグループ コンピューター

Powershell コマンドを実行してワークグループ コンピューターを検出するには、次のいずれかのユーザーに対して MID サーバーの認証情報を設定します。

- ワークグループ コンピューターに組み込まれたアドミニストレーターアカウント。
- ワークグループ コンピューターのドメイン ユーザー。

マルチドメイン構成

Windows 資格情報を複数のドメイン間で機能させるには、正しい名前形式と MID サーバー構成を使用していることを確認してください。

ディスカバリーとオーケストレーションでは、Windows ドメイン認証情報をユーザープリンシパル名とダウンレベル ログオン名の両方のユーザー名形式でサポートしています。たとえば、**Domain\UserName** または **UserName@example.domain.com** です。WORKGROUP\UserName の形式で Windows ワークグループの認証情報を入力できます。

注: また、**.** ユーザー名を使用してローカル アカウントを入力することもできます。

これらの追加の操作は、認証情報を複数の Windows ドメインにわたって機能させるために必要です。

条件	必要な他のアクション
Windows ターゲットと同じドメイン上の MID サーバーホスト。	なし
Windows ターゲットとは異なるドメイン上の MID サーバーホスト。	PowerShell 3.0 (以降 5.1 以前) が MID サーバーホストにインストールされていることを確認してください。
Microsoft SQL Server ターゲットとは異なるドメイン上の MID サーバーホスト。	「MSSQL サーバーのディスカバリー」を参照してください。

Windows 認証情報タイプ

次のフィールドは、Windows の認証情報フォームで使用できます。

フィールド	説明
名前	この認証情報のわかりやすい一意の名前を入力します。
アクティブ	これらの認証情報の使用を有効または無効にします。

フィールド	説明
ユーザー名	[認証情報] テーブルに、作成するユーザー名を入力します。ユーザー名の先頭または末尾にはスペースを使用しないでください。プラットフォームでユーザー名の先頭または末尾のスペースが検出されると、警告が表示されます。CIM ディスカバリーの場合、ユーザーには admin ロールが必要です。
パスワード	パスワードを入力します。
認証情報 ID	外部認証情報システム用に MID サーバーにアップロードされた JAR ファイルに外部認証情報用に設定された一意のキーを入力します。[認証情報 **ID]** フィールドには 40 文字の制限があります。 このフィールドは、[外部の認証情報ストア] チェックボックスがオンになっている場合にのみ表示されます。
認証情報エイリアス	ワークフロー作成者は、個々の認証情報をオーケストレーションワークフロー内の任意のアクティビティに割り当てたり、オーケストレーションワークフロー内で同じアクティビティタイプが発生するたびに異なる認証情報を割り当てたりすることができます。 サービスマッピングおよびディスカバリーパターンを使用してこの CI タイプに属していない CI を検出するために認証情報を使用するには、CI が属する CI タイプのテーブル名 (cmdb_ci_apache_web_server など) を入力します。
外部の認証情報ストア	外部認証情報ストレージシステムを使用するには、このチェックボックスをオンにします。このオプションを選択すると、[ユーザー名] フィールドと [パスワード] フィールドが [認証情報 **ID]** フィールドに置き換えられます。外部の認証情報ストレージは、外部認証情報ストレージ プラグインをアクティブ化した場合にのみ使用できます。 **注:** 現在サポートされている外部ストレージ システムは CyberArk だけです。
適用先	これらの認証情報をネットワーク内の [すべての **MID** サーバー] または 1 つ以上の [特定の **MID** サーバー] に適用するかどうかを選択します。**[MID** サーバー] フィールドでこれらの認証情報を使用する MID サーバー を指定します。
MID サーバーを使用	使用可能な MID サーバーのリストから 1 つ以上の MID サーバーを選択します。このレコードで設定された認証情報は、このリストの MID サーバーで使用できます。このフィールドは、[適用先] フィールドで [特定の **MID** サーバー] を選択した場合のみ使用できます。 **注:** [特定の **MID** サーバー] を選択しても、MID サーバーの選択には影響しません。これは、認証情報を表示する必要がある MID サーバーを決定するためにのみ使用されます。[特定の **MID** サーバー] はオーケストレーションアクティビティではサポートされていません。
順序	ディスカバリー がデバイスにログオンしようとするときに、この認証情報を試行する順序 (シーケンス)。番号が小さいほど、この認証情報がリストの上位に表示されます。多くの認証情報を使用する場合、またはログイン試行が 3 回失敗してセキュリティによってユーザーがロックアウトされた場合、認証情報の順序を確立します。すべての認証情報の順序番号が同じ (または順序番号がない) 場合、インスタンスは認証情報をランダムな順序で試行します。

フィールド	説明
Windows MID サーバー サービスアカウント	有効な場合、定義された認証情報は MID サーバー サービスアカウントを表します。

MID サーバー用の Windows 資格情報の設定

独自の Windows サービスの認証情報または認証情報 [discovery_credentials] テーブルの認証情報を使用するように MID サーバーを設定します。

始める前に
必要なロール：admin

手順

1. MID サーバーサービスアカウントの認証情報を使用するように MID サーバーを設定します。
   a. [Windows MID サーバーサービスの認証情報の設定] を、権限の要件を満たすユーザーに設定します。
   b. ユーザー名が名前形式の要件を満たしていることを確認します。
   c. 必要に応じて、フォームのフィールドに入力します。
   d. 認証情報がドメイン要件を満たしていることを確認します。
2. 認証情報 [discovery_credentials] テーブルの認証情報を使用するように MID サーバーを設定します。

   a. 個々の Windows 認証情報を認証情報 [windows_credentials] テーブルに追加します。
      - 各認証情報が権限の要件を満たしていることを確認します。
      - 各ユーザー名が名前形式の要件を満たしていることを確認します。
      - 各認証情報が Windows ドメイン要件を満たしていることを確認します。

   b. オプション: mid.use_powershell パラメーターを **true** に設定することで、PowerShell を使用するように MID サーバーを設定します。
   「MID サーバーの構成」を参照してください。

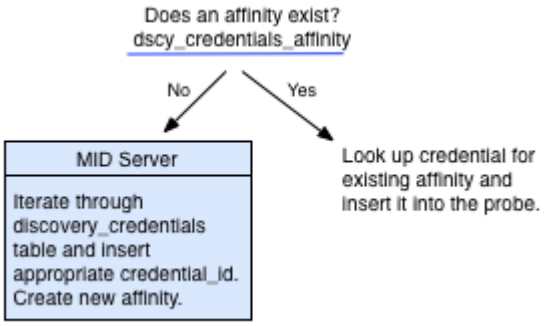
   c. 特定の Windows ユーザーまたはドメインアカウントとして実行する Windows MID サーバーサービスアカウントを表す認証情報を作成するには、[Windows MID サーバーサービスアカウント (Windows MID Server Service Account)] チェックボックスをオンにします。

ディスカバリーとオーケストレーションの認証情報親和性

認証情報親和性とは、一連の認証情報とネットワーク上のデバイス間の関連付けです。

ディスカバリー またはオーケストレーションで初めてデバイスにアクセスしようとすると、正しい認証情報が見つかるまで利用可能なすべての認証情報が試行されます。デバイスの認証情報を特定した後、ディスカバリー とオーケストレーションでは、認証情報親和性 [dscy_credentials_affinity] テーブルを使用して認証情報とデバイス間の親和性が作成されます。後続のすべての検出またはオーケストレーション アクティビティで、このテーブルの認証情報が、親和性が存在するデバイスと照合されます。デバイスの認証情報が変更された場合、 ディスカバリー とオーケストレーションでは、新しい親和性を作成するまで利用可能な認証情報をすべて再試行します。

認証情報親和性の図



注: オーケストレーションと ディスカバリー がインストールされ、認証情報エイリアスが有効になっている場合、複数の親和性が存在する可能性があります。この場合、プラットフォームでは、親和性ごとに認証情報が検索され、順序が最も低い親和性の認証情報がプローブに挿入されます。

認証情報のトラブルシューティング

ECC キュー ペイロードの <credentials_debug> セクションをレビューして、認証情報に関する問題をトラブルシューティングします。

特定のプローブでは認証情報のデバッグがサポートされています。認証情報のデバッグにより、MID サーバーが返すペイロードの <credentials_debug> セクションがインスタンス ECC キューに挿入されます。<credentials_debug> セクションを表示して、認証情報ルックアップに関する詳細を確認できます。

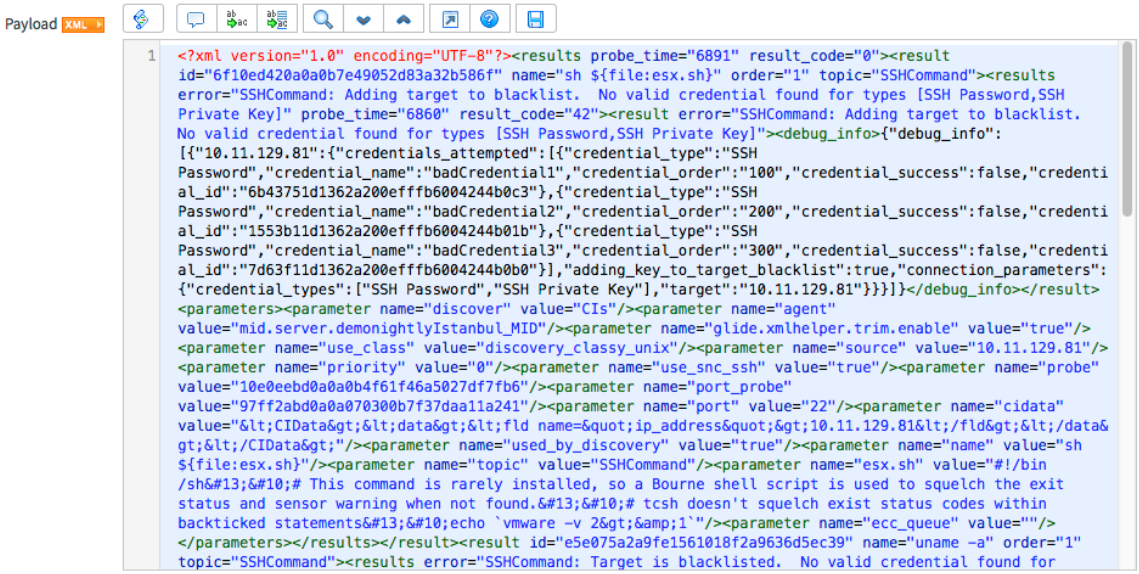
次の場合に、<credentials_debug> セクションがペイロードに表示されます。

- WMIRunner プローブ、 PowerShell プローブ、 JMS、または SSHCommand プローブ の認証情報が失敗した場合。
- WMIRunner プローブ、PowerShell プローブ、または SSHCommand プローブの *credentials_debug* パラメーターを **true** に設定します。そのパラメーターを true に設定すると、認証情報ルックアップが成功した場合でも、<credentials_debug> セクションが表示されます。

<credentials_debug> セクションに表示される内容は次のとおりです。

- 認証情報のタイプ、タグ、親和性など、認証情報の検索に関する情報。
- ターゲットとなる IP アドレス。
- タイプ、分類、タグ、名前、システム ID、外部認証情報 ID (存在する場合) など、MID サーバーが使用した各認証情報 (順番に) に関する情報。

無効な認証情報を示すサンプル ペイロード



以下の場合、PowerShell パラメーターの詳細が表示されます。

- すべての Windows 資格情報が失敗した後にローカル MID サーバー認証情報が使用された場合。
- MID サーバーがオンになっている同じマシンを検出しようとしているために認証情報がスキップされた場合。
- mid.powershell.use_credentials パラメーターが `true` に設定されている場合。

以下の場合、SSHCommand の詳細が表示されます。

- ターゲット IP が除外されているために認証情報の検索がスキップされた場合。
- ターゲット IP が除外リストに追加された場合。

注: MID サーバーでは、失敗した認証情報検索の IP アドレスがキャッシュメモリーの除外リストに保存されます。この除外リストにより、MID サーバーでアクセスの試行を停止するデバイスが指定されます。すべての認証情報が失敗すると、IP アドレスが除外リストに追加されます。IP アドレスは、MID サーバーが再起動された場合、またはインスタンスの認証情報レコードが更新された場合のいずれかで、5 分後に除外リストキャッシュから消去されます。

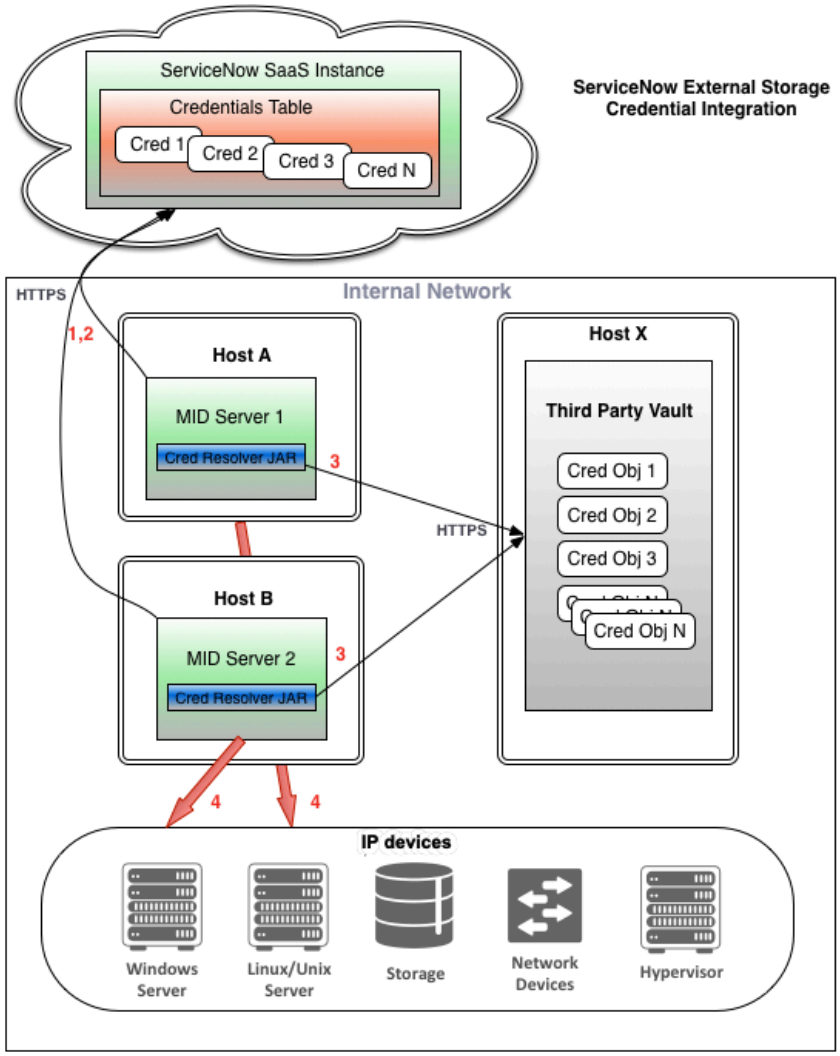
外部認証情報ストレージ

インスタンスは、 ディスカバリー、オーケストレーション、および サービスマッピング で使用される認証情報を ServiceNow 認証情報レコードに直接保存する代わりに、外部認証情報リポジトリーに保存することができます。

インスタンスは、各認証情報の一意の識別子、認証情報タイプ (SSH、SNMP、Windows など)、および任意の認証情報親和性を保持します。MID サーバーは、インスタンスから認証情報識別子を取得してから、顧客が提供した JAR ファイルを使用してその識別子をリポジトリーから使用可能な認証情報に解決します。現在、ServiceNow プラットフォームでは、外部認証情報ストレージとして、CyberArk ボールトまたは BeyondTrust の使用がサポートされています。

外部認証情報ストレージのアーキテクチャ

外部認証情報ストレージのアーキテクチャ



認証情報プロセス フロー

MID サーバーは、次のプロセスを使用して外部ストアから認証情報を取得します。

1. MID サーバーは、ターゲット ボールトの対応する認証情報 ID が含まれている ServiceNow 認証情報 [discovery_credentials] から認証情報オブジェクトをダウンロードします。
2. 各プローブまたはパターンが ディスカバリー または オーケストレーション ジョブから実行されると、MID サーバーは、認証情報 ID、ターゲット IP アドレス、認証情報タイプなどの情報を認証情報リゾルバー Java Jar ファイルに渡すことによって認証情報を要求します。ボールトから取得する正しい認証情報オブジェクトに関する詳細は、認証情報リゾルバーによって決定されます。

   CyberArk などの多くの認証情報リゾルバーは、MID サーバーと同じマシン上で実行されているサードパーティのボールトベンダーが提供するアプリケーションを呼び出します。多くの場合、このアプリケーションは認証情報をキャッシュするように構成でき、ボールトで認証情報が変更されたときにキャッシュを更新させることができます。これは、MID サーバーが認証情報を要求するたびにボールトへの不要なネットワーク呼び出しを回避するために非常に重要です。認証情報リゾルバー (オプションのベンダーアプリケーションが存在する場合はそれを使用) はボールトを呼び出し、実際のユーザー名やパスワードなどを取得します。

すぐに利用可能な認証情報リゾルバー (現在は CyberArk のみ) の場合、MID サーバーは、MID サーバープロセスメモリーの暗号化を使用して、最大数秒間だけ認証情報をキャッシュします。これは、単一のデバイスを検出する場合でも、MID サーバーが同じ認証情報の認証情報リゾルバーに対して複数の要求を行うことができることを意味します。他の認証情報リゾルバーのキャッシュ実装については、サードパーティベンダーにお問い合わせください。

3. MID サーバーは、適切な認証情報を使用してプローブを実行します。

注: 認証情報親和性は引き続き適用されます。メカニズムは同じままです。MID サーバーの観点との唯一の違いは、実際の認証情報の詳細 (ユーザー名とパスワード) がサードパーティのボールトから取得される点です。

外部認証情報ストレージのログ記録

MID サーバーから外部認証情報ストレージに関する関するログ メッセージが投稿されます。

認証情報要求を解決しようとしているときにリポジトリにエラーが発生した場合、MID サーバーは「クライアントの CredentialResolver に関する問題：」というプリフィックスが付いたログメッセージを送信します。

外部認証情報ストレージとともにインストールされるコンポーネント

ビジネスルール

外部認証情報ストレージのビジネスルールは、アドミニストレーターが [外部認証情報ストレージの有効化] プロパティを変更すると、次のタスクを実行します。

- 認証情報レコードリストおよびフォームのビューを外部ストレージビューに変更します。このビューを使用すると、ユーザーはリスト内の [認証情報 **ID]** 列を確認できます。
- MID サーバーに、認証情報の取得方法の変更に備えて、認証情報キャッシュを更新するように指示します。

プロパティ

外部認証情報ストレージの有効化 [com.snc.use_external_credentials] と呼ばれるプロパティは、External Credential Storage プラグインがアクティブ化された後で、それを有効または無効にします。プロパティの場所: ディスカバリー定義 > プロパティ および オーケストレーション > **MID** サーバープロパティで、プラグインをアクティブ化すると有効になります。

システムプロパティを使用して外部認証情報ストレージを無効にする場合、システムは自動的にすべての外部認証情報をインスタンス内で非アクティブ化します。このプロパティを使用して機能を再度有効にしても、外部認証情報レコードはアクティブにリセットされません。それぞれの認証情報レコードを手動で再アクティブ化する必要があります。

ディスカバリーおよびオーケストレーション用の外部認証情報ストレージの要求

外部認証情報ストレージ プラグインは、リクエストにより利用可能です。

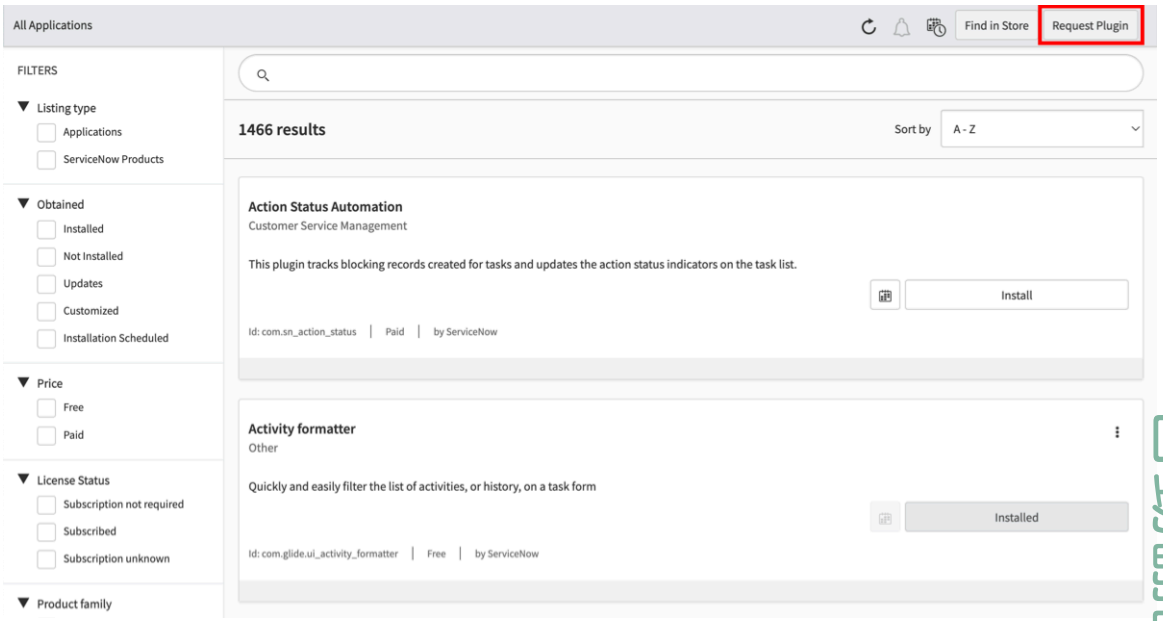
始める前に
必要なロール：admin

このタスクについて
プラグインを要求するには、次の 2 つの方法があります。

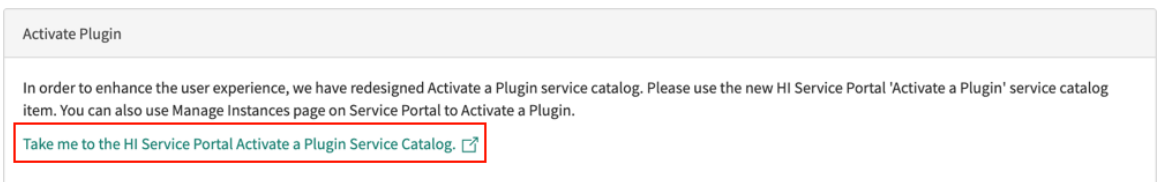
- Now Supportサービスカタログに直接アクセスするには、次を選択します。 すべて > サービスカタログ > プラグインをアクティブ化 Now Supportに。
- 次の手順に従い、インスタンスの [すべてのアプリケーション] ページから Now Support サービスカタログにアクセスします。

手順

1. 移動先 すべて > システムアプリケーション > 利用可能なすべてのアプリケーション > すべて.
2. [すべてのアプリケーション] ページで [プラグインの要求] をクリックして、Now Support で [プラグインをアクティブ化] フォームを開きます。



3. Now Support で、Now Support サービスポータル サービスカタログ にアクセスするリンクを選択します。



4. インスタンスを選択します。
5. [アクション] > [プラグインのアクティブ化] を選択します。
6. [プラグインのアクティブ化] フォームで、次の情報を入力します。

[プラグインのアクティブ化] フォーム

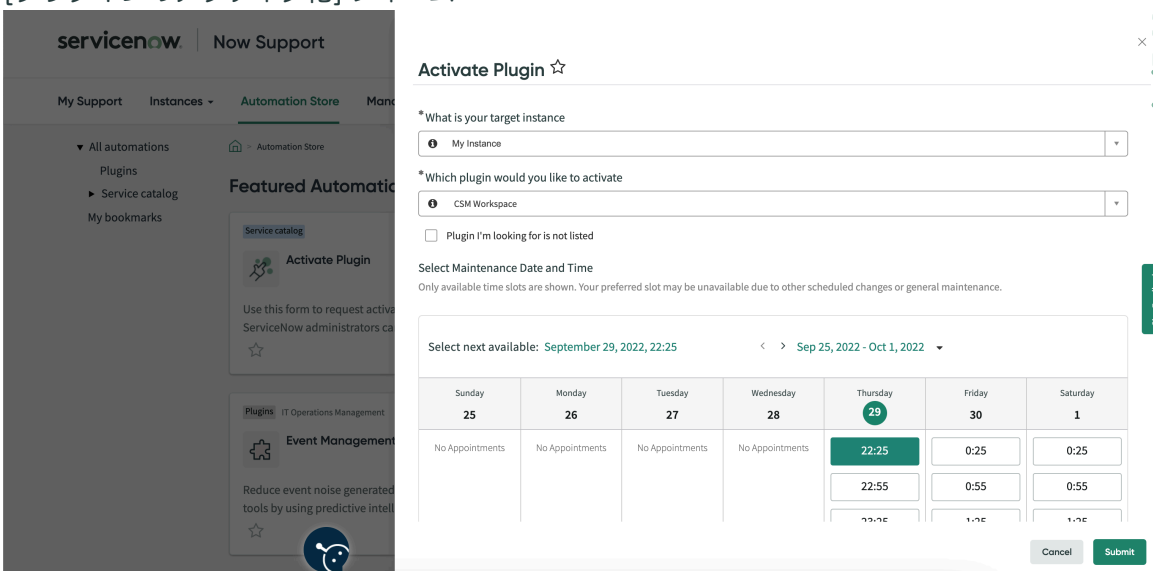
フィールド	説明
ターゲットインスタンスは何ですか	プラグインをアクティブ化するインスタンス。
どのプラグインをアクティブ化しますか	アクティブ化するプラグインの名前です。

フィールド	説明
	注: 必要なプラグインが表示されない場合、または OEM またはオンプレミスのインスタンスでプラグインをアクティブ化している場合は、[探しているプラグインが表示されていません **(Plugin I'm looking for is not listed)]** チェックボックスをオンにして、プラグインの名前を入力します。
メンテナンスの日時を選択 (Select Maintenance Date and Time)	プラグインをアクティブ化する日時。 注: プラグインは、米国太平洋標準時で、毎営業日の朝と夕方の 2 回のバッチでアクティブ化されます。特定の時刻にプラグインをアクティブ化する必要がある場合は、[理由/コメント **(Reason/ Comments)]** フィールドに要求を入力します。

Example

たとえば、 [自分のインスタンス (My Instance)] という名前のインスタンスで CSM Workspace プラグインをアクティブ化するには、次のフォームを参照してください。

[プラグインのアクティブ化] フォーム



7. **[Submit** (送信)] を選択します。
   プラグインの要求の詳細については、次を参照してください。 のサービスカタログ [KB0751715] 記事からのプラグインの要求 Now Support ナレッジベース.

外部認証情報ストレージの設定

リモート リポジトリーから認証情報を取得するようにインスタンスを設定します。

次の手順では、保護する認証情報で外部リポジトリーを設定していることを前提としています。ServiceNow インスタンスで設定されている認証情報識別子は、JAR ファイルを使用してリポジトリー内の実際の認証情報にマップする必要があります。

> **注:** ServiceNow では、同時に 2 つの外部ボールトがサポートされています。1 つはデフォルトの CyberArk 認証情報リゾルバー、もう 1 つはカスタム外部認証情報リゾルバーです。カスタム CyberArk 認証情報リゾルバーを作成すると、引き続き 2 つ目のカスタム外部ボールトが使用され、追加のカスタム外部ボールトを使用することはできません。

外部認証情報ストレージを設定するには、次のタスクを順に実行します。

認証情報を解決するための **JAR** ファイルの作成

MID サーバーから送信された認証情報の識別子をリポジトリーの実際の認証情報に解決するために、JAR ファイルを作成します。

始める前に

必要なロール：agent_admin または admin

インスタンスが予期するすべての認証情報要素 (秘密鍵など) を含めるようにしてください。

認証情報を解決するための JAR ファイルを作成するには、次の手順に従います。

手順

ServiceNow github で提供されているテンプレートまたはサンプル Java ファイルのいずれかを使用します。

> **警告:** これらのサンプルはテンプレートとしてのみ使用してください。お使いの環境に合わせて変更しないまま本番環境でこのコードを使用しないでください。

a. ServiceNow github からの指示に従って、オープンソースの JAR ファイルをダウンロードします。

- HashiCorp 外部認証情報リゾルバー
- CyberArk 外部認証情報リゾルバー

b. 次のサンプル Java ファイルをテンプレートとして使用し、環境に合わせて変更します。

Example

```
package com.snc.discovery;

import java.util.*;
import java.io.*;

/**
 * Basic implementation of a CredentialResolver that uses a properties file.
 */

public class CredentialResolver {

 private static String ENV_VAR = "CREDENTIAL_RESOLVER_FILE";
 private static String DEFAULT_PROP_FILE_PATH = "C:\\dummycredentials.properties";

 // These are the permissible names of arguments passed INTO the resolve()
```

```
// method.

// the string identifier as configured on the ServiceNow instance...
public static final String ARG_ID = "id";

// a dotted-form string IPv4 address (like "10.22.231.12") of the target
// system...
public static final String ARG_IP = "ip";

// the string type (ssh, snmp, etc.) of credential as configured on the
// instance...
public static final String ARG_TYPE = "type";

// the string MID server making the request, as configured on the
// instance...
public static final String ARG_MID = "mid";

// These are the permissible names of values returned FROM the resolve()
// method.

// the string user name for the credential, if needed...
public static final String VAL_USER = "user";

// the string password for the credential, if needed...
public static final String VAL_PSWD = "pswd";

// the string pass phrase for the credential if needed:
public static final String VAL_PASSPHRASE = "passphrase";

// the string private key for the credential, if needed...
public static final String VAL_PKEY = "pkey";

// the string authentication protocol for the credential, if needed...
public static final String VAL_AUTHPROTO = "authprotocol";

// the string authentication key for the credential, if needed...
public static final String VAL_AUTHKEY = "authkey";

// the string privacy protocol for the credential, if needed...
public static final String VAL_PRIVPROTO = "privprotocol";

// the string privacy key for the credential, if needed...
public static final String VAL_PRIVKEY = "privkey";


private Properties fProps;

public CredentialResolver() {
}

private void loadProps() {
 if(fProps == null)
  fProps = new Properties();

 try {
  String propFilePath = System.getenv(ENV_VAR);
  if(propFilePath == null) {
```

自動翻訳

```
   System.err.println("Environment var "+ENV_VAR+" not found. Using default file:
"+DEFAULT_PROP_FILE_PATH);
   propFilePath = DEFAULT_PROP_FILE_PATH;
  }

  File propFile = new File(propFilePath);
  if(!propFile.exists() || !propFile.canRead()) {
   System.err.println("Can't open "+propFile.getAbsolutePath());
  }
  else {
   InputStream propsIn = new FileInputStream(propFile);
   fProps.load(propsIn);
  }

 //fProps.load(CredentialResolver.class.getClassLoader().getResourceAsStream("dummycred
entials.properties"));
  } catch (IOException e) {
   System.err.println("Problem loading credentials file:");
   e.printStackTrace();
  }
 }

 /**
  * Resolve a credential.
  */
 public Map resolve(Map args) {
  loadProps();
  String id = (String) args.get(ARG_ID);
  String type = (String) args.get(ARG_TYPE);
  String keyPrefix = id+"."+type+".";

  if(id.equalsIgnoreCase("misbehave"))
   throw new RuntimeException("I've been a baaaaaaaaad CredentialResolver!");

  // the resolved credential is returned in a HashMap...
  Map result = new HashMap();
  result.put(VAL_USER, fProps.get(keyPrefix + VAL_USER));
  result.put(VAL_PSWD, fProps.get(keyPrefix + VAL_PSWD));
  result.put(VAL_PKEY, fProps.get(keyPrefix + VAL_PKEY));
  result.put(VAL_PASSPHRASE, fProps.get(keyPrefix + VAL_PASSPHRASE));
  result.put(VAL_AUTHPROTO, fProps.get(keyPrefix + VAL_AUTHPROTO));
  result.put(VAL_AUTHKEY, fProps.get(keyPrefix + VAL_AUTHKEY));
  result.put(VAL_PRIVPROTO, fProps.get(keyPrefix + VAL_PRIVPROTO));
  result.put(VAL_PRIVKEY, fProps.get(keyPrefix + VAL_PRIVKEY));

  System.err.println("Error while resolving credential id/type["+id+"/"+type+"]");

  return result;
 }


 /**
  * Return the API version supported by this class.
  */
 public String getVersion() {
  return "1.0";
 }
```

自動翻訳

```
public static void main(String[] args) {
 CredentialResolver obj = new CredentialResolver();
 obj.loadProps();

 System.err.println("I spy the following credentials: ");
 for(Object key: obj.fProps.keySet()) {
  System.err.println(key+": "+obj.fProps.get(key));
 }

 }
}
```

認証情報を解決するための **JAR** ファイルのインポート

MID サーバーから送信された認証情報識別子をリポジトリーの実際の認証情報に解決するために作成した JAR ファイルをインポートします。

始める前に

必要なロール：agent_admin または admin

JAR ファイルの作成後、そのファイルをインスタンスにインポートします。ここで、MID サーバーにアクセス可能になります。

手順

1. JAR ファイルとプロパティ ファイルの作成後、プロパティ ファイルを MID サーバーにコピーします。
2. 移動先 **MID** サーバー > **JAR** ファイル.
3. [**New**] をクリックします。
4. 次のフィールドに入力します。

フィールド	説明
名前	インスタンス内のファイルを識別するためのわかりやすい一意の名前。
バージョン	ファイルのバージョン番号 (使用可能な場合)。
ソース	参照用の JAR ファイルの場所。ソース情報はシステムで使用されません。
説明	インスタンス内の JAR ファイルとその目的の簡単な説明。

5. バナーにあるクリップ アイコンをクリックして JAR ファイルをレコードに添付します。

Example

JAR ファイルの添付

6. [送信] をクリックします。
7. MID サーバーサービスを再起動します。
プラットフォームにより、インスタンスと通信するように設定された任意の MID サーバーで JAR ファイルが使用可能になります。

認証情報識別子の設定

インスタンスで認証情報識別子を設定します。

始める前に

必要なロール：admin
次の項目を確認します。

- 外部認証情報ストレージ プラグインが有効になっていること。
- 外部認証情報ストレージディスカバリープロパティが有効になっていること。

手順

1. 移動先 すべて > ディスカバリー > 認証情報 または オーケストレーション > 認証情報.
2. [**New**] をクリックします。
3. 認証情報タイプを選択します。
4. **[External credential store]** (外部の認証情報ストア) チェック ボックスをオンにします。
   [ユーザー名] フィールドおよび [パスワード] フィールドは消去され、[認証情報 **ID]** フィールドおよび [認証情報ストレージ **Vault]** メニューが表示されます。
5. [認証情報ストレージ **Vault]** メニューから、[なし]、[CyberArk vault]、またはカスタム外部認証情報ストレージ vault のいずれかを選択します。

   注:

   [CyberArk vault] が選択されている場合は、[ルックアップキー] メニューに、[認証情報 ID]、[IP アドレス]、[FQDN]、[上記のすべて] の 4 つのルックアップキーの選択肢が表示されます。[上記のすべて] を選択すると、 vault に複数回アクセスする必要があるため、パフォーマンスが低下する可能性があります。

   a. カスタム外部認証情報ストレージ vault を使用するには、インスタンスの Vault 構成 [vault_configuration.list] に移動します。

   b. カスタム認証情報リゾルバーのインポートされた JAR ファイルに関連付けられた名前を使用して、新しいレコードを作成します。

   カスタム外部認証情報ストレージ vault の作成については、手順「認証情報を解決するための JAR ファイルの作成」および「認証情報を解決するための JAR ファイルのインポート」を参照してください。

6. 以下のテーブルのフィールドを使用して、認証情報フォームに入力します。

フィールド	説明
名前	この認証情報のわかりやすい一意の名前を入力します。
アクティブ	これらの認証情報の使用を有効または無効にします。
認証情報 ID	外部認証情報システム用に MID サーバーにアップロードされた JAR ファイルに外部認証情報用に設定された一意のキーを入力します。これは、パラメーター マップで Java クラスに渡される ID です。 `public static final String ARG_ID = "id";` MID サーバーは、この識別子を使用して、リポジトリー上の実際の認証情報を解決します。

フィールド	説明
	注: このフィールドは、[外部の認証情報ストア] チェックボックスがオンになっている場合にのみ表示されます。
タグ	ワークフロー作成者は、個々の認証情報をオーケストレーションワークフロー内の任意のアクティビティに割り当てたり、オーケストレーションワークフロー内で同じアクティビティタイプが発生するたびに異なる認証情報を割り当てたりすることができます。
外部の認証情報ストア	外部認証情報ストレージシステムを使用するには、このチェックボックスをオンにします。このオプションを選択すると、[ユーザー名] フィールドと [パスワード] フィールドが [認証情報 **ID]** フィールドに置き換えられます。外部の認証情報ストレージは、 外部認証情報ストレージ プラグインをアクティブ化した場合にのみ使用できます。
認証情報ストレージ Vault	利用可能な Vault のリストから外部認証情報ストレージ Vault を選択します。メニューは、Vault 構成 [vault_configuration.list] のレコードで構成されています。新しいレコードを追加し、カスタム認証情報リゾルバー JAR ファイルに関連付けられた名前を使用できます。カスタム外部認証情報ストレージ vault の作成については、手順「認証情報を解決するための JAR ファイルの作成」および「認証情報を解決するための JAR ファイルのインポート」を参照してください。
適用先	これらの認証情報をネットワーク内の [すべての **MID** サーバー] または 1 つ以上の [特定の **MID** サーバー] に適用するかどうかを選択します。**[MID** サーバー] フィールドでこれらの認証情報を使用する MID サーバーを指定します。
MID サーバーを使用	使用可能な MID サーバーのリストから 1 つ以上の MID サーバーを選択します。このレコードで設定された認証情報は、このリストの MID サーバーで使用できます。このフィールドは、[適用先] フィールドで [特定の **MID** サーバー] を選択した場合のみ使用できます。
順番	プラットフォームがデバイスにログオンしようとするときに、この認証情報を試行する順序 (シーケンス) を入力します。番号が小さいほど、この認証情報がリストの上位に表示されます。多くの認証情報を使用する場合、またはログイン試行が 3 回失敗してセキュリティによってユーザーがロックアウトされた場合、認証情報の順序を確立します。すべての認証情報に同じ順序番号を設定した場合、または何も設定しない場合、ディスカバリーまたはオーケストレーションでは、認証情報をランダムな順序で試行します。

7. [送信] をクリックします。

AWS の認証情報識別子の設定

リモート リポジトリーから認証情報を取得するようにインスタンスを設定します。

始める前に
必要なロール：cloud_admin

次のプラグインがアクティブであり、MID サーバーがインストールされていることを確認します。

- ディスカバリー [com.snc.discovery]
- クラウドプロビジョニングとガバナンス [com.snc.cloud.mgmt]
- 外部認証情報ストレージ [com.snc.discovery.external_credentials]

自動翻訳

このタスクについて
次の手順では、保護する認証情報で外部リポジトリーを設定していることを前提としています。ServiceNow インスタンスで設定されている認証情報識別子は、JAR ファイルを使用してリポジトリー内の実際の認証情報にマップする必要があります。

手順

1. 移動先 すべて > ディスカバリー > 認証情報.
2. 外部認証情報ストレージプロバイダーがサポートする認証情報を選択します。
3. テーブル内のフィールドを使用して、フォームに入力します。

フィールド	説明
名前	この認証情報にわかりやすい一意の名前。たとえば、Amazon Web サービスです。
アクティブ	認証情報を有効または無効にするためのチェック ボックス。
認証情報 ID	この認証情報が外部認証情報ストレージプロバイダーに保存される名前を入力します。
MID サーバー	これらの認証情報を使用できる MID サーバーを 1 つ以上選択します。
外部の認証情報ストア	外部認証情報ストレージシステムを使用するには、このチェックボックスをオンにします。外部ストレージが有効になっている場合は、[認証情報 **ID**] フィールドが表示されます。このチェックボックスが表示されていない場合は、ヘッダーバーのメニューアイコンをクリックして、 表示 > 外部ストレージ] を選択します。
認証情報ストレージ Vault	**[CyberArk]** を選択します。

4. [**Submit** (送信)] を選択します。

CyberArk 認証情報ストレージの統合

MID サーバー と CyberArk ボールトの統合により、インスタンスに認証情報を保存せずに、ServiceNow オーケストレーション、ServiceNow ディスカバリー および ServiceNow サービスマッピング を実行することができます。

CyberArk の概要

CyberArk の Application Identity Management (AIM) 製品では、Privileged Account Security ソリューションを使用して、アプリケーション、スクリプト、または構成ファイルに組み込まれたアプリケーション パスワードを保存する必要性をなくします。さらに、この製品により、CyberArk ボールト内でこれらの非常に機密性の高いパスワードの保存、ログ記録、および管理を一元的に行うことができます。このアプローチにより、組織は定期的なパスワード交換の社内要件および規制要件を遵守し、オンプレミスかクラウドかに関係なく、あらゆる種類の特権 ID に関連付けられたアクティビティを監視することができます。

インスタンスは、各認証情報の一意の識別子、認証情報タイプ (SSH、SNMP、Windows など)、および任意の 認証情報親和性を保持します。MID サーバー は、インスタンスから認証情報識別子、認証情報タイプ、および IP アドレスを取得し、CyberArk ボールトを使用してこれらの要素を使用可能な認証情報に解決します。認証情報リゾルバーは、ホスト名「fqdn」を検索し、リバース DNS ルックアップを使用して「fqdn」を取得することもできます。

CyberArk 統合にはServiceNow 外部認証情報ストレージプラグインが必要です。これは、 システム定義 > プラグイン.MID サーバー は、CyberArk AIM/API クライアントと同じマシンにインストールする必要があります。CyberArk Application Access Manager (AAM) 認証情報プロバイダーバージョン 12.0.1 以降がサポートされています。

CyberArk とともにインストールされる内容

- ビジネスルール：外部認証情報ストレージのビジネス ルールは、アドミニストレーターが外部認証情報ストレージ プロパティを変更すると、次のタスクを実行します。
  - 認証情報レコードリストおよびフォームのビューを外部ストレージビューに変更します。このビューを使用すると、ユーザーはリスト内の [認証情報 ID] 列を確認できます。
  - MID サーバー に、認証情報の取得方法の変更に備えて、非外部認証情報キャッシュを更新するように指示します。
- システムプロパティ：外部認証情報ストレージの有効化と呼ばれるプロパティ [com.snc.use_external_credentials] は、外部認証情報ストレージプラグインがアクティブにされた後で、それを有効または無効にします。このプロパティの場所:ディスカバリー定義 > プロパティそしてオーケストレーション > **MID** サーバープロパティで、プラグインをアクティブ化すると有効になります。

> **注:** システム プロパティを使用して外部認証情報ストレージを無効にすると、すべての外部認証情報がインスタンス内で非アクティブに自動的に設定されます。このプロパティを使用して機能を再度有効にしても、外部認証情報レコードはアクティブにリセットされません。それぞれの認証情報レコードを手動で再アクティブ化する必要があります。

サポートされている認証情報タイプ

CyberArk の統合では、次の ServiceNow 認証情報タイプがサポートされています。

- GCP
- Azure
- CIM
- JMS
- SNMP フォーラム
- SNMPv3
- 基本認証
- SSH キーペア
- SSH 秘密鍵 (キー、パスフレーズ、およびパスワードを含む)
- VMware
- Windows
- 適用可能な認証情報

> **注:** CyberArk 統合を GCP 認証情報タイプと使用するには、外部認証情報ストレージ jar を変更する必要があります。詳細については、「CyberArk を使用した ServiceNow GCP 認証情報リゾルバー」を参照してください。

また、次のネットワークプロトコルを使用する ServiceNow AI Platform 機能では、CyberArk ボールトに保存された認証情報の使用がサポートされています。

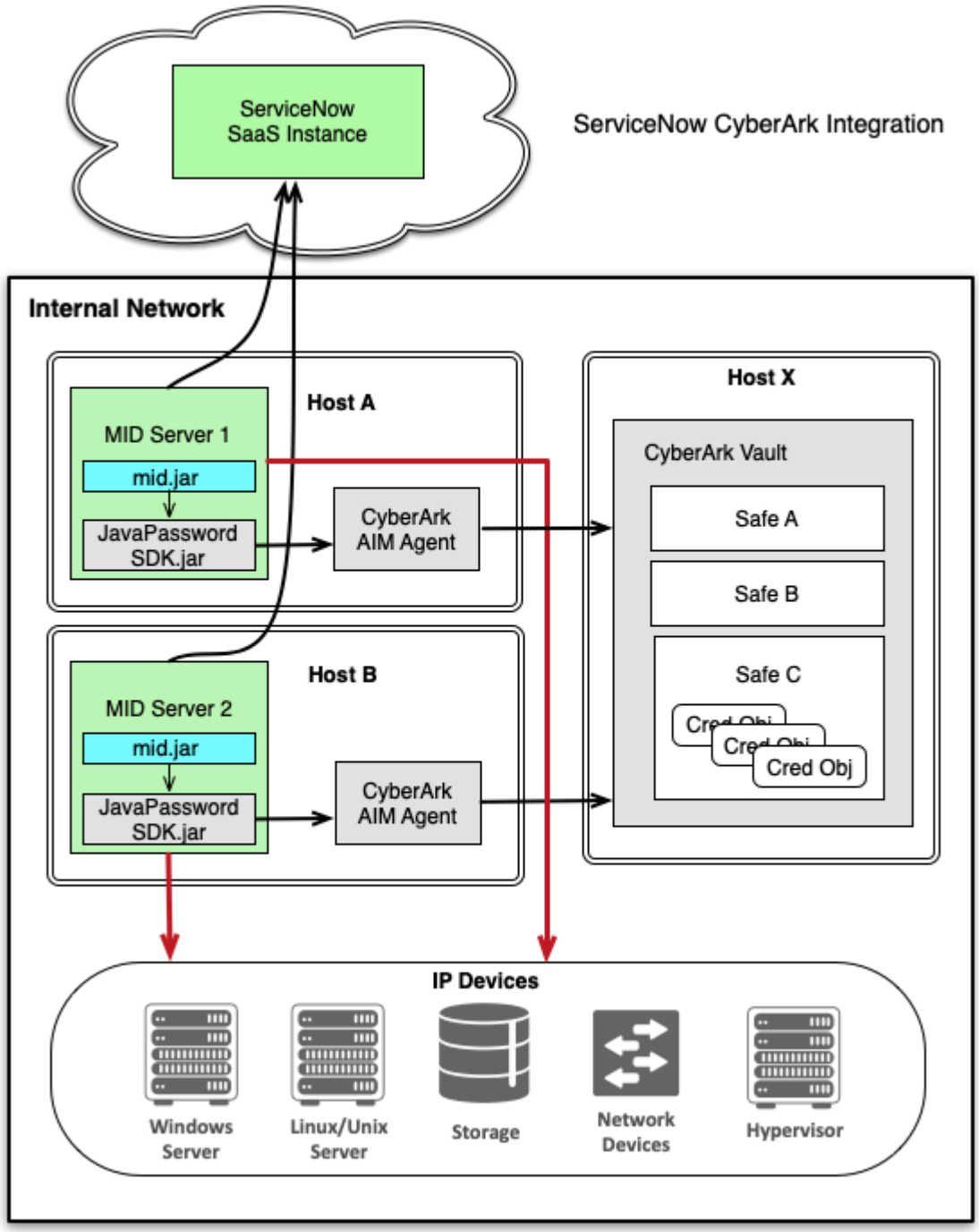
ネットワークプロトコルでサポートされている認証情報

ネットワークプロトコル	ServiceNow フローデザイナー でのサポート	オーケストレーション でのサポート
SOAP	SOAP ステップ	ベーシック認証を上書きして、SOAP Web サービスアクティビティを作成 します。
REST	REST ステップ	ベーシック認証を上書きして、REST Web サービスアクティビティを作成 します。
JDBC	JDBC ステップ	JDBC アクティビティ
SSH	SSH ステップ	SSH アクティビティ
PowerShell	PowerShell ステップ	PowerShell アクティビティ
SFTP	SFTP ステップ	SFTP アクティビティ
JMS		JMS アクティビティ

重要: 同じ MID サーバー を使用して、CyberArk ボールトに保存されている認証情報とカスタムの外部認証情報ストレージを管理することはできません。MID サーバー は、CyberArk AIM/API クライアントと同じマシンにインストールする必要があります。

CyberArk アーキテクチャ

CyberArk のストレージアーキテクチャ



注: CyberArk は、ベースシステムの `mid.jar` ファイルを使用して認証情報を解決します。

MID サーバー による Windows アカウントの処理方法

まず、認証情報ルックアップでは、指定した認証情報 ID と CyberArk ボールトの [名前] フィールドの照合を試行します。一致するものが見つかると、その認証情報が返されます。一致するものが見つからない場合は、認証情報ルックアップで IP アドレスを使用して一致の検索が試行されま

す。IP アドレス ルックアップが同じサーバー上の Windows や Tomcat など複数の認証情報と一致する場合、そのルックアップは失敗します。この問題を回避するには、MID サーバー の config.xml ファイルにある *ext.cred.type_specifier* パラメーターを **[true]** に設定し、CyberArk が認証情報タイプと IP アドレスの両方に一致する認証情報を返すようにします。たとえば、IP アドレスが Windows と Tomcat の両方で共有されている場合、Windows の認証情報タイプにより、Windows 認証情報のみが返されます。

CyberArk ライブラリのアップグレード

保護された構成パラメーターが必要な場合は、CyberArk ライブラリをアップグレードできます。

config.xml で次の構成パラメーターを確認します。`<parameter name="mid.secure_config.provider" value="com.service_now.mid.services.config.CyberArkSecuredConfigProvider"/>`

保護された構成パラメータープロバイダーが構成されている場合は、次の手順を実行してアップグレードを実行します。

1. CyberArk クライアントバージョンの名前を「JavaPasswordSDK_MajorVersion_minorVersion_patchNum.jar」に変更します。
2. 名前変更 jar を添付できる ecc_agent テーブルに新しい jar エントリを作成します。この新しいエントリが MID サーバーにダウンロードされます。このステップにより、2 つの jar (Passworsdk.jar と JavaPasswordSDK _12_X_X.jar) が生成されます。
3. インスタンスから古い ecc_agent エントリを削除します。このステップにより、MID サーバーから Passworsdk.jar が削除され、JavaPasswordSDK _12_X_X.jar はシステムに残ります。

CyberArk の統合設定

次の手順には、適切な CyberArk ドキュメントへの参照など、CyberArk と ServiceNow の両方の設定タスクが含まれています。

ServiceNow インスタンスで設定されている認証情報識別子は、CyberArk ボールト内の認証情報名にマップする必要があります。認証情報を参照するときに、MID サーバーは認証情報識別子をボールト内の名前と照合して、認証情報を検索します。名前は一意でなければなりません。認証情報識別子が空の場合、MID サーバーは IP アドレスで認証情報を検索します。IP アドレスで認証情報を特定するために、認証情報タイプが調べられ、そのアドレスでそのタイプの認証情報が 1 つだけ存在することが確認されます。たとえば、Windows サーバーと vCenter が両方とも同じ IP アドレスで実行されている場合です。このような厳密な認証情報要件を SSH 環境でサポートするために、MID サーバーの設定パラメーターを使用すると、要求された認証情報タイプが CyberArk から返されるタイプと一致することを要求できます。

> **注:** [認証情報 **ID]** フィールドは、SNMPv3 を除くすべてのケースで認証情報を CyberArk にマッピングするために必要な唯一のフィールドです。[プライバシー認証情報 **ID]** フィールドはオプションであり、SNMPv3 認証情報を使用しており、認証情報にプライバシープロトコルを使用している場合にのみ必要です。詳細は、CyberArk 認証情報識別子の設定を参照してください。

CyberArk ボールトから認証情報を取得するようにインスタンスを設定するには、下記に示す順序で次のタスクを実行します。

CyberArk ボールトの設定と AIM API のインストール

MID サーバーにアクセスできるように CyberArk ボールトを設定し、MID サーバーマシンに CyberArk AIM API をインストールします。

始める前に
必要なロール：admin

この手順を開始する前に、外部認証情報ストレージプラグインがアクティブ化されていることを確認してください。CyberArk Application Access Manager (AAM) 認証情報プロバイダーバージョン 12.0.1 以降がサポートされています。

手順

1. 認証情報を要求するすべての MID サーバーで使用されるアプリケーション ID および認証詳細を使用して、CyberArk ボールトを構成します。
詳細については、『CyberArk Credential Provider and ASCP Implementation Guide (CyberArk Credential Provider および ASCP 実装ガイド)』を参照してください。

   a. CyberArk で *ServiceNow_MID_Server* と呼ばれるアプリ ID を作成して、MID サーバーがボールトにアクセスできるように CyberArk が設定されていることを確認します。

   b. *ServiceNow_MID_Server* アプリ ID へのアクセスに MID サーバーに必要なすべての認証情報が与えられることを確認します。

      注: ext.cred.app_id パラメーターを使用して、MID サーバーの config.xml ファイルにあるデフォルトの **ServiceNow_MID_Server** アプリ ID を上書きすることができます。このパラメーターの値を変更する場合は、ボールトで一致する値を設定してください。

2. 認証情報ストアへのアクセスに使用使用する MID サーバーサービスをホストする各マシンに、CyberArk Credential Provider (AIM API を含む) をインストールします。
3. CyberArk アカウントをプロビジョニングし、アプリケーション アクセスの権限を設定します。
詳細については、『CyberArk Privileged Account Security Implementation Guide (CyberArk Privileged Account Security 実装ガイド)』を参照してください。

   a. CyberArk Password Safe で、さまざまなデバイスにアクセスするためにディスカバリー、オーケストレーション、またはサービスマッピングで必要な特権アカウントを作成し、これらのアカウントが、必要な認証情報が保存されているセーフのメンバーであることを確認します。

   b. Credential Provider およびアプリケーションのユーザーを、アプリケーションのパスワードが保存されている Password Safe のメンバーとして追加します。

CyberArk JAR ファイルのインポート

CyberArk JavaPasswordSDK.jar ファイルをインスタンスにインポートして、MID サーバーにアクセスできるようにします。

始める前に
必要なロール：agent_admin または admin
この手順を開始する前に、MID サーバーが認証情報にアクセスできるように CyberArk が設定されていることを確認してください。ボールトへのアクセスに使用する MID サーバーをホストする各サーバーに CyberArk AIM API がインストールされていることを確認してください。

注: 2 つの別々の CyberArk 統合は、MID サーバーではサポートされていません。同じバージョンの CyberArk AIM API を、同じインスタンスに接続されているすべての MID サーバーにインストールする必要があります。

このタスクについて
JavaPasswordSDK.jar ファイルがすでに MID サーバーに存在する場合でも、このプロセスを使用します。

手順

1. 移動先 すべて > **MID** サーバー > **JAR** ファイル.
2. [**New**] をクリックします。

3. テーブル内のフィールド使用して、フォームに値を入力します。

JAR ファイル フォームのフィールド

フィールド	説明
名前	インスタンス内のファイルを識別するためのわかりやすい一意の名前。
バージョン	ファイルのオプション バージョン番号 (使用可能な場合)。
ソース	JAR ファイルのプロバイダー。ソース情報はシステムで使用されません。
説明	インスタンス内の JAR ファイルとその目的の簡単な説明。

4. このレコードに JAR ファイルを添付します。
AIM JavaPasswordSDK.jarファイルは、AIM SDK インストール ファイルに付属しており、通常、MID サーバーの AIM インストールディレクトリー <install_dir>/CyberArk/ApplicationPasswordSdk にあります。

> **注:** 添付されている AIM JavaPasswordSDK.jarは、インストールされている CyberArk AIM API のバージョンと一致する必要があります。JavaPasswordSDK.jarファイルに不一致があると、予期しない動作や潜在的な機能上の問題が発生する可能性があります。

5. [**Submit** (送信)] を選択します。
6. MID サーバーサービスを再起動します。
プラットフォームにより、インスタンスと通信するように設定された任意の MID サーバーで JAR ファイルが使用可能になります。

CyberArk 用の **MID** サーバーの設定

MID サーバーに CyberArk ボールトへのアクセスを付与するように config.xml を設定します。

始める前に
必要なロール：admin
この手順を開始する前に、JavaPasswordSDK.jar ファイルをインスタンスにインポートします。

手順

次のパラメーターを使用して MID サーバーの MID サーバーのパラメーターを追加 ファイルを手動で設定します。

この設定は、インスタンスから実行することはできません。

必要な設定パラメーター

パラメーター	値	説明
ext.cred.safe_folder	NameOfFolder	すべての認証情報ルックアップに使用するフォルダー。たとえば、ルートです。
ext.cred.use_cyberark	true	この MID サーバーが CyberArk と統合されることを示すブーリアン パラメーター。

オプションの設定パラメーター

パラメーター	値	説明
ext.cred.safe_timeout	5 (秒)	ボールト内の各認証情報ルックアップのタイムアウト (秒単位)。
ext.cred.safe_name	NameOfSafe	すべての認証情報ルックアップに使用されるデフォルトの安全な名前。パラメーターが複数のセーフにある場合、認証情報 ID は *<safeName>:<CredentialID>* の形式で指定することができます。このように設定した場合、**[NameOfSafe]** フィールドは無視されます。すべての外部認証情報にこの形式で指定された認証情報 ID がある場合は、**[NameOfSafe]** フィールドをスキップします。 **注:** デフォルトでは、この形式の区切り文字はコロンです。区切り文字として任意の文字を割り当てるには、safe.cred.split.string=<string> の行を CredMap.properties ファイルに追加します。
ext.cred.app_id	ServiceNow_MID_Server	CyberArk ボールトにアクセスするために MID サーバーに許可を与えるために使用されるアプリ ID を指定します。CyberArk ボールトでデフォルト値 **ServiceNow_MID_Server** を定義する必要があります。このパラメーターを使用して、デフォルトを上書きして独自のアプリ ID を指定することができます。このパラメーターでアプリ ID を編集する場合は、CyberArk が一致するようにこれを設定してください。
ext.cred.type_specific	true	CyberArk プラットフォーム ID と IP アドレスの両方に一致する認証情報を返すよう IP アドレスルックアップに強制します。たとえば、IP アドレスが Windows と Tomcat の両方で共有されている場合、**Win** で始まるプラットフォーム ID を含む認証情報により、Windows 資格情報のみが返されます。このパラメーターを true に設定すると、CyberArk では、次で始まるプラットフォーム ID が検索されます。 • Win：Windows • Unix：SSH • VMWare：VMware
ext.cred.check_ssh_type	false	true に設定した場合、CyberArk から返される SSH 認証情報のタイプが、要求された認証情報のタイプと一致する必要があります。たとえば、通常の SSH ユーザー名/パスワード認証情報が要求され、SSH キーのみが使用可

パラメーター	値	説明
		能な場合、認証情報ルックアップは失敗します。

SNMPv2 認証情報用の CyberArk の設定

システムで SNMPv2 が使用される場合は、認証情報の属性をコミュニティ文字列にマップする特別なファイルを作成することができます。

始める前に

必要なロール：admin
この手順を開始する前に、CyberArk ボールトにアクセスできるように MID サーバーを設定してください。

このタスクについて

注: コミュニティ文字列が CyberArk 認証情報のパスワード フィールドに表示される場合は、この手順を実行する必要はありません。

SNMPv2 は、CyberArk ではネイティブにサポートされていません。組織が、コミュニティ文字列が認証情報のパスワード フィールドに表示されないカスタムの SNMPv2 認証情報を作成した場合は、この手順を使用して、認証情報の属性をコミュニティ文字列にマップします。

手順

1. テキスト エディターで、次のコードを含む、CredMap.properties と呼ばれるファイルを作成します。
   `SNMPv2.Community=attribute_name`
2. このファイルを MID サーバーインストールの /agent ディレクトリーに保存します。
   認証情報ルックアップで、MID サーバーは認証情報のこの属性の検索を試行します。属性が見つからない場合、MID サーバーはパスワード フィールドを確認します。パスワード フィールドが空の場合、認証情報ルックアップは失敗します。

CyberArk 認証情報識別子の設定

外部リポジトリー内の特定の認証情報を識別するために CyberArk で使用できる一意のキーを作成します。

始める前に

必要なロール：admin
この手順を開始する前に、外部認証情報ストレージ プラグインが有効化され、com.snc.use_external_credentials システム プロパティが true に設定されていることを確認してください。

手順

1. 移動先 すべて > ディスカバリー > 認証情報 または オーケストレーション > 認証情報.
2. [**New**] をクリックします。
3. 認証情報タイプのリストから、 CyberArk の外部ストレージをサポートするタイプを選択します。
4. 認証情報タイプのフィールドを使用してフォームに入力します。
5. **[External credential store]** (外部の認証情報ストア) チェック ボックスをオンにします。
   [ユーザー名] フィールドと [パスワード] フィールドが [認証情報 **ID]** フィールドに置き換えられます。

注: チェックボックスが表示されていない場合は、ヘッダーバーのメニューアイコンをクリックして、 表示 > 外部ストレージ.

6. 次のいずれかの形式を使用して、[認証情報 **ID]** フィールドに式を入力します。
   - すべての認証情報が同じセーフにある場合は、ext.cred.safe_name パラメーターを使用して MID サーバーの config.xml ファイルにこの安全な名前を設定してから、認証情報 ID を **<credential ID>** として名前のみで指定します。
   - 特定のセーフにある指定したプラットフォームの認証情報に名前を付けるには、認証情報 ID を **<safe>:<credential ID>:<platform ID>** として定義します。
   - 認証情報が複数のセーフにある場合は、認証情報 ID を **<safe>:<credential ID>** の形式で指定します。
   - CyberArk が別のセーフを使用して IP アドレスで認証情報を検索できるようにする場合は、認証情報 ID を **<safe>:** の形式で指定します。
   - CyberArk が同じセーフで別のプラットフォーム ID の認証情報を検索できるようにする場合は、**::<platform ID>** の形式を使用します。
   - CyberArk が、設定されたセーフで認証情報を認証情報 ID ではなく IP アドレスで検索できるようにする場合は、このフィールドを空白のままにします。これは、各サーバーに一意の認証情報があるインストールを処理するためのベスト プラクティスです。このタイプのルックアップがない場合は、環境内のすべてのサーバーのインスタンスで認証情報 ID レコードを作成する必要があります。

   注: 認証情報 ID は、CyberArk アカウントの [名前] フィールドの値と一致する必要があります。[認証情報 **ID]** フィールドには 180 文字の制限があります。

7. SNMPv3 認証情報を CyberArk に保存していて、プライバシープロトコルおよびプライバシーキーを使用している場合は、次のように ID を設定します。

   a. [認証情報ストアタイプ] フィールドで **[CyberArk]** を選択します。
   [プライバシー認証情報 **ID]** フィールドが表示されます。

   b. [プライバシー認証情報 **ID]** フィールドに CyberArk SNMPv3 プライバシーアカウントの [名前] を入力します。

8. [送信] をクリックします。

CyberArk ボールトでの AWS 認証情報の設定

インスタンスで使用するために取得する AWS 認証情報を使用して CyberArk ボールトを設定します。

始める前に
必要なロール：admin

このタスクについて
認証情報をアカウントとして CyberArk ボールトに保存します。インスタンスで Vault へのアクセスを設定するときは、アカウントに付けた名前も認証情報 ID として使用する必要があります。

注: 次の手順は、CyberArk Password Vault v14.2.1 を参照しています。別のバージョンを使用している場合は、CyberArk Password Vault の公式ドキュメントに従って設定してください。

手順

1. CyberArk で、 アカウント > アカウントと要求 > アカウントビュー > アカウントの追加.
2. システムタイプ: クラウドサービスを選択します。
3. アサイン先プラットフォーム: **Amazon Web Services - AWS -** アクセスキー。
4. セーフに保存: リストからセーフを選択します。
5. プロパティの定義:次の情報を入力します。

CyberArk 認証情報

フィールド	値
AWS IAM ユーザー名	AWS から提供された AWS アクセス キーを入力します。
AWS アクセスキーシークレット (オプション)	AWS から提供された AWS 秘密アクセス キーを入力します。
アカウント名をカスタマイズ	このキーのカスタム名を入力するには、スライダーを切り替えます。
AWS Access Key ID (AWS アクセス キー ID)	AWS から提供された AWS アクセスキーをもう一度入力します。
AWS アカウント ID 番号	AWS から提供された AWS アクセスキーをもう一度入力します。
AWS アカウントエイリアス名 (オプション)	アカウントのエイリアス名を入力します。

6. [追加] を選択します。

次のタスク
まだ作成していない場合は、インスタンスで認証情報識別子を作成して CyberArk ボールトへのアクセスを設定します。詳細については、「AWS の外部認証情報ストレージへのアクセスの設定」を参照してください。

クラシック **UI** を使用して **CyberArk** ボールトで **AWS** 認証情報を設定する

インスタンスで使用するために取得する AWS 認証情報を使用して CyberArk ボールトを設定します。

始める前に
必要なロール：admin

このタスクについて
CyberArk Password Vault v14.2.1 には、認証情報をアカウントとして保存するための [アカウントビュー] と [アカウントビュー (クラシック UI)] の両方の設定オプションが含まれています。インスタンスで Vault へのアクセスを設定するときは、アカウントに付けた名前も認証情報 ID として使用する必要があります。

注: 次の手順は、CyberArk Password Vault v14.2.1 を参照しています。別のバージョンを使用している場合は、CyberArk Password Vault の公式ドキュメントに従って設定してください。

手順

1. CyberArk で、 アカウント > アカウントと要求 > アカウントビュー (クラシック **UI)** > アカウントの追加.
2. 次の情報を入力します。

CyberArk 認証情報

フィールド	値
デバイス タイプ	[クラウド サービス] を選択します。
Platform Name (プラットフォーム名)	**[Amazon Web** サービス] **- [AWS] -** [アクセスキー] を選択します。
AWS IAM ユーザー名	AWS から提供された AWS アクセス キーを入力します。
AWS Access Key ID (AWS アクセス キー ID)	AWS から提供された AWS アクセスキーをもう一度入力します。
AWS アカウント ID 番号	AWS から提供された AWS アクセスキーをもう一度入力します。
AWS アクセスキーシークレット	AWS から提供された AWS 秘密アクセス キーを入力します。
名前	このキーのカスタム名を入力します。

3. [保存] を選択します。

次のタスク
まだ作成していない場合は、インスタンスで認証情報識別子を作成して CyberArk ボールトへのアクセスを設定します。詳細については、「AWS の外部認証情報ストレージへのアクセスの設定」を参照してください。

CyberArk ボールトでの **Azure** 認証情報の設定

インスタンスで使用するために取得する Azure 認証情報を使用して CyberArk ボールトを設定します。

始める前に
必要なロール：admin

このタスクについて
Azure 認証情報を保存するには、最初に CyberArk ボールトに Azure 認証情報テンプレートを作成します。このプロセスは、ボールトに対して 1 回だけ完了する必要があります。

手順

1. 管理モードで CyberArk にログインします。
2. [管理] タブに移動します。
3. [システム構成] で **[Platform Management]** を編集します。
4. [クラウドプロバイダーテンプレート **(Cloud Provider Template)]** に移動して複製します。
5. Azure 認証情報のテンプレートを編集します。
6. 次の 2 つのプロパティを追加します。

- *Username* として *Name*、および *Client ID* として *Display Name*
- *Address* として *Name*、および *Tenant ID* として *DisplayName*

7. 変更を適用します。
8. [アカウント] セクションに移動し、[アカウントの追加] を選択します。
9. [安全] を選択します。
10. [デバイスタイプ] を [クラウドサービス] に設定します。
11. 以前に編集した Azure テンプレートを選択します。
12. [クライアント **ID**]、[テナント **ID**]、および [パスワード] フィールドの情報を入力します。
13. [保存] を選択します。

外部認証情報ストレージを使用した **MID** サーバー経由の **OAuth 2.0** 認証

OAuth 2.0 認証情報 (クライアント ID とクライアントシークレット) を、ServiceNow インスタンスではなく、CyberArk ボールトに保存します。MID サーバーは、OAuth トークンを取得するために認証情報が必要なときに、CyberArk ボールトから取得します。トークンは MID サーバーに保存され、有効期限が切れると自動的に更新されます。

CyberArk Application Identity Management (AIM) 製品では、Privileged Account Security ソリューションを使用して、アプリケーション、スクリプト、または構成ファイルに組み込まれたアプリケーション パスワードを保存する必要性をなくします。さらに、この製品により、CyberArk ボールト内でこれらの非常に機密性の高いパスワードの保存、ログ記録、および管理を一元的に行うことができます。OAuth 2.0 認証情報を ServiceNow 認証情報レコードに直接保存するのではなく、CyberArk ボールトに保存するように構成できます。CyberArk の詳細については、「CyberArk 認証情報ストレージの統合」を参照してください。

MID サーバー要求の **OAuth 2.0** 認証のアーキテクチャ

アーキテクチャには ServiceNow インスタンスと、Application Identity Manager (AIM) クライアントおよび MID サーバーが構成される環境の 2 つの部分があります。環境の例としては、クラウドや顧客の環境などがあります。

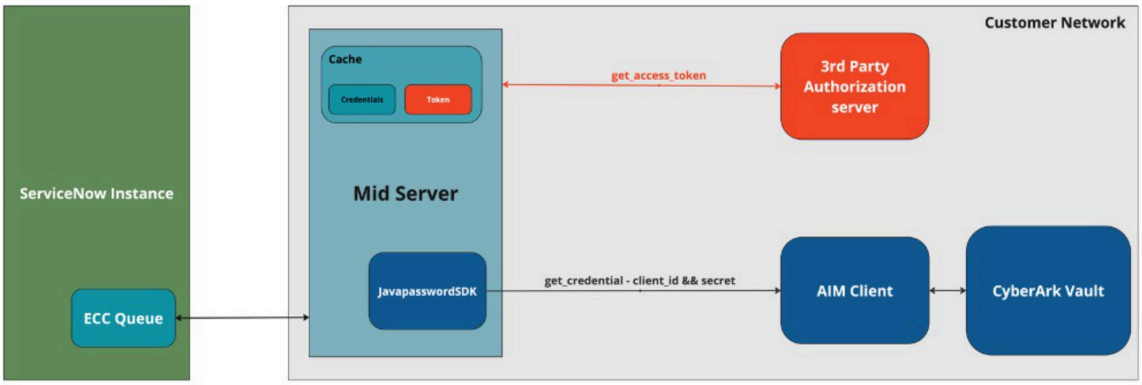
MID サーバーと Application Identity Manager (AIM) クライアントは同じ環境上に構成する必要があり、Application Identity Manager (AIM) は CyberArk 外部ボールトとやり取りするように構成する必要があります。CyberArk 外部ボールトは、MID サーバーおよび Application Identity Manager (AIM) と同じ環境にホストすることも、別の環境にホストすることもできます。

ServiceNow インスタンスは、CyberArk ボールトに保存されている特定の OAuth 2.0 認証情報にマップする認証情報識別子を保持します。MID サーバーは、OAuth トークン要求を送信する前に、インスタンスから認証情報識別子を取得し、顧客提供の JAR ファイルを使用してその識別子を AIM クライアントに送信します。AIM クライアントは CyberArk ボールトに要求を送信します。CyberArk ボールトは、AIM クライアントを通じて OAuth 2.0 認証情報を MID サーバーに送り返します。MID サーバーは、OAuth 2.0 認証情報を受信した後、OAuth トークン要求をサードパーティ認証サーバーに送信します。トークン要求は、CyberArk に保存されるクライアントとクライアントシークレット、OAuth スコープ、インスタンスに保存されるトークン URL などの情報で構成されます。MID サーバーは、認証サーバーが OAuth トークンを発行すると、それをキャッシュメモリーに保存します。

注: この機能は、クライアント認証情報権限許可タイプをサポートしています。

この画像は、MID サーバー要求認証プロセスを示しています。

注: サードパーティ認証サーバーと CyberArk ボールトが顧客ネットワークにホストされていることを前提としています。



JAR ファイルと認証情報識別子の構成

JAR ファイルによって、認証情報識別子が CyberArk 外部ボールトの実際の認証情報に解決されるように、JAR ファイルと認証情報識別子を構成します。このプロセスにより、MID サーバーは OAuth 2.0 認証情報を取得して OAuth トークン要求に含めることができます。

始める前に
必要なロール：agent_admin または admin

CyberArk 外部ボールトは機密認証情報を格納し、ServiceNow インスタンスはボールトに格納される特定の認証情報名にマップされる認証情報識別子を格納します。JAR ファイルを使用すると、認証情報識別子を外部ボールト内の特定の認証情報名にマッピングできます。マッピングにより、MID サーバーは必要な認証情報を取得して OAuth トークン要求に含めることができます。認証情報識別子と JAR ファイルは、それぞれ ServiceNow インスタンスと MID サーバーで構成する必要があります。

手順

1. CyberArk JAR ファイルのインポート。
2. 外部ボールトを使用して MID サーバー経由で OAuth 要求を送信する接続の構成。

CyberArk の構成

OAuth 2.0 認証情報を保存し、MID サーバー からの OAuth 2.0 認証情報の要求に応答するように CyberArk ボールトを構成します。

始める前に
ServiceNow 統合ハブ標準パックインストーラーに登録していることを確認します。詳細については、「https://www.servicenow.com/content/dam/servicenow-assets/public/en-us/doc-type/legal/snc-addendum-integrationhub.pdf 」を参照してください。

必要なロール：agent_admin または admin

手順

1. CyberArk ボールトの設定と AIM API のインストール。
2. CyberArk JAR ファイルのインポート。
3. CyberArk 上の OAuth 2.0 認証情報の構成。
4. 外部ボールトを使用して MID サーバー経由で OAuth 要求を送信する接続の構成。

CyberArk 上の OAuth 2.0 認証情報の構成

ServiceNow インスタンスが要求する OAuth 2.0 認証情報で CyberArk ボールトを構成します。

始める前に
必要なロール：admin

このタスクについて
OAuth 2.0 認証情報を保存するには、最初に CyberArk ボールトに OAuth 2.0 認証情報テンプレートを作成します。このプロセスは、ボールトに対して 1 回だけ実行する必要があります。

手順

1. OAuth 2.0 認証情報を初めて構成する場合は、次の手順を実行します。

   a. 管理モードで CyberArk にログインします。

   b. [管理] タブに移動します。

   c. [プラットフォーム管理 **(Platform Management)]** を選択します。
   [プラットフォーム管理 (Platform Management)] ページの [ターゲット] タブにプラットフォームが表示されます。

   d. プラットフォームタイプを展開します。

   e. プラットフォームテンプレートに対応する設定アイコン ( ) を選択し、[複製] を選択します。

   f. [プラットフォームの複製 (Duplicate Platform)] ウィンドウで、テンプレートの名前を入力し、[作成]を選択します。

      注: 複製テンプレートを作成したシステムタイプを書き留めます。たとえば、クラウドサービスはシステムタイプです。

   g. 作成した複製プラットフォームテンプレートに対応する設定アイコン ( ) を選択し 、[編集] を選択します。

   h. プロパティを追加します。
   ユーザー名としての名前とクライアント ID としての表示名。

   i. 他のプロパティの名前を変更します。

      ヒント: OAuth 2.0 に関連するプロパティ名を選択します。

   j. [適用] を選択します。
   更新が適用されます。

   k. **[OK]** を選択します。

   l. 左側のパネルで、[アカウント] に移動します。

   m. [アカウントの追加] を選択します。

n. [システムのタイプ] 見出しの下で、プラットフォームテンプレートを作成したシステムタイプを選択します。
[システムのタイプ] には、プラットフォームテンプレートが表示されます。

o. [プラットフォームを選択] 見出しの下で、作成したプラットフォームテンプレートを選択します。

p. [SAFe] を選択します。
後で使用するために SAFe 値を保存します。

q. [クライアント ID]、[パスワード] (クライアントシークレットを値として指定)、必要に応じてその他のフィールドに情報を入力します。

r. [追加] を選択します。
アカウントが追加されます。

s. [アカウントビュー (Account View)] ページで、アカウントを開きます。

t. [詳細] タブを選択します。

u. [アカウント名] セクションに保存されている値をコピーし、後で使用するために保存します。
コピーして保存した値は、ServiceNow インスタンスで認証情報識別子として使用されます。

2. OAuth 2.0 認証情報を構成するのが初めてではない場合は、次の手順を実行します。

a. 管理モードで CyberArk にログインします。

b. [アカウント] セクションに移動し、[アカウントの追加] を選択します。

c. OAuth 2.0 を格納するためのテンプレートが存在するシステムタイプを [システムのタイプ **]** から選択します。

d. 以前に作成した、必要なプラットフォーム テンプレートを選択し、**[SAFe]** を選択します。
後で使用するために SAFe 値を保存します。

e. [クライアント ID]、[パスワード] (クライアントシークレットを値として指定)、必要に応じてその他のフィールドに情報を入力します。

f. [追加] を選択します。
アカウントが追加されます。

g. [アカウントビュー (Account View)] ページで、アカウントを開きます。

h. [詳細] タブを選択します。

i. [アカウント名] セクションに保存されている値をコピーし、後で使用するために保存します。
コピーして保存した値は、ServiceNow インスタンスで認証情報識別子として使用されます。

外部ボールトを使用して **MID** サーバー経由で **OAuth** 要求を送信する接続の構成

OAuth 2.0 トークンの要求を MID サーバー経由でサードパーティ認証サーバーに送信するように接続を構成します。MID サーバーは、CyberArk 外部ボールトの OAuth 2.0 認証情報 (クライアント ID とクライアントシークレット)、OAuth スコープ、およびインスタンスのトークン URL を要求に追加して、それらをサードパーティ認証サーバーに送信します。

始める前に

以下の項目を確認します。

- ServiceNow 統合ハブ標準パックインストーラーに登録していることを確認します。詳細については、「https://www.servicenow.com/content/dam/servicenow-assets/public/en-us/doc-type/legal/snc-addendum-integrationhub.pdf 」を参照してください。
- MID サーバーが CyberArk 外部ボールトで認証情報を解決できるようにする JAR ファイルを作成済みです。「認証情報を解決するための JAR ファイルの作成」を参照してください。
- MID サーバーに JAR ファイルをインポート済みです。
- MID サーバーが CyberArk 用に設定されています。「CyberArk 用の MID サーバーの設定」を参照してください。
- OAuth 2.0 が CyberArk の外部ストレージ上で構成されています。「CyberArk 上の OAuth 2.0 認証情報の構成」を参照してください。

必要なロール：なし

このタスクについて

接続と認証情報は、構成テンプレートを使用して構成することも、手動で構成することもできます。構成テンプレートは、接続レコードフォームのコンポーネントを定義し、他の接続レコードに再利用できます。手動の場合は、この目的のためだけに接続を構成できます。このトピックでは、両方のアプローチを示します。

手順

1. 構成テンプレートを使用して接続を構成します。
   a. 移動先 すべて > 統合ハブ > 構成テンプレート.
   b. **[New** (新規)] を選択します。
   c. **[OAuth** クライアント認証情報権限許可タイプを使用した **HTTP** 接続 (外部ストレージ)] を選択します。
   d. 必要に応じて、フォームを更新します。
      たとえば、[デフォルトのデータテンプレート] セクションの oauth_entity_profile_scope フィールドと oauth_entity_scope フィールドにスコープを指定できます。構成テンプレートの作成の詳細については、「構成テンプレートの作成」を参照してください。
   e. [送信] を選択します。
      構成テンプレートが作成されました。
   f. 移動先 すべて > 統合ハブ > 接続と認証情報エイリアス.
   g. フォームを更新します。

自動翻訳

接続および資格情報エイリアス

フィールド	説明
名前	エイリアスの名前。エイリアスには、英数字とアンダースコアだけを使用できます。 アップグレード時に、認証情報レコードのタグが接続および資格情報エイリアスに移行します。認証情報タグに英数字とアンダースコア以外の特殊文字が含まれている場合、タグ名はアップグレード後も保持されます。この移行されたエイリアスは引き続き使用できますが、名前を変更して命名の制限を満たすまでエイリアスを更新することはできません。
アプリケーション	形式 scope_name.alias_name に基づく、接続および資格情報エイリアスの一意の識別子です。 ▪ スコープがグローバルである場合、ID はエイリアス名です。たとえば、グローバルスコープで作業日エイリアスを作成すると、ID は workday に設定されます。 ▪ HR アプリケーションスコープで作業日エイリアスを作成すると、ID は x_hr_app.workday に設定されます。
親エイリアス	親エイリアスを選択するオプション。選択したエイリアスの下に、この接続および資格情報エイリアスを作成します。作成している接続および資格情報エイリアスは、子エイリアスになります。子エイリアスは、親「接続および資格情報エイリアス」ページの [子エイリアス] タブに一覧表示されます。 
タイプ	作成しているエイリアスのタイプを示すオプション。次のオプションから選択します。 ▪ 認証情報：認証情報レコードを含むエイリアス。 ▪ 接続と資格情報：接続および資格情報レコードの両方を含むエイリアス。このオプションはデフォルトで選択されています。 [接続と資格情報] が選択されていることを確認します。
複数の有効な接続をサポート	エイリアスが複数の有効な接続をサポートするかどうかを指定する指定子です。接続テーブルを使用して接続を追加し、接続関連リストを使用してその接続をエイリアスに関連付けます。

フィールド	説明
デフォルトの再試行ポリシー	エイリアスの再試行ポリシーです。詳細については、「再試行ポリシー」を参照してください。
構成テンプレート	構成テンプレートを選択するオプション。このテンプレートに基づいて接続および資格情報エイリアスを作成します。前の手順で作成した **[OAuth** クライアント認証情報権限許可タイプを使用した **HTTP** 接続 (外部ストレージ)] タイプのテンプレートを選択します。

h. [送信] を選択します。
接続および資格情報エイリアスレコードが作成されました。

i. 移動先 すべて > 統合ハブ > 接続ダッシュボード.

j. [すべての接続を検索] フィールドに、作成した「接続および資格情報エイリアス」レコードの名前を入力します。

k. 接続および資格情報エイリアスレコードで、[詳細を表示] を選択します。

l. [構成] を選択します。

m. フォームに入力します。

接続の設定フォーム

フィールド	説明
接続名	接続の名前。名前を更新することはできません。
接続 URL	サードパーティサーバーの URL。
MID を使用	MID サーバーを有効にするオプション。 **重要:** このオプションが選択されていることを確認します。
MID 選択	接続の MID 構成を指定するオプション。 ▪ 特定の **MID** サーバー：特定の MID サーバーの手動選択を示すオプション。 ▪ **MID** サーバーの自動選択：MID サーバーを自動的に選択することを示すオプション。 ▪ 特定の **MID** クラスター：複数の MID サーバーから構成される MID クラスターの手動選択を示すオプション。

フィールド	説明
	重要: 選択した MID サーバーが、CyberArk ボールトにアクセスするよう構成された CyberArk AIM クライアントと同じマシン上に存在することを確認します。「CyberArk ボールトの設定と AIM API のインストール」を参照してください。
MID サーバー	MID サーバーを手動で指定するオプション。このオプションは、**[MID** 選択] フィールドで [特定の **MID** サーバー] を選択した場合に表示されます。
MID クラスター	手動で MID クラスターを指定するオプション。このオプションは、**[MID** 選択] フィールドで [特定の **MID** クラスター] を選択した場合に表示されます。
外部の認証情報ストア	CyberArk 外部認証情報ストレージを使用するオプション。このオプションを選択すると、MID サーバーは外部認証情報ストレージから OAuth 2.0 認証情報 (クライアント ID とクライアントシークレット) を取得します。 **重要:** このオプションが選択されていることを確認します。
認証情報 ID	クライアント ID とクライアントシークレットの詳細を保持する CyberArk アカウントの識別子。認証情報識別子を取得する手順については、「CyberArk 上の OAuth 2.0 認証情報の構成」を参照してください。 [認証情報 ID] フィールドに、次のいずれかの形式で式を入力します。 ▪ すべての認証情報が同じセーフにある場合は、ext.cred.safe_name パラメーターを使用して MID サーバーの config.xml ファイルにこの安全な名前を設定してから、認証情報 ID を **<credential ID>** として名前のみで指定します。 ▪ 特定のセーフにある指定したプラットフォームの認証情報に名前を付けるには、認証情報 ID を **<safe>:<credential ID>:<platform ID>** として定義します。 ▪ 認証情報が複数のセーフにある場合は、認証情報 ID を **<safe>:<credential ID>** の形式で指定します。

自動翻訳

フィールド	説明
OAuth トークン URL	OAuth サーバーからアクセストークンを取得するエンドポイントを指定する URL。

n. [接続を設定] を選択します。

2. 接続を手動で構成します。

a. 移動先 すべて > システム **OAuth** > アプリケーションレジストリー.

b. **[New** (新規)] を選択します。

c. [外部ボールトを使用してサードパーティ **OAuth** プロバイダーに接続] を選択します。

d. フォームに入力します。

新しいアプリケーションレジストリの詳細

フィールド	説明
名前	アプリケーションレジストリレコードを識別する名前です。たとえば、「MID アプリを介した OAuth 2.0 トークン要求」と入力します。
アプリケーション	このアプリケーションレジストリにアクセスできるアプリケーションの名前を指定するオプション。このフィールドは読み取り専用です。
デフォルトの権限許可タイプ	クライアントアプリケーションが OAuth サーバーからアクセストークンを取得するデフォルトの方法。デフォルトの読み取り専用権限許可タイプは [クライアント認証情報] です。
アクセス可能	このアプリケーションレジストリにアクセスできるアプリケーションを指定するオプション。
トークン URL	OAuth サーバーからアクセストークンを取得するエンドポイントを指定する URL。
認証情報の送信	要求本文で OAuth 2.0 を送信する方法を指定するオプション。
コメント	関連するコメントを入力します。

e. **[OAuth** スコープ] 列で、次の手順を実行して 1 つ以上の OAuth スコープを作成します。

i. [名前] 列でフィールドをダブルクリックし、OAuth スコープの名前を入力します。

ii. [OAuth スコープ] 列で、フィールドをダブルクリックしてスコープを入力します。

f. [送信] を選択します。
OAuth エンティティプロファイルとアプリケーションレジストリが作成されました。

g. 移動先 すべて > 統合ハブ > 接続 **&** 認証情報 > 認証情報.

h. **[New** (新規)] を選択します。

i. **[OAuth 2.0** 認証情報] を選択します。

j. [外部ストレージビュー] を選択します。

> **重要:** ビューが OAuth 2.0 認証情報外部ストレージフォームのビューと異なる場合にのみ、[外部ストレージビュー] を選択します。

k. フォームに入力します。

OAuth 2.0 認証情報

フィールド	説明
名前	認証情報レコードの名前。
適用先	すべての MID サーバーまたは特定の MID サーバーに認証情報レコードが適用可能な場合に指定するオプション。特定の MID サーバーの場合は、MID サーバーを追加します。 次のオプションからいずれかを選択します。 ▪ すべての MID サーバー：MID サーバーのコレクションから MID サーバーが自動的に選択されます。 ▪ 特定の MID サーバー：1 つ以上の MID サーバーを指定するオプション。
MID サーバー	1 つ以上の MID サーバーを指定するオプション。 **注:** このフィールドは、[適用先] フィールドで [特定の **MID** サーバー] を選択した場合に表示されます。
アクティブ	認証情報レコードが使用可能かどうかを指定するオプション。デフォルトでは使用可能です。
OAuth エンティティプロファイル	認証情報で使用する OAuth エンティティプロファイルを指定するオプション。上で作成した OAuth エンティティプロファイルを選択します。「OAuth エンティティプロファイルの構成」を参照してください。
外部の認証情報ストア	認証情報を ServiceNow インスタンスではなく外部ストレージに保存することを指定するオプション。 **重要:** このオプションが選択されていることを確認します。
認証情報 ID	クライアント ID とクライアントシークレットを保持する CyberArk アカウントの認証情報識別子を指定するオプション。認証情報識別子を取得する手順については、「CyberArk

自動翻訳

フィールド	説明
	上の OAuth 2.0 認証情報の構成」を参照してください。
認証情報ストレージ Vault	外部認証情報ストレージボールトの名前を指定するオプション。CyberArk を選択していることを確認します。

l. [送信] を選択します。
認証情報レコードが作成されました。

m. 接続および資格情報エイリアスを作成します。
手順については、「接続情報および認証情報エイリアスの作成」を参照してください。

n. 移動先 すべて > 統合ハブ > 接続.

o. **[New** (新規)] を選択します。

p. **[HTTP(s)** 接続] を選択します。

q. フォームに入力します。

接続フォーム

フィールド	説明
名前	この HTTP(S) 接続の一意の名前。
アクティブ	作成している接続をアクティブに設定するオプション。このオプションはデフォルトで選択されています。
認証情報	接続を許可するために使用する認証情報レコードを選択します。上で作成した認証情報を選択します。
接続エイリアス	この接続に関連付けるエイリアス レコードを選択します。エイリアスを使用すると、エイリアスを使用するアクションまたはアクティビティを再設定することなく、接続レコードを更新できます。
URL ビルダー	手動で接続 URL を入力するか、システムを使用して入力に基づいて URL を作成します。デフォルトはオフです。オンにすると、次のフィールドから接続 URL が計算されます。 ▪ [相互認証] - 相互認証を使用する場合はオンにします。 ▪ [プロトコル] - 相互認証を使用しない場合は、プロトコルを入力します。デフォルトは HTTPs です。 ▪ [プロトコル プロファイル] - 相互認証を使用する場合は、sys_protocol_profile からプロトコル プロファイルを入力します。 ▪ ホスト

フィールド	説明
	▪ ポート ▪ ベースパス - 接続文字列のパス。 注: 相互認証をオンにした場合は、接続 URL が作成されます (プロトコル + :// + host:port +URL)。相互認証をオフにした場合は、接続 URL が作成されます (プロトコル プロファイル + :// + host:port +URL)。
接続 URL	URL ビルダーをオフにした場合は、このフィールドに接続 URL を入力します。 注: 相互認証をオンにした場合は、接続 URL が作成されます (プロトコル + :// + host:port +URL)。相互認証をオフにした場合は、接続 URL が作成されます (プロトコル プロファイル + :// + host:port +URL)。
MID サーバーを使用	MID サーバー経由で OAuth トークン要求を送信することを指定するオプション。 重要: このオプションが選択されていることを確認します。
接続タイムアウト	システムがホスト接続の成功を待機するミリ秒数。この間に接続が成功しない場合、接続要求はタイムアウトします。システムのデフォルトの接続タイムアウト値を使用するには、このフィールドを空のままにします。
MID 選択	以下のオプションのいずれかを指定するオプション。 ▪ MID サーバーの自動選択：MID サーバーがクラスターのメンバーであるかどうかに関係なく、MID サーバーの基準に基づいて MID サーバーから選択します。 ▪ 特定の MID サーバー：MID サーバーを手動で選択します。 ▪ 特定の MID クラスター：別の MID サーバーへの自動再アサインでは、指定されたクラスターのメンバーからのみ選択されます。 重要: 選択した MID サーバーが、CyberArk ボールトにアクセスするよう構成された CyberArk AIM クライアントと同じマシン上に存在することを確認します。「CyberArk ボールトの設定と AIM API のインストール」を参照してください。

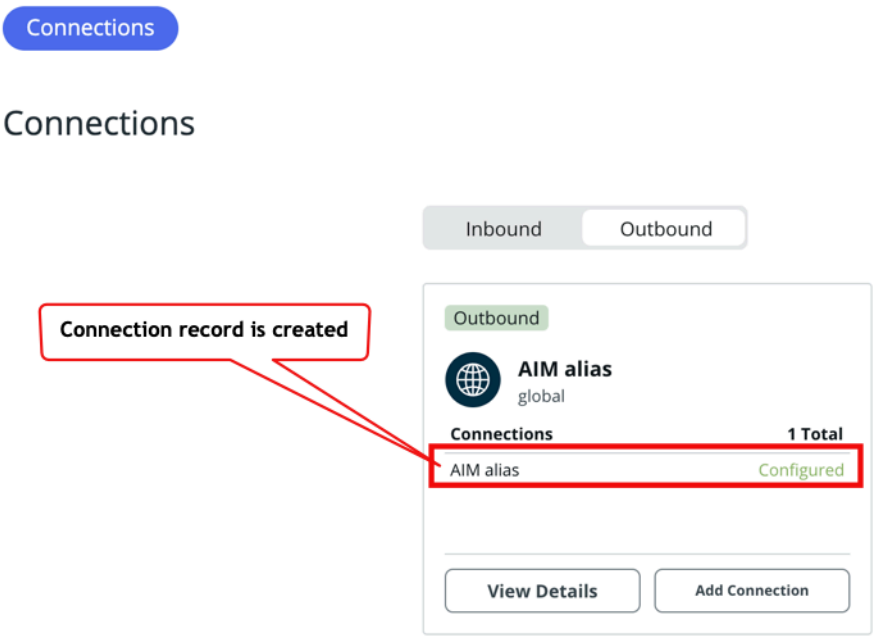
フィールド	説明
機能	1 つ以上の MID サーバー機能を選択するオプション。機能は、IP アドレス範囲内の MID サーバーの特定の機能を定義し、アプリケーションが最適な MID サーバーを選択できるようにします。MID 機能 を選択して、1 つ以上の機能を選択します。 注: ▪ 選択した MID サーバーが、CyberArk ボールトにアクセスするよう構成された CyberArk AIM クライアントと同じマシン上に存在することを確認します。「CyberArk ボールトの設定と AIM API のインストール」を参照してください。 ▪ このオプションは、[MID 選択] フィールドで **[MID** サーバーの自動選択] を選択した場合に表示されます。
MID サーバー	MID サーバーを手動で選択します。このオプションは、[MID 選択] フィールドで [特定の MID サーバー] を選択した場合に表示されます。
MID クラスター	MID クラスターを手動で選択します。このオプションは、[MID 選択] フィールドで [特定の MID クラスター] を選択した場合に表示されます。
MID アプリケーション	MID アプリケーションを指定するか、デフォルトのアプリケーション選択を承認するオプション。 注: このオプションは、[MID 選択] フィールドで **[MID** サーバーの自動選択] を選択した場合に表示されます。デフォルトでは、[すべて] オプションが選択されています。

r. **[Submit** (送信)] を選択します。
HTTP(s) 接続レコードが作成されました。

s. 移動先 すべて > プロセス自動化 > 統合ハブ > 接続ダッシュボード.

t. [すべての接続を検索] フィールドに、作成した接続レコードの名前を入力します。
接続エイリアスレコードが表示されます。

OAuth 2.0 接続レコードが作成されました。



認証アルゴリズム

認証アルゴリズムを使用して送信者の本人確認を行う

統合ステップで、複雑な、または非標準の接続メカニズムや認証情報メカニズムを必要とする Web サービスを使用した認証が可能になります。認証アルゴリズムを認証情報エイリアスや接続エイリアスに関連付けることで、統合ステップを手動で設定する必要がなくなります。

認証アルゴリズムを使用して、統合ステップで使用するカスタム認証データを生成できます。統合ステップでは、この動的データを使用して、ターゲット Web サービスを使用した認証に必要なカスタムアーティファクトを作成できます。たとえば、REST ステップでは、認証ヘッダーやクエリーパラメーター、トークンを作成できます。

認証アルゴリズムは、次のステップをサポートしています。

- 接続情報の取得ステップ
- REST ステップ
- SOAP ステップ

詳細については、「統合ステップ」を参照してください。

認証アルゴリズムのタイプ

- **Amazon** 署名バージョン **4**：これは、Amazon Web Services に接続するためのビルド済み認証アルゴリズムです。
- カスタム認証：これは、開発者が独自の認証アルゴリズムを作成するために使用できるテンプレートです。

認証アルゴリズムの設定方法の詳細については、「認証アルゴリズムの構成」を参照してください。

スクリプト

インスタンス認証スクリプトは、*sys_script_include* テーブルのインスタンススクリプト部分にあります。

インスタンスでの認証スクリプト

RequestAuthInternal	送信要求と一緒に送信される AWS V4 署名またはカスタム認証の生成をサポートする、インスタンスの読み取り専用スクリプト。
RequestAuthAWSV4Signer	AWS V4 署名を生成する署名者を実装するための RequestAuthInternal を拡張するスクリプト。
RequestAuthTwitterSigner	OAuth 1.0a を使用して Twitter の署名を生成する署名者を実装するための RequestAuthInternal を拡張するスクリプト。
RequestAuthSampleCustomSigner	インスタンスでカスタム署名者を記述する方法を理解するための RequestAuthInternal を拡張するサンプルスクリプト。

MID 認証スクリプトは、*ecc_agent_script_include* テーブルの MID スクリプト部分にあります。

MID での認証スクリプト

RequestAuthInternal	送信要求と一緒に送信される AWS V4 署名またはカスタム認証の生成をサポートする、MID の読み取り専用スクリプト。
RequestAuthAWSV4MIDSigner	AWS V4 署名を生成する署名者を実装するための RequestAuthInternal を拡張するスクリプト。
RequestAuthTwitterSigner	OAuth 1.0a を使用して Twitter の署名を生成する署名者を実装するための RequestAuthInternal を拡張するスクリプト。
RequestAuthSampleMidCustomSigner	MID でカスタム署名者を記述する方法を理解するための RequestAuthInternal を拡張するサンプルスクリプト。

JavaScript API

認証アルゴリズム用の JavaScript API は次のとおりです。

- AuthCredential
- HttpRequestAuthedData
- HttpRequestData
- RequestAuthAPI

認証アルゴリズムの構成

送信 HTTP 要求に署名できるように、認証アルゴリズムを構成します。

始める前に
認証アルゴリズムを構成する前に、スクリプトインクルードを構成しておく必要があります。
必要なロール：admin

手順

1. 移動先 すべて > 認証情報 **&** 接続 > 認証アルゴリズムをクリックし、[新規] をクリックします。
2. フォームで、フィールドに入力します。
   [フォーマット] フィールドのデータベース選択により、利用可能なフィールドが決定されます。

   認証フォーム

フィールド	説明
名前	このアルゴリズムの一意の名前
アルゴリズム	発信リクエストタイプ
説明	アルゴリズムの実行内容の説明
アプリケーション	アプリケーションが実行されるスコープ
インスタンスでの認証スクリプト	スクリプトインクルードテーブルから選択したスクリプト
MID での認証スクリプト	MID サーバースクリプトインクルード [ディスカバリービュー] テーブルから選択したスクリプト

3. [送信] をクリックします。

Amazon 署名ベースのカスタムアルゴリズムの設定

スクリプトを実行して、Web サービスへの認証に必要な Amazon 署名ベースのデータを生成します。

始める前に

- JavaScript に関する知識
- REST に関する知識
- ターゲットの Web サービス API に関する知識
- 接続、資格情報、およびエイリアスに関する知識
- 必要なロール：開発者

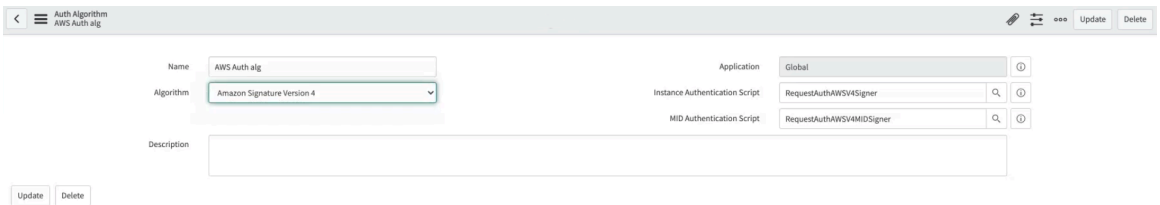
このタスクについて
認証には接続エイリアスおよび認証情報エイリアスと Amazon 署名バージョン 4 ベースのアルゴリズムを使用します。

手順

1. 移動先 すべて > 認証情報 **&** 接続 > 認証アルゴリズムをクリックし、[新規] をクリックします。
2. フォームで、フィールドに入力します。
   [フォーマット] フィールドのデータベース選択により、利用可能なフィールドが決定されます。

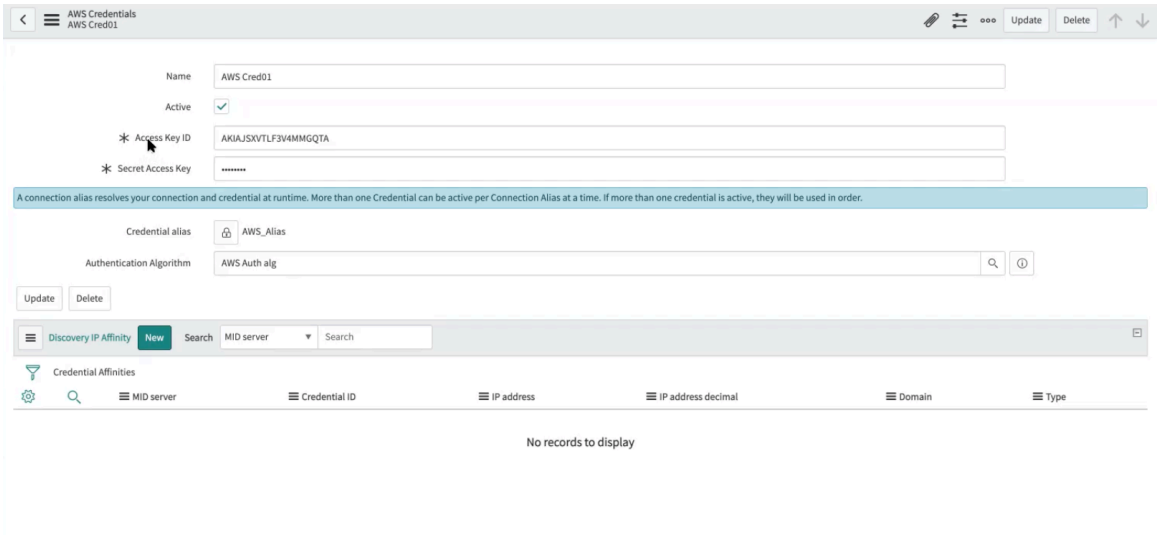
認証フォーム

フィールド	説明
名前	このアルゴリズムの一意の名前
アルゴリズム	発信リクエストタイプ**[Amazon** 署名バージョン **4]** を選択します。
説明	アルゴリズムの実行内容の説明
アプリケーション	アプリケーションが実行されるスコープ
インスタンスでの認証スクリプト	スクリプトインクルードテーブルから選択したスクリプト**[Amazon** 署名バージョン **4]** アルゴリズムの場合は、**[RequestAuthAWSV4Signer]**を選択します。使用可能なスクリプトは次のとおりです。 ○ RequestAuthAWSV4Signer ○ RequestAuthInternal ○ RequestAuthSampleCustomSigner ○ RequestAuthTwitterSigner **注:** スクリプトの詳細を確認するには、フィールドの横にある情報アイコンをクリックします。[名前]、[API 名]、[アプリケーション]、[アクセス可能]、[スクリプト] などのスクリプトの詳細が表示されます。
MID での認証スクリプト	MID サーバースクリプトインクルード [ディスカバリービュー] テーブルから選択したスクリプト使用可能なスクリプトは次のとおりです。 ○ RequestAuthAWSV4Signer ○ RequestAuthInternal ○ RequestAuthSampleCustomSigner ○ RequestAuthTwitterSigner



3. **[Update** (更新)] をクリックします。
4. 移動先 すべて > 接続 **&** 認証情報 > 認証情報.
5. [**New**] をクリックします。
6. 認証アルゴリズムを使用して AWS 認証情報を作成します。
   この場合は **AWS Auth alg** です。
7. 以下の項目を指定します。
   - 名前
   - 有効

- アクセスキー ID
- 秘密アクセスキー
- 認証情報エイリアス
- 認証アルゴリズム



8. [更新] をクリックします。

結果
選択したスクリプトと認証アルゴリズムに基づき、設定された認証情報 (アクセスキー **ID** と秘密アクセスキー)、またはユーザーの認証情報 (アクセスキー **ID**、秘密アクセスキー、セッショントークン) によって、ServiceNow からプロバイダー (この場合は AWS) に送信要求として送信される Amazon V4 署名が生成されます。

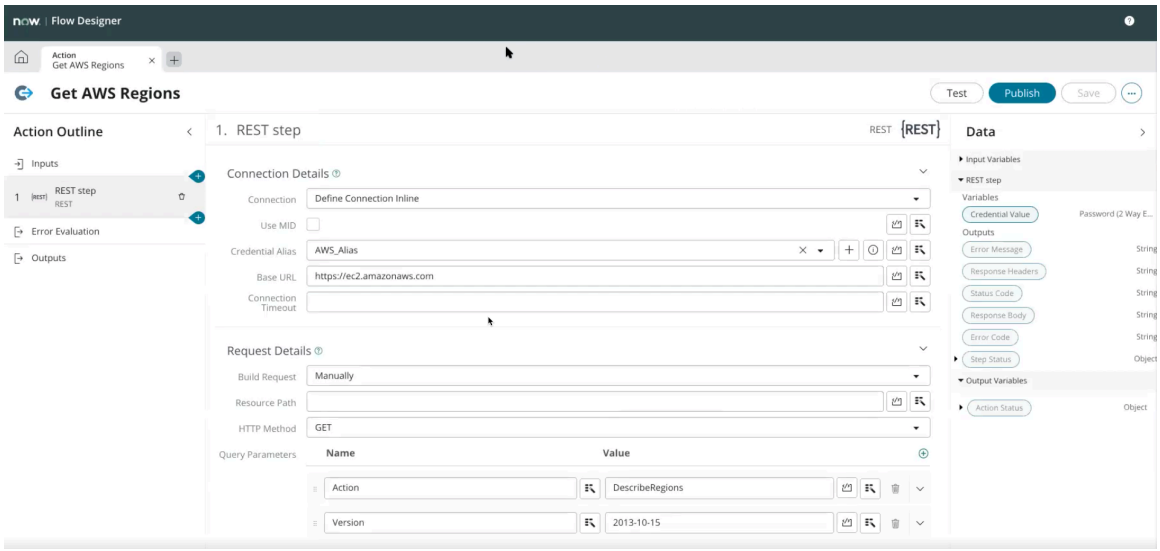
Example: AWS での REST ステップ

注: Amazon V4 署名ベースの認証は、スクリプトのバックグラウンドからも使用できます。

アクション：AWS リージョンを取得する

AWS での REST ステップを次のように入力します。

- 認証情報エイリアス：AWS 用に作成されたエイリアス。
- ベース **URL**：AWS からのベース URL の詳細。
- **HTTPS** メソッド：この場合は GET メソッドです。
- クエリーパラメーター：アクション。値は **DescribeRegions**。



アクションをテストできます。テストすると関連するリージョンが表示されます。応答の本文は次のとおりです。

自動翻訳

Amazon V4 は、認証メカニズムをサポートするアルゴリズムの標準セットで定義されています。このアルゴリズムを使用すると、REST ステップを使用して認証 (HTTP 要求) の認証ヘッダーとして署名が追加されます。

カスタムの認証アルゴリズムの構成

スクリプトを実行して、Web サービスへの認証に必要なカスタムのデータを生成します。

始める前に

- JavaScript に関する知識
- REST に関する知識
- ターゲットの Web サービス API に関する知識
- 接続、資格情報、およびエイリアスに関する知識
- 必要なロール：開発者

このタスクについて
認証には接続エイリアスおよび認証情報エイリアスと、カスタム認証ベースのアルゴリズムを使用します。

手順

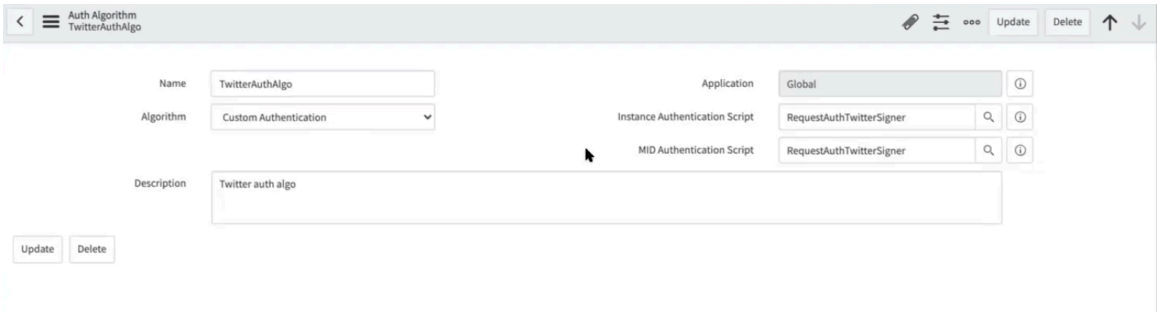
1. 移動先 すべて > 認証情報 **&** 接続 > 認証アルゴリズムをクリックし、[新規] をクリックします。
2. フォームで、フィールドに入力します。
   [フォーマット] フィールドのデータベース選択により、利用可能なフィールドが決定されます。

認証フォーム

フィールド	説明
名前	このアルゴリズムの一意の名前
アルゴリズム	発信リクエストタイプ[カスタム認証] を選択します。
説明	アルゴリズムの実行内容の説明
アプリケーション	アプリケーションが実行されるスコープ
インスタンスでの認証スクリプト	スクリプトインクルードテーブルから選択したスクリプト使用可能なスクリプトは次のとおりです。 ○ RequestAuthAWSV4Signer ○ RequestAuthInternal ○ RequestAuthSampleCustomSigner ○ RequestAuthTwitterSigner **注:** ○ スクリプトの詳細を確認するには、フィールドの横にある情報アイコンをクリックします。[名前]、[API 名]、[アプリケーション]、[アクセス可能]、[スクリプト] などのスクリプトの詳細が表示されます。 ○ Twitter でのカスタム認証の場合は、**RequestAuthTwitterSigner**を選択できます。この方法では OAuth 1.0a の認証方法を使用するためです。その際、認証ヘッダーで渡す署名の作成に使用できる **API** キーとシークレット、アクセストークンとシークレットなどの情報が必要です。詳細については、「Twitter での認証(Authentication in Twitter) 」を参照してください。
MID での認証スクリプト	MID サーバースクリプトインクルード [ディスカバリービュー] テーブルから選択したスクリプト使用可能なスクリプトは次のとおりです。

自動翻訳

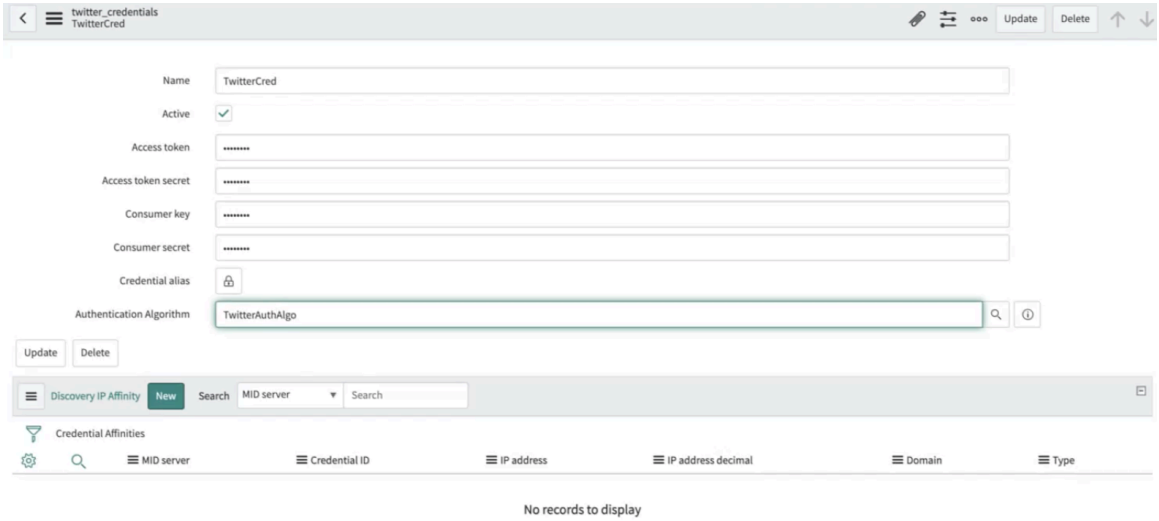
フィールド	説明
	○ RequestAuthAWSV4Signer ○ RequestAuthInternal ○ RequestAuthSampleCustomSigner ○ RequestAuthTwitterSigner



選択したスクリプトと認証アルゴリズムに基づき、設定された認証情報が送信要求として ServiceNow からプロバイダーに送信されます。

3. **[Update** (更新)] をクリックします。
4. 移動先 すべて > 接続 **&** 認証情報 > 認証情報.
5. [**New**] をクリックします。
6. 認証アルゴリズムを使用して Twitter 認証情報を作成します。
   この場合は **TwitterAuthAlgo** です。
7. 以下のフィールドを指定します。
   - 名前
   - 有効
   - アクセストークン
   - アクセストークンシークレット
   - コンシューマーキー
   - コンシューマーシークレット
   - 認証情報エイリアス
   - 認証アルゴリズム

自動翻訳



8. [更新] をクリックします。

Example: Twitter での REST ステップ

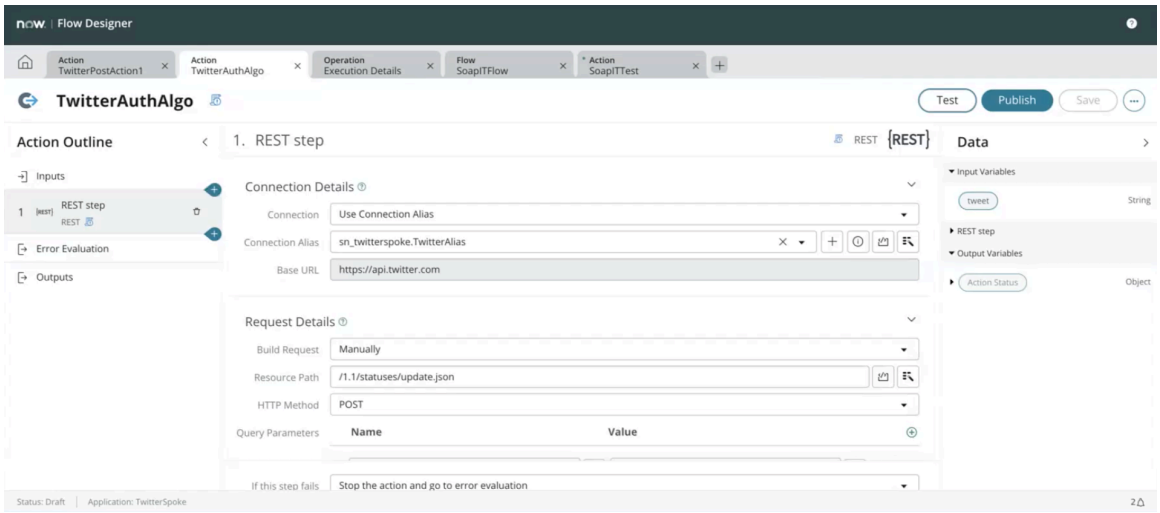
Twitter の場合は、次のスポークまたは認証情報が利用可能であることを確認する必要があります。

- アクセストークン
- アクセストークンシークレット
- コンシューマーキー
- コンシューマーシークレット
- 認証アルゴリズム

アクション：TwitterAuthAlgo

Twitter で REST ステップを次のように入力します。

- 認証情報エイリアス：Twitter 用に作成されたエイリアス。
- ベース **URL**：Twitter からのベース URL の詳細。
- **HTTPS** メソッド：この場合は POST メソッドです。ツイートを投稿します。
- クエリーパラメーター：[アクション] が **tweet**。



アクションをテストできます。ツイートが Twitter ページに投稿されます。

ディスカバリー と オーケストレーション における **IP** サービスの親和性の確認

IP サービステーブルで、プロトコルに関連付けられている IP アドレスのリストを確認できます。

始める前に

必要なロール：admin

このタスクについて

IP サービステーブルは、ポートをプロトコルにマッピングします。HTTP の場合はポート 80、SSH の場合はポート 22、SNMP の場合はポート 161 など、一般的に使用されるポートとプロトコルの組み合わせに対して、デフォルトでいくつかのマッピングが提供されています。
`glide.discovery.ip_service_affinity` と呼ばれるシステムプロパティにより、検出された IP アドレスの最後のポートを ディスカバリー で記憶できます。

重要: 組織でカスタムポートを使用しない限り、IP サービスを変更しないでください。

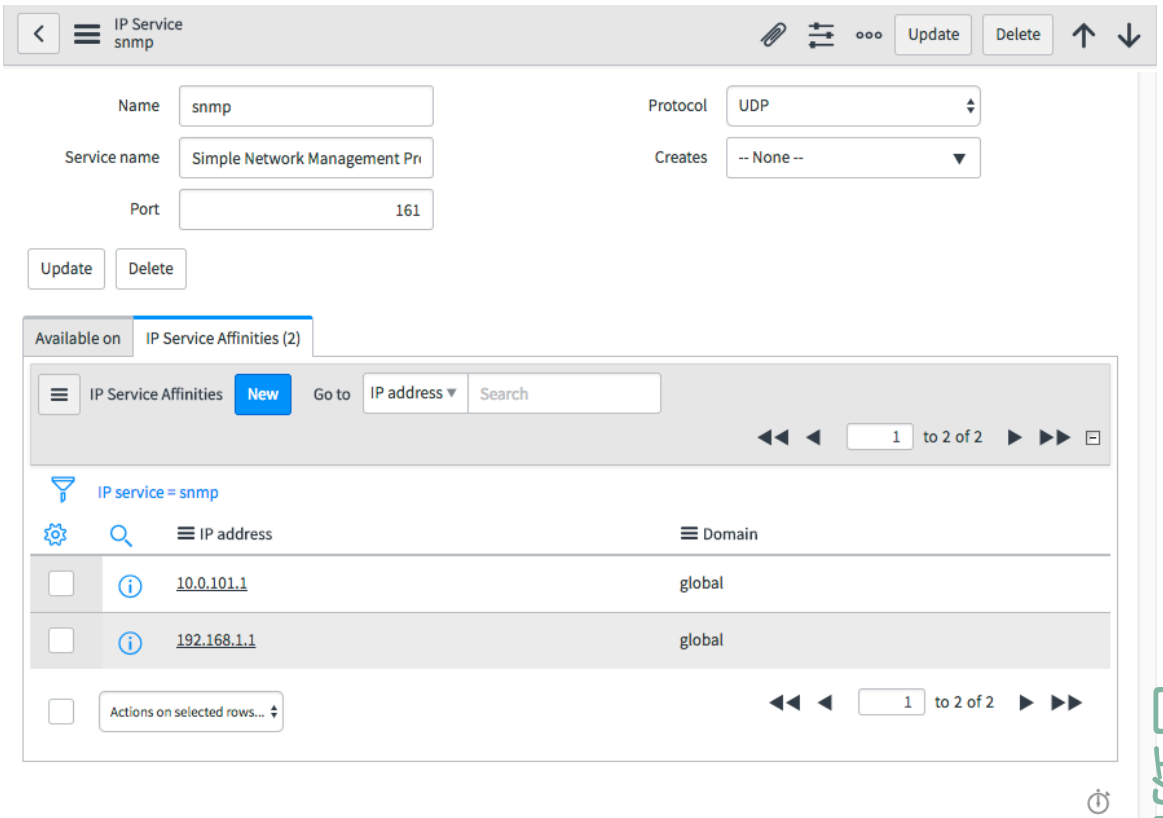
手順

1. 移動先 すべて > ディスカバリー定義 > **IP** サービス.
2. リストをフィルタリングして、適切な IP サービスを探します。
3. そのサービスの名前をクリックしてその IP サービスページに移動します。

自動翻訳

4. そのサービスに関連付けられている IP アドレスのリストの **[IP** サービスの親和性] タブをクリックします。

IP サービスの親和性



ServiceNow® アクセス制御

SNC アクセス制御プラグイン (com.snc.snc_access_control) を有効にして、どのカスタマーサービス & サポート従業員がインスタンスにいつアクセスできるかを制御できます。

ServiceNow アクセス制御の詳細

ServiceNow アクセス制御を有効にする

ServiceNow® アクセス制御の機能について説明します。

アクティブ ServiceNow® アクセス制御。

ServiceNow アクセス制御を設定する

ServiceNow® アクセス制御の構成方法について説明します。

監査ログ

ServiceNow® アクセス制御監査ログについて確認します。

自動翻訳

Explore ServiceNow® アクセス制御

SNC アクセス制御プラグイン (com.snc.snc_access_control) を有効にして、どのカスタマーサービス & サポート従業員がインスタンスにいつアクセスできるかを制御できます。

初めてプラグインを有効にすると、カスタマーサービス & サポート の従業員はインスタンスにログインできません。現在ログインしている カスタマーサービス & サポート の従業員はログインしたままになります。特定の SNC 従業員またはすべての従業員にアクセス権を付与するレコードを SNC アクセス制御テーブルに作成します。

プラグインにより、カスタマーサービス & サポートの担当者が特別な許可なしにインスタンスにアクセスできないようにします。ただし、製品をサポートおよび管理する権限を持ち、使用状況を検証する、その他の認定された ServiceNow 運用担当者は、基礎となるインフラストラクチャで管理アクションを実行する必要があります。このインフラストラクチャには、SaaS を構成する他のインフラストラクチャコンポーネントの中でも、サーバーとデータベースが含まれます。このアクセス方法は監査可能および追跡済み。

このプラグインを使用すると、特別な権限なしにインスタンスへのアクセスを制限できるため、サポートのサービスレベルと可用性 SLA に影響を与える可能性があります。可用性 SLA は、サポートスタッフ担当者がインスタンスへのアクセスを許可された時間から測定されます。

ログインセキュリティ

権限のあるカスタマーサービス & サポートの従業員によるインスタンスへのログインのセキュリティは、安全なサーバーで生成された暗号化されたトークンを使用して確保されます。適切に認証されたカスタマーサービス & サポートの従業員のみがインスタンスへのアクセスを許可されます。SNC アクセス制御プラグインがない場合、セキュリティサーバーは hi.service-now.com でアクセス権を適用します。プラグインが有効な場合、暗号化されたログイントークンは、これらのレコードで定義された条件を使用して、プラグインで提供されるアクセスリストの名前と一致する必要があります。この認証方法により、カスタマーサービス & サポートの従業員の誰がいつインスタンスにアクセスできるかを正確に判断できます。

このシステムに選択されたアーキテクチャには、インスタンスのセキュリティを強化するように設計されたいくつかの機能があります。

セキュリティサーバー

セキュリティサーバーは、ServiceNow セキュリティ担当者のみがアクセスできる、ロックダウンされた Linux ホストです。このサーバーは、ログイントークンを生成するために必要な重要な秘密暗号化キーにアクセスできる唯一のシステムです。このコンパートメント化 (標準的なセキュリティプラクティス) を使用することで、万一、攻撃者が HI インスタンスを侵害した場合でも秘密キーが保護されます。

合成ユーザー

権限のあるカスタマーサービス & サポートの従業員がインスタンスにログインできるようにするインスタンスの機能では、そのインスタンスにアカウントをプロビジョニングする必要はありません。プロビジョニングされたユーザーレコードはなく、永続的な認証情報もありません。代わりに、カスタマーサービス & サポートの従業員のログオンごとに合成ユーザーが作成されます。このユーザーはメモリ内にのみ存在し、継続的な権限を提供しません。SNC アクセス制御プラグインが有効になっている場合、いつでもカスタマーサービス & サポートの従業員の許可を解除できます。

トークン

セキュリティトークンは、インスタンスと特定のカスタマーサービス & サポートの従業員に固有です。さらに、トークンを生成するメカニズムは、代理操作されたユーザーではなく、HI への実際のカスタマーサービス & サポート従業員ログインでのみ機能します。セキュリティトークンが生成されると、特定のカスタマーサービス & サポートの従業員のみがインスタンスにログインできます。

時間制限

セキュリティトークンは、生成されてから 4 時間後に期限切れになります。この有効期限は、この短い期間にのみ使用できるハイジャックされたトークンのユーティリティを制限します。

ログ記録

カスタマーサービス & サポートの従業員によるインスタンスへのログインは、ログインイベントとして記録されます。

- ログインしているカスタマーサービス & サポートの従業員が実行したすべてのアクションは、データベースのトランザクションログに追加されます。
- また、ほとんどの ServiceNowの従業員がアクセスできないファイルシステムのインスタンスログにも追加されます。
- ユーザー名はすべて @snc で終わるため、カスタマーサービス & サポートの従業員のログインとアクションは簡単に識別できます (例：frodo.baggins@snc)。

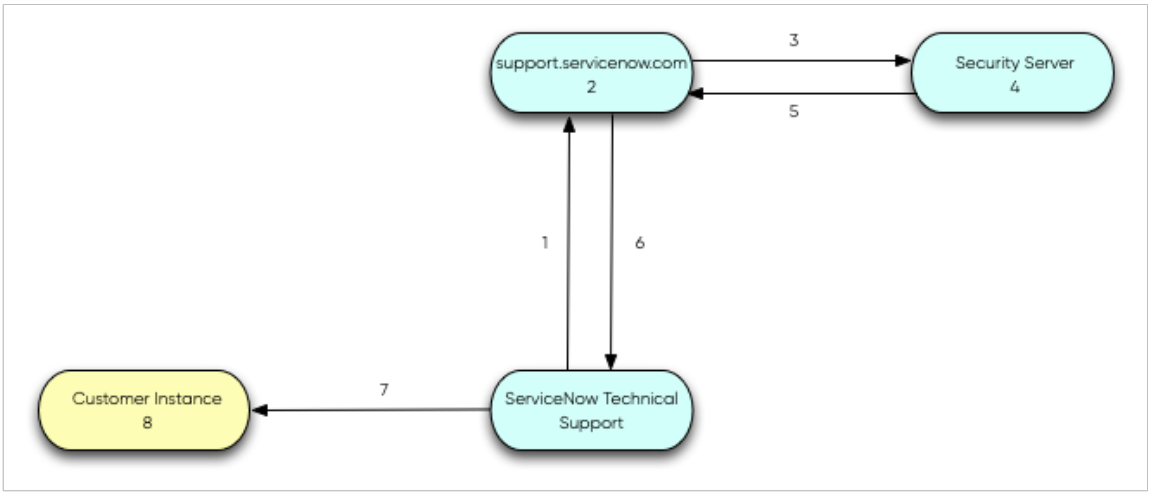
これらのアクションにより、従業員以外のアクセスに対して、使いやすく、堅牢で信頼性の高いセキュリティログ記録が提供されます。

セキュリティ処理フロー

カスタマーサービス & サポートの従業員がインスタンスにログインする場合、セキュリティ処理フローは次のようになります。

1. カスタマーサービス & サポートの技術者が hi.service-now.com からインスタンスへのログインを要求します。
2. HI は、技術者がインスタンスへのアクセスを許可する適切なロールを持っていることを確認します。
3. ユーザーが適切なロールを持っている場合、HI はアクセス要求をセキュリティサーバーに送信します。
4. セキュリティサーバーは、要求が HI IP アドレスから送信されたことを確認し、要求 (ユーザー、ロール、および要求者の IP アドレス) を評価します。要求が有効な場合、セキュリティサーバーはその要求を承認してトークンをビルドします。このトークンには、ユーザー名、ロール、インスタンス ID、および時刻 (4 時間のトークン有効期間の開始) が含まれています。最後に、セキュリティサーバーは秘密暗号化キーを使用してトークンを暗号化します。
5. セキュリティサーバーが暗号化されたトークンを HI に送信します。
6. HI はサポート技術者のブラウザにトークンを送信します。
7. サポート技術者のブラウザは、`@snc` で終わる特別なユーザー名を使用して、インスタンスへのログインを開始します。
8. インスタンスは公開鍵を使用してトークンを復号化します。トークンを検証するために、インスタンスはトークンを前のステップで指定されたユーザー名、インスタンス ID、および許可された時間枠と照合します。SNC アクセス制御プラグインが有効になっている場合、インスタンスはユーザーが次の条件を満たしていることを確認します。
   - リスト済み
   - 有効
   - 現在の期間内にインスタンスにアクセスするように設定されています。
9. ユーザーが認証されると、インスタンスは指定されたロールを持つ合成ユーザーをメモリ内に作成します。このユーザーは、時間制限が切れるか、ユーザーがログオフするか、またはインスタンスが再起動された後は存続しません。

ServiceNow セキュリティアクセスプロセスフロー



自動翻訳

監査ログ

次のログ記録は、カスタマーサービス & サポートの従業員によるログインとアクティビティを追跡します。

- イベントログ：イベントログには、インスタンスへのすべてのカスタマーサービス & サポートのログインが表示されます。
- トランザクションログ：トランザクションログには、ログの削除を含むインスタンスのすべてのアクティビティが表示されます。

注: このプラグインの詳細については、「インスタンスセキュリティ強化設定」の「SNC アクセスコントロールプラグインを有効化する (セキュリティセンター 1.3 で更新)」を参照してください。

ServiceNow®アクセス制御を有効にする

SNC Access Control プラグイン (com.snc.snc_access_control) のアクティブ化を要求できます。

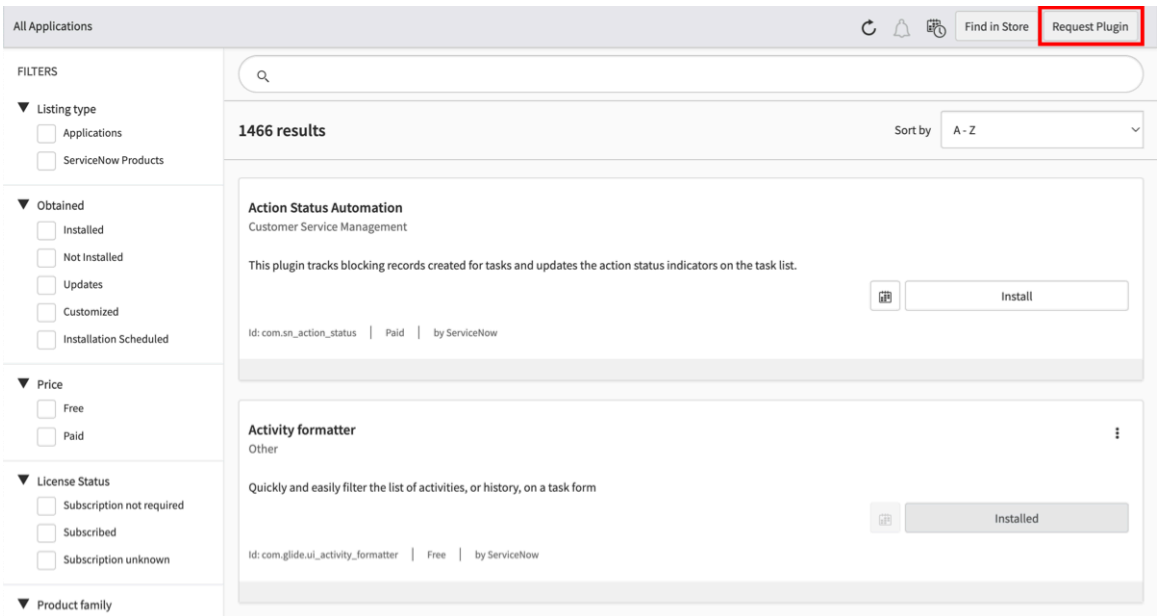
始める前に
必要なロール：admin

このタスクについて
プラグインを要求するには、次の 2 つの方法があります。

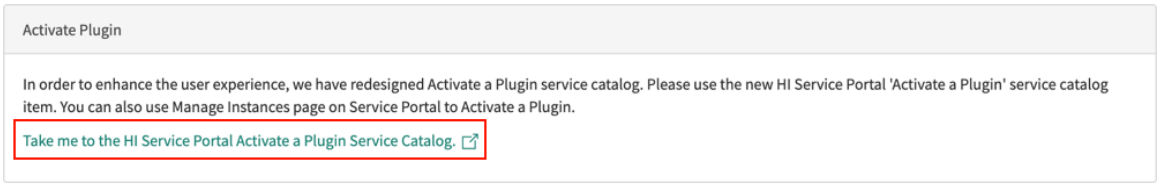
- Now Supportサービスカタログに直接アクセスするには、次を選択します。 すべて > サービスカタログ > プラグインをアクティブ化 Now Supportに。
- 次の手順に従い、インスタンスの [すべてのアプリケーション] ページから Now Support サービスカタログにアクセスします。

手順

1. 移動先 すべて > システムアプリケーション > 利用可能なすべてのアプリケーション > すべて.
2. [すべてのアプリケーション] ページで [プラグインの要求] をクリックして、Now Support で [プラグインをアクティブ化] フォームを開きます。



3. Now Support で、Now Support サービスポータル サービスカタログ にアクセスするリンクを選択します。



4. インスタンスを選択します。
5. [アクション] > [プラグインのアクティブ化] を選択します。
6. [プラグインのアクティブ化] フォームで、次の情報を入力します。

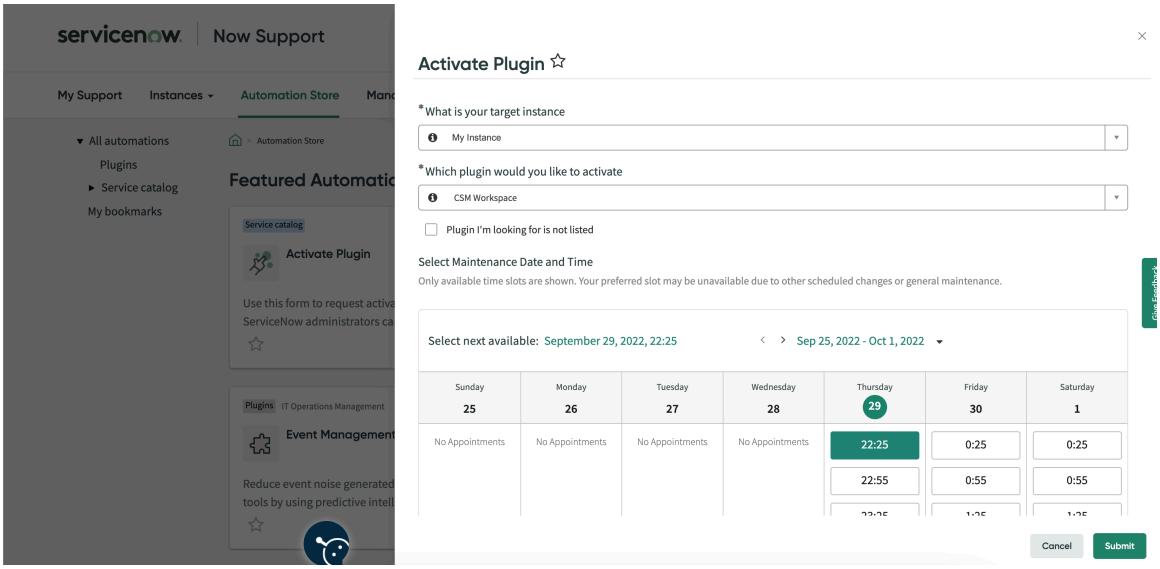
[プラグインのアクティブ化] フォーム

フィールド	説明
ターゲットインスタンスは何ですか	プラグインをアクティブ化するインスタンス。
どのプラグインをアクティブ化しますか	アクティブ化するプラグインの名前です。 **注:** 必要なプラグインが表示されない場合、または OEM またはオンプレミスのインスタンスでプラグインをアクティブ化している場合は、[探しているプラグインが表示されていません **(Plugin I'm looking for is not listed)]** チェックボックスをオンにして、プラグインの名前を入力します。
メンテナンスの日時を選択 (Select Maintenance Date and Time)	プラグインをアクティブ化する日時。 **注:** プラグインは、米国太平洋標準時で、毎営業日の朝と夕方の 2 回のバッチでアクティブ化されます。特定の時刻にプラグインをアクティブ化する必要がある場合は、[理由/コメント **(Reason/ Comments)]** フィールドに要求を入力します。

Example

たとえば、 [自分のインスタンス (My Instance)] という名前のインスタンスで CSM Workspace プラグインをアクティブ化するには、次のフォームを参照してください。

[プラグインのアクティブ化] フォーム



7. **[Submit** (送信)] を選択します。
   プラグインの要求の詳細については、次を参照してください。 のサービスカタログ [KB0751715] 記事からのプラグインの要求 Now Support ナレッジベース.

ServiceNow®アクセス制御の構成

アクセス制御レコードを設定して、インスタンスにログインする権限を持つ 1 人以上の カスタマーサービス & サポート の従業員を指定します。

始める前に

必要なロール：admin

このタスクについて

注: SNC アクセス制御 (com.snc.snc_access_control) プラグインにより、カスタマーサービス & サポートの担当者が特別な権限なくインスタンスにアクセスできなくなります。ただし、製品をサポートおよび管理する権限を持つ ServiceNow 運用担当者は、基礎となるインフラストラクチャで管理アクションを実行する必要があります。このインフラストラクチャには、SaaS を構成する他のインフラストラクチャコンポーネントの中でも、サーバーとデータベースが含まれます。このアクセス方法は監査可能および追跡済み。

このプラグインを使用すると、特別な権限なしにインスタンスへのアクセスを制限できるため、サポートのサービスレベルと可用性 SLA に影響を与える可能性があります。可用性 SLA は、サポートスタッフ担当者がインスタンスへのアクセスを許可された時間から測定されます。

手順

1. 移動先 すべて > システムセキュリティ > **SNC** アクセスコントロール.
2. [**New**] をクリックします。
3. フォームフィールドに入力します (表を参照)。
4. [送信] をクリックします。

自動翻訳

SNC アクセス制御

フォームのフィールド	説明
名前	このインスタンスにログインする権限を持つ各カスタマーサービス & サポートの従業員の名前。 ○ 名前を小文字の firstname.lastname として、ピリオドで区切って表します (例：john.smith)。各名前には、*support.servicenow.com* で対応するユーザーレコードが必要です。 ○ 複数のカスタマーサービス & サポートの従業員がこのインスタンスにログインする権限を持っている場合は、複数の名前を入力し、それらをカンマで区切ります。 ○ カスタマーサービス & サポートの従業員全員がインスタンスにアクセスできるようにするには、名前の代わりにアスタリスク (*) を入力します。 ○ インスタンスへのカスタマーサービス & サポート従業員のアクセスを制限する場合は、[名前] フィールドの値にアスタリスク (*) を使用しないでください。
理由	アクセス権を付与する理由を説明する、分かりやすいフィールド。このフィールドはオプションです。
開始	指定されたカスタマーサービス & サポートの従業員がログインアクセス権を持つ期間の開始日時を指定します。このフィールドは必須です。
終了	指定されたカスタマーサービス & サポートの従業員がログインアクセス権を持っている期間の終了日時を指定します。このフィールドは必須です。
有効	この権限レコードがアクティブかどうかを制御します。デフォルトは [アクティブ] です。

監査ログ

次のログ記録は、ServiceNowの従業員のログインとアクティビティを追跡します。

イベントログ	イベントログには、カスタマーインスタンスへのすべての ServiceNow ログインが表示されます。
トランザクションログ	トランザクションログには、ログの削除を含む、インスタンスのすべてのアクティビティが表示されます。

ID

インスタンス内の ID の詳細を確認します。

アクセスアナライザー

アクセスアナライザーは ServiceNow® アセット診断ツールである Tools アプリ。リソース (複数のインスタンス) にアクセスできるユーザーを判断するのに役立ちます。

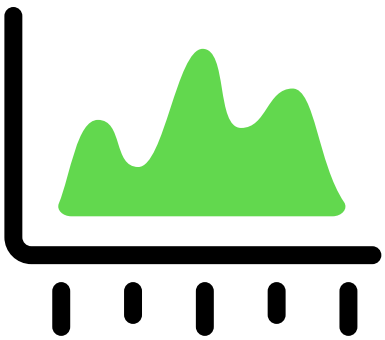
Global Identity

Global Identity は、ユーザー属性を 1 つのインスタンスから (複数の) インスタンスに更新できる ServiceNow® ストアアプリです。

ID センター

ID センターでは ID ベースのリスクとセキュリティギャップを監視および管理し、最小限に抑えることができます。

クロスドメイン ID 管理システム (SCIM)



クロスドメイン ID 管理システム (SCIM) API は、SCIM プロトコルを使用してユーザーとグループの操作を作成、読み取り、更新、および削除するためのエンドポイントを提供します。

ID とアクセスの監査

ID とアクセスの監査を使用して、ユーザー、グループ、ロール、および ACL の変更を把握します。

自動翻訳

アクセスアナライザー

ServiceNow® アクセスアナライザーは、リソースの ID の権限を表示するのに役立つアクセス診断ツールです。

探索	使用方法
	
アクセスアナライザーの機能とビジネス価値について説明します。	アクセスアナライザーの使用方法を理解します。

自動翻訳

権限評価

権限がどのように評価されるかを把握します。

よく寄せられる質問

アクセスアナライザーに関するよくある質問の詳細を確認します。

アクセスアナライザーの概要

ServiceNow インスタンスの ID を分析します。

ServiceNow アクセスアナライザーは、アドミニストレーターが選択したユーザー、ロール、グループの権限を表示するのに役立つアプリケーションです。

注:

- アクセスアナライザーは ServiceNow Store 製品です。ServiceNow ストア Web サイトにアクセスして、利用可能なすべてのアプリを表示し、要求をストアに送信する方法について確認してください。
- アクセスアナライザーは ID レコードを代理操作して権限に関する詳細を取得しますが、ID の個人データや機密データを読み取ったり保存したりすることはありません。
- Access Analyzer の評価結果は、Zero Trust アクセス (ZTA) などの、ユーザーに対して定義されているアクセスポリシーに関係なく同じです。ポリシーは実際のユーザーのログイン中にのみ評価され、Access Analyzer フロー中には評価されません。
- アクセスアナライザーには、管理対象スコープリソースと委任開発者に関連するリソースのアクセスの正確な評価に関する制限があります。

Evaluate Access

Evaluate Access は、ServiceNow アクセスアナライザーの機能の 1 つで、アドミニストレーターが選択したユーザー、ロール、グループの権限を表示するのに役立ちます。

これにより、ユーザー、グループ、テーブルのロール、クライアント呼び出し可能スクリプトインクルード、UI ページ、および REST エンドポイントの権限を分析および表示できます。

アクセスアナライザーを使用すると、組織はセキュリティ体制、ID ガバナンス、リスク管理を改善し、コンプライアンス目標を達成して、どのユーザー (ID) が何 (リソース) にアクセスできるかを把握できます。

Compare Access

Compare Access は、ServiceNow アクセスアナライザー V2 の機能の 1 つです。アドミニストレーターは、ユーザーアクセスを比較し、ServiceNow インスタンス上のユーザーの適切なアクセスレベルを決定できます。

Compare Access は、ユーザー記録とアクセス制御を目的としてユーザー間で実行できます。

Compare Access により、次の分析を実行できます。

- レベル 1：ユーザーレコードを比較して、属性、ロール、およびグループを把握します。
- レベル 2：アクセス制御を比較し、アクセスの問題を見つけて根本原因分析を実行します。

利点

アクセスアナライザーを使用するメリットは次のとおりです。

- リソース (テーブル) へのアクセスを分析します。
- 2 人のユーザーのアクセスを比較します。
- 2 人のユーザーのロールとグループを比較します。
- ID がリソース (テーブル) にアクセスできるかどうかを記載したレポートを生成します。
- 重要なセキュリティハイジーンのアクセス権を持つユーザーを把握します。
- 権限のオーバープロビジョニングの防止に役立ちます。
- アクセス制御を実装するときに、最低限の権限のプリンシパルを実現します。
- アプリケーション、テーブル、行または列、その他のリソースを含む特定のデータへのアクセスを制限します。
- アナライザーの結果のレポート機能を提供します。
- ユーザー記録とアクセス制御におけるアクセスを比較します。
- ServiceNow インスタンスのユーザーの適切なアクセスレベルを決定します。

Access Analyzer の使用

ServiceNow® インスタンスの ID とそのアクセスを分析します。

始める前に
必要なロール:admin、access_analyzer_admin

次の手順では、アクセスアナライザーにアクセスし、アクセスアナライザー内のさまざまな機能を使用するステップについて説明します。

注: アクセスアナライザーは ServiceNow® Store 製品です。

自動翻訳

手順

1. 移動先 すべて > アクセスアナライザー > 権限を分析.
   [アクセスと権限の分析 (Analyze access and permissions)] ホームページが表示されます。

   アクセスアナライザーには次の機能があります。

   - Evaluate Access
   - ユーザーレコードの比較
   - アクセス制御の比較
2. それぞれのタブを選択して、要件に基づいてアクセスアナライザーを使用します。

Evaluate Access® の使用

ServiceNow インスタンスの ID を分析します。

始める前に
必要なロール:admin、access_analyzer_admin

次の手順では、アクセスアナライザーの Evaluate Access にアクセスし、さまざまな機能を使用するステップについて説明します。

注: アクセスアナライザーは ServiceNow Store 製品です。

手順

1. 移動先 すべて > アクセスアナライザー > 権限を分析.
   [アクセスと権限の分析 (Analyze access and permissions)] ホームページが表示されます。
2. **[Evaluate Access]** タブを選択します。
3. 次のように基準を選択します。

アクセスと権限を分析するための基準を選択します

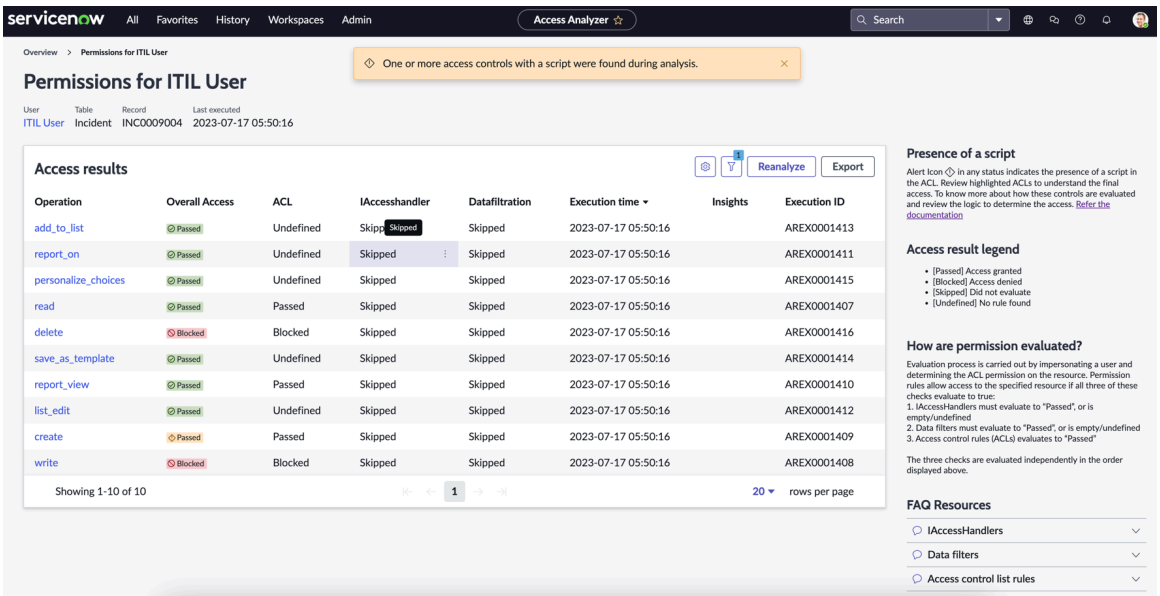
フィールド	説明
分析者 (Analyze by) *	ユーザー、ロール、グループのアクセスを分析します。
ユーザーを選択 *	リストから選択するユーザー名を指定します。
ルールタイプ *	テーブル、UI ページ、REST エンドポイント、またはクライアント呼び出し可能スクリプトインクルードのアクセスを分析します。
テーブルを選択 *	リストから選択するテーブル名を指定します。
レコードを選択	リストから選択するレコード名を指定します。
フィールドを選択	リストから選択するフィールド名を指定します。

4. [説明] フィールドに説明を指定します。
5. [権限を分析 **(Analyze Permissions)]** を選択します。

   ユーザーのアクセス結果が表示されます。同様に、次のルールタイプについて、グループまたはロールの権限を分析できます。

- テーブル (レコード)
- クライアント呼び出し可能スクリプトインクルード
- REST エンドポイント

アクセス結果が表示されます。



アクセス結果テーブルには、次のフィールドが含まれます。

アクセス結果

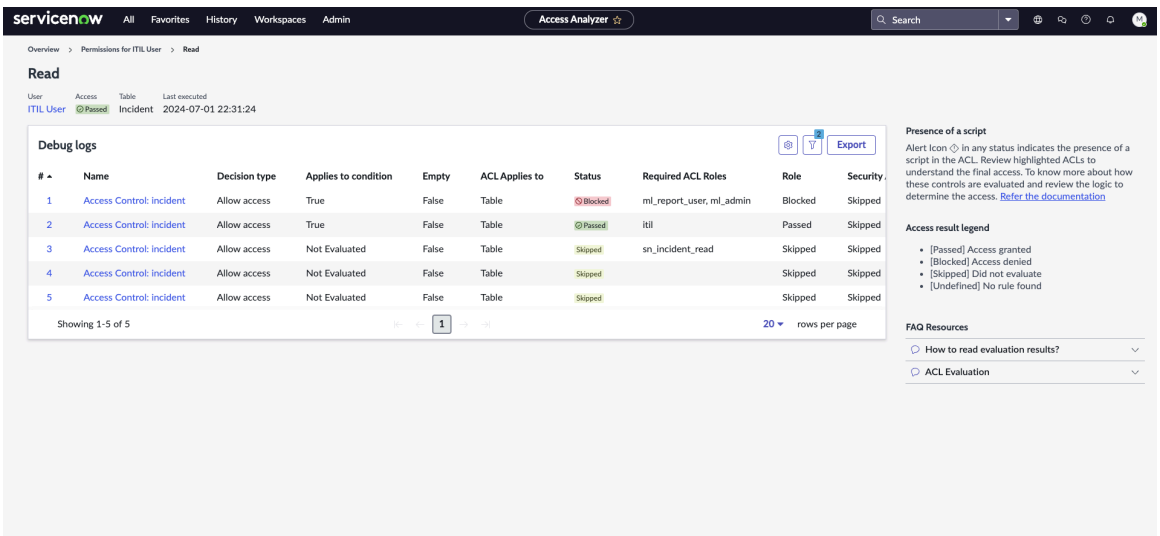
フィールド	説明
操作	選択したテーブル、レコード、またはフィールドに対してユーザー、グループ、またはロールが実行できる操作のタイプ。
全体的なアクセス (Overall Access)	全体的なアクセスの結果。結果は次のとおりです。 ○ [合格] アクセスが許可されました ○ [ブロック済み] アクセスが拒否されました ○ [スキップ済み] 評価されませんでした ○ [未定義] ルールが見つかりません
ACL	選択した操作に対して ACL が定義されているかどうか
アクセスハンドラー	プラットフォーム上の非表示のソースコードを使用する内部システムチェック。IAccessHandler は、ACL を評価せずにリソースへのアクセスを許可または拒否できます。IAccessHandler が無視されると、ACL が評価されます。
データフィルタリング	データフィルターは、インスタンスの既存のアクセス制御ルール (ACL) と連携して機能するように設計されたアクセス制御の形式です。

フィールド	説明
実行時間	アクセス結果が実行された時刻
インサイト	選択した操作に関する詳細情報
実行 ID	各アクセス結果実行の一意の ID

6. 操作を選択すると、ACL の詳細が表示されます。
   たとえば、 **[read]** を選択すると、読み取りに関連するアクセス制御が表示されます。

ACL の詳細

フィールド	説明
名前	ACL の名前。
意思決定タイプ	ACL に構成された意思決定タイプ。[アクセスを許可 **(Allow access)]** または [アクセスを拒否 **(Deny access)]**。
条件に適用	ACL が条件に適用されるかどうか。
ACL の適用先	ACL が適用されるリソースの詳細。
ステータス	ACL またはアクセス結果のステータス。
必要な ACL ロール	リソースにアクセスするために必要なロールの詳細。
ロール	ロールのステータス。合格、スキップ、またはブロック済み。



ユーザーの権限の表示

アクセスアナライザーを使用して選択したユーザーの権限を表示します。

始める前に

必要なロール:admin、access_analyzer_admin

次の例では、アクセスアナライザーの Evaluate Access を使用して、選択したユーザー (**[ITIL** ユーザー **(ITIL User)]**) の権限を表示して、[インシデント] テーブルの権限を表示する手順例を示します。

注: アクセスアナライザーは ServiceNow® Store 製品です。

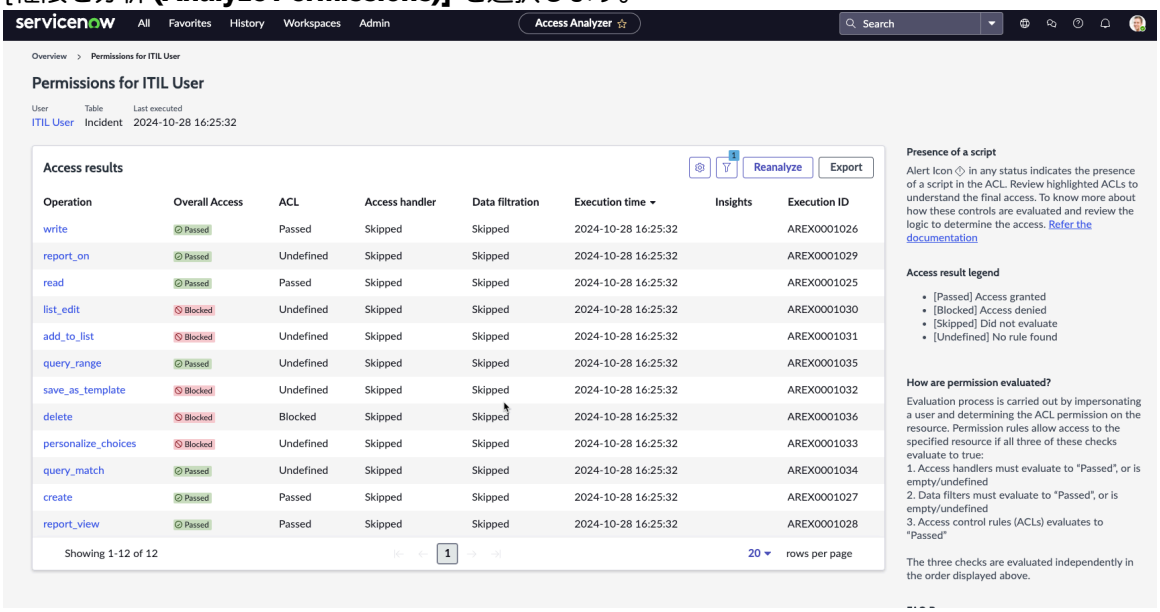
手順

1. 移動先 すべて > アクセスアナライザー > 権限を分析.
   [アクセスと権限の分析 (Analyze access and permissions)] ホームページが表示されます。
2. 次のように基準を選択します。

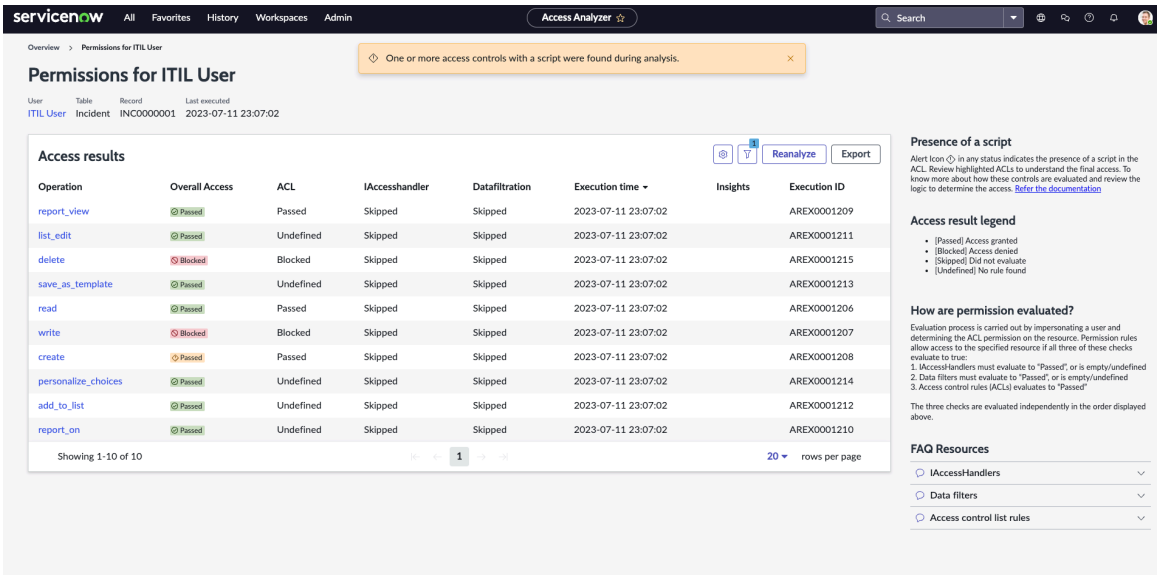
アクセスと権限を分析するための基準を選択します

フィールド	説明
分析者 (Analyze by) *	[ユーザー] を選択します。
ユーザーを選択 *	リストから選択するユーザー名を指定します。この例では、**[ITIL ユーザー (ITIL User)]** です。
ルールタイプ *	テーブル、**UI** ページ、**REST** エンドポイント、またはクライアント呼び出し可能スクリプトインクルードのアクセスを分析します。この例では、テーブルです。
テーブルを選択 *	リストから選択するテーブル名を指定します。この例では、[インシデント] です。
レコードを選択	リストから選択するレコード名を指定します。この例では、**INC0000001** です。
フィールドを選択	リストから選択するフィールド名を指定します。このフィールドは、フィールドレベルで権限を分析するためにも使用できます。たとえば、アクティブ、作成者などです。

3. [説明] フィールドに説明を指定します。
4. [権限を分析 **(Analyze Permissions)**] を選択します。



[ITIL ユーザー (ITIL User)] の [アクセス結果 **(Access results)**] が表示されます。



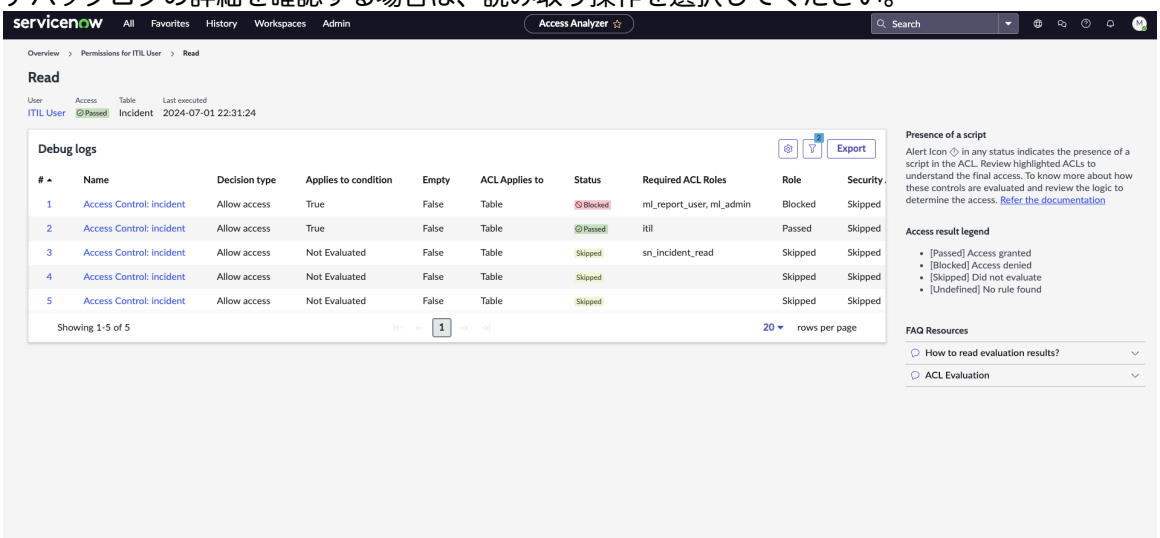
この結果は、凡例、アクセス制御リスト (ACL)、IAccesshandler、データフィルターを参照して読み取ることができます。

[読み取り] 操作の例を確認します。**[ITIL ユーザー (ITIL User)]** の場合、全体的なアクセス権は「合格」です。これは、ユーザーが正しい権限 (ACL) でレコードの読み取りができることを意味します。

同様に、[作成] 操作では、全体的なアクセス権は合格であり、アラートアイコンが表示されます。これは、ACL 評価用のスクリプトが存在する可能性があることを意味します。

> **注:** この例では、選択したユーザーの [書き込み] 操作と [削除] 操作がブロックされ、ユーザーは選択したレコード (INC0000001) を編集または削除できません。

5. デバッグログの詳細を確認する場合は、読み取り操作を選択してください。

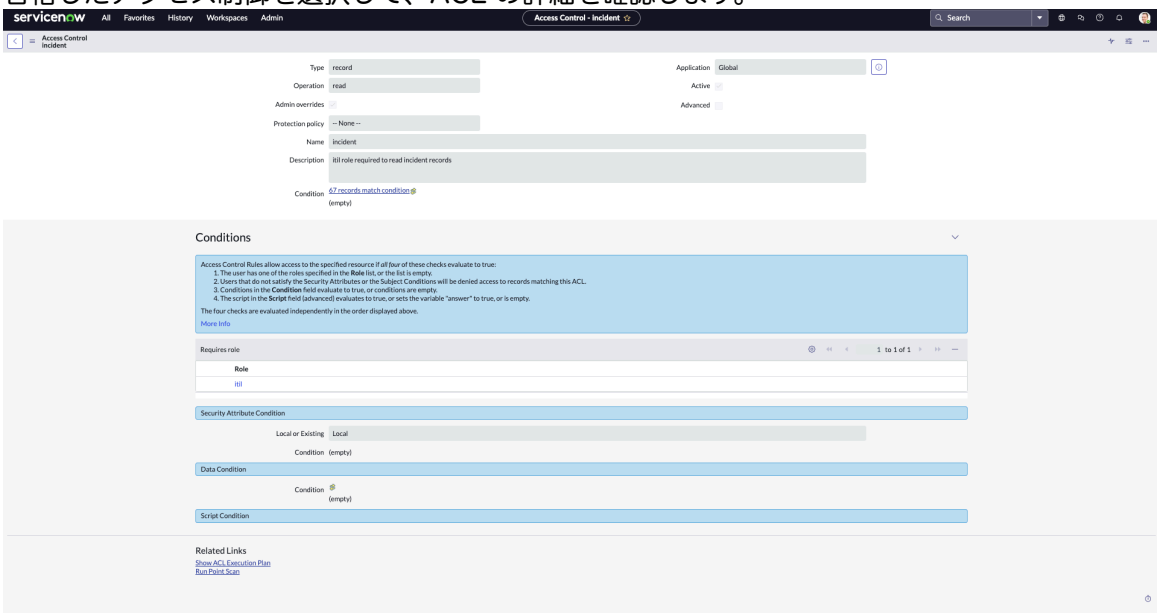


[デバッグログ] ページには、レコードの [読み取り] 操作を実行するために必要なビジネスルールと関連する ACL が表示されます。

デバッグログには、読み取り操作に関連付けられたビジネスルールと 4 つの ACL があることが示されています。

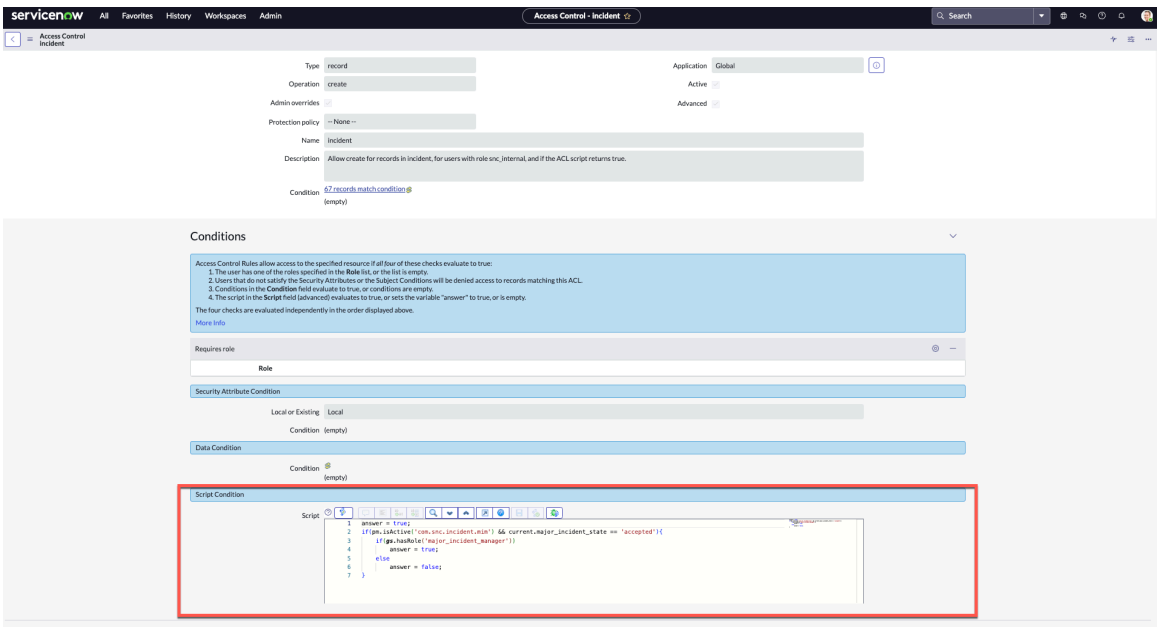
いずれかの ACL に [合格] ステータスが表示されています。これは、選択したレコードを読み取り、ユーザーに必要な ACL があり、レコードを読み取ることができることを意味しています。ACL の 1 つが [合格]であるため、他の ACL 評価は [スキップ済み]になります。

6. 合格したアクセス制御を選択して、ACL の詳細を確認します。



選択した ACL のアクセス制御の詳細が表示されます。

選択した操作は [合格] であり、スクリプトが存在します。[アクセス制御] ページには、レコードに関連するスクリプトが表示されます。



ロールの権限の表示

Access Analyzer を使用して選択したロールの権限を表示します。

始める前に
必要なロール:admin、access_analyzer_admin

次の例では、アクセスアナライザーの Evaluate Access を使用して、選択したロール (**user_admin**) の権限を表示して、REST API エンドポイントの権限を表示する手順例を示します。

注: アクセスアナライザーは ServiceNow® Store 製品です。

手順

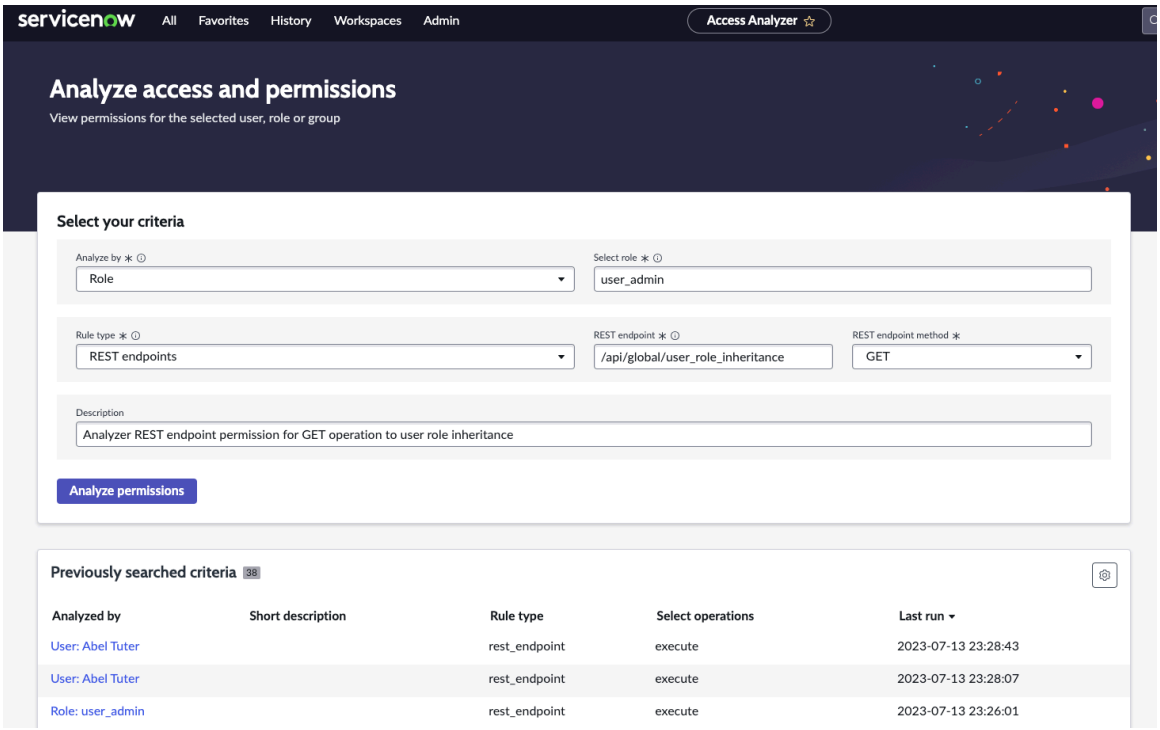
1. 移動先 すべて > アクセスアナライザー > 権限を分析.
   [アクセスと権限の分析 (Analyze access and permissions)] ホームページが表示されます。
2. 次のように基準を選択します。

アクセスと権限を分析するための基準を選択します

フィールド	説明
分析者 (Analyze by) *	[ロール] を選択します。
ユーザーを選択 *	リストから選択するロールを指定します。 例：**user_admin**
ルールタイプ *	テーブル、UI ページ、REST エンドポイント、またはクライアント呼び出し可能スクリプトインクルードのアクセスを分析します。 例：**REST** エンドポイント **(REST endpoint)**
REST エンドポイント (REST endpoint)*	REST エンドポイントを指定します。 例：/api/global/user_role_inheritance 注: 選択したフィールドを使用する場合は、完全な REST エンドポイントを使用する必要があります。
REST エンドポイントメソッド (REST endpoint method) *	REST エンドポイントメソッドを指定します。 例：GET

3. [説明] フィールドに説明を指定します。
4. [権限を分析 **(Analyze Permissions)**] を選択します。

自動翻訳



user_admin ロールの [アクセス結果 **(Access results)]** が表示されます。

この結果は、凡例、アクセス制御リスト (ACL)、IAccesshandler、データフィルターを参照して読み取ることができます。

ロールの全体的なアクセス権が合格します。これは、ロール (**user_admin**) が、選択した **[GET]** メソッドの **[REST** エンドポイント **(REST endpoint)]** にアクセスできることを意味します。

グループの権限の表示

アクセスアナライザーを使用して選択したグループの権限を表示します。

始める前に

必要なロール:admin、access_analyzer_admin

次の例では、アクセスアナライザーの Evaluate access を使用して、選択したグループ ([インシデント管理]) の権限を表示して、[インシデント] UI ページの権限を表示する手順例を示します。

注: アクセスアナライザーは ServiceNow® Store 製品です。

手順

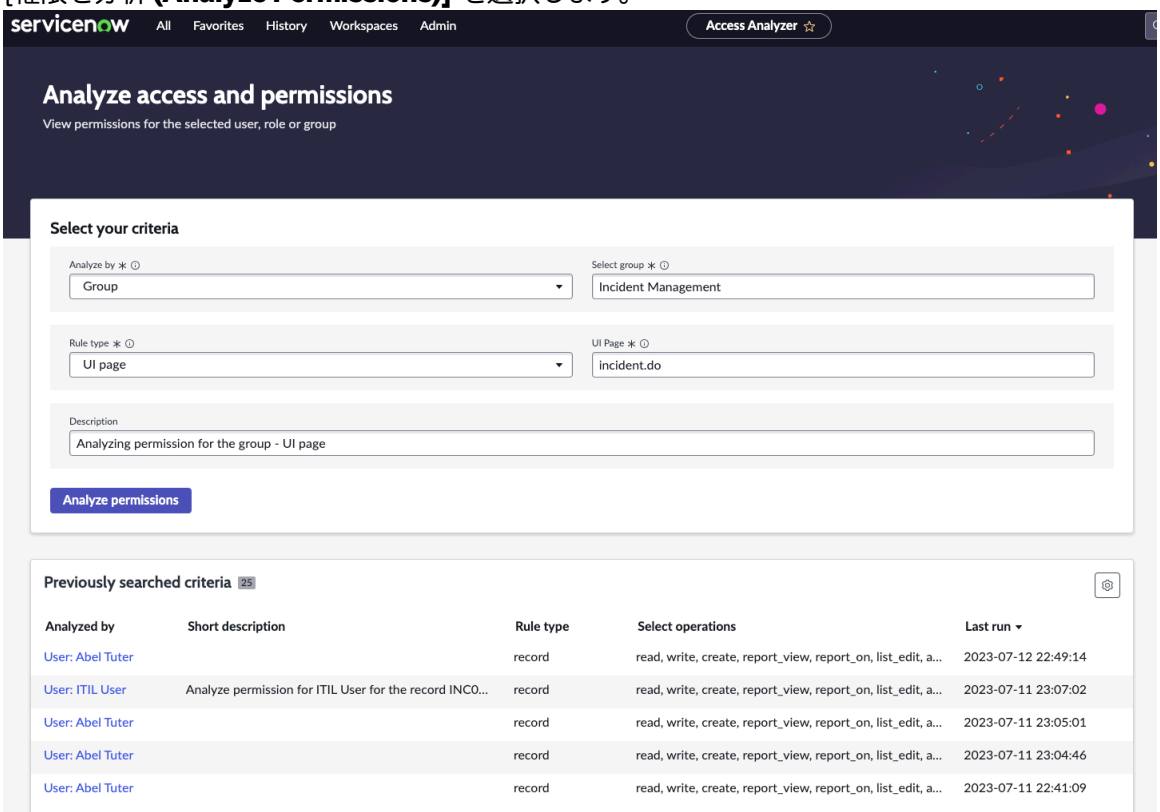
1. 移動先 すべて > アクセスアナライザー > 権限を分析.
   [アクセスと権限の分析 (Analyze access and permissions)] ホームページが表示されます。
2. 次のように基準を選択します。

アクセスと権限を分析するための基準を選択します

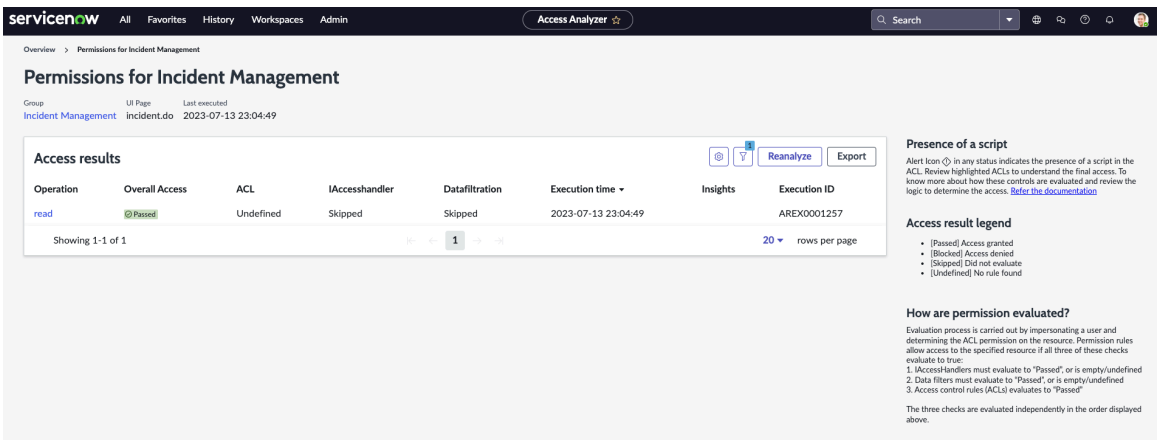
フィールド	説明
分析者 (Analyze by) *	[グループ] を選択します。
ユーザーを選択 *	リストから選択するユーザー名を指定します。 例：インシデント管理

フィールド	説明
ルールタイプ *	テーブル、UI ページ、REST エンドポイント、またはクライアント呼び出し可能スクリプトインクルードのアクセスを分析します。例：**UI ページ**
UI ページ*	UI ページを指定します。例：**incident.do**

3. [説明] フィールドに説明を指定します。
4. [権限を分析 **(Analyze Permissions)**] を選択します。



[インシデント管理] グループの [アクセス結果 **(Access results)**] が表示されます。



この結果は、凡例、アクセス制御リスト (ACL)、IAccesshandler、データフィルターを参照して読み取ることができます。

グループの全体的なアクセス権は合格です。これは、グループ内のユーザー (**Incident Management**) がインシデントレコードにアクセスできることを意味します。

Access Analyzer クエリのエクスポート

アクセスアナライザーを使用して分析されたクエリをエクスポートします。

始める前に
必要なロール:admin、access_analyzer_admin

次の手順では、アクセスアナライザーにアクセスし、アクセスアナライザー内のさまざまな機能を使用するステップについて説明します。

注: アクセスアナライザーは ServiceNow® Store 製品です。

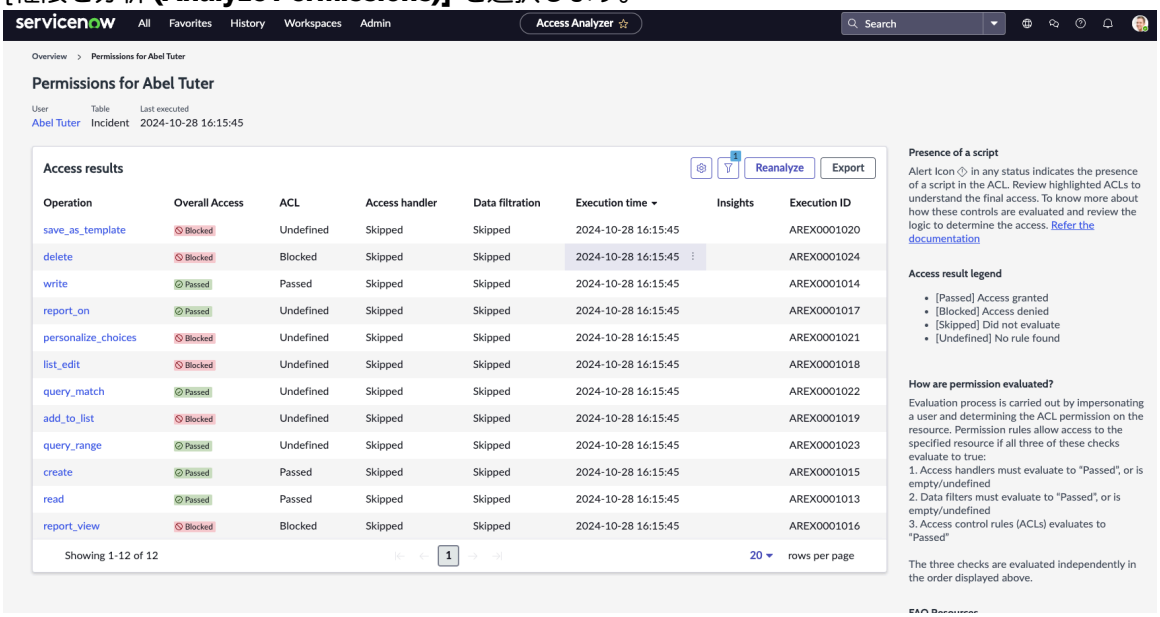
手順

1. 移動先 すべて > アクセスアナライザー > 権限を分析.
   [アクセスと権限の分析 (Analyze access and permissions)] ホームページが表示されます。
2. 次のように基準を選択します。

アクセスと権限を分析するための基準を選択します

フィールド	説明
分析者 (Analyze by) *	ユーザー、ロール、グループのアクセスを分析します。
ユーザーを選択 *	リストから選択するユーザー名を指定します。
ルールタイプ *	テーブル、UI ページ、REST エンドポイント、またはクライアント呼び出し可能スクリプトインクルードのアクセスを分析します。
テーブルを選択 *	リストから選択するテーブル名を指定します。
レコードを選択	リストから選択するレコード名を指定します。
フィールドを選択	リストから選択するフィールド名を指定します。

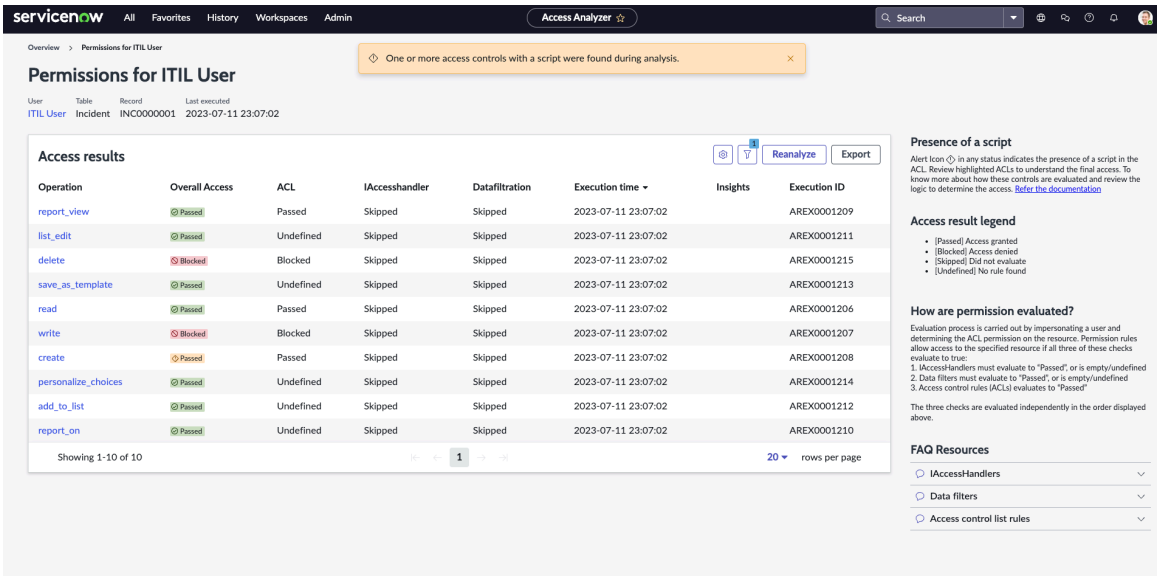
3. [説明] フィールドに説明を指定します。
4. [権限を分析 **(Analyze Permissions)**] を選択します。



ユーザーのアクセス結果が表示されます。同様に、次のルールタイプについて、グループ、ロールの権限を分析できます。

- テーブル (レコード)
- クライアント呼び出し可能スクリプトインクルード
- REST エンドポイント

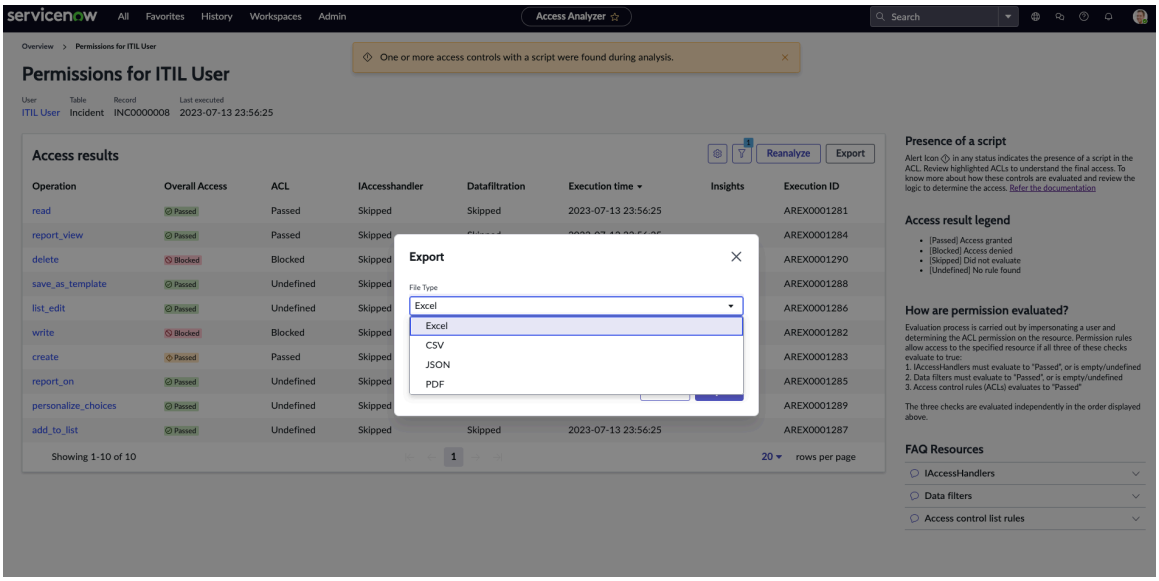
選択したルールタイプの [アクセス結果 **(Access results)]** が表示されます。



5. [エクスポート] をクリックします。

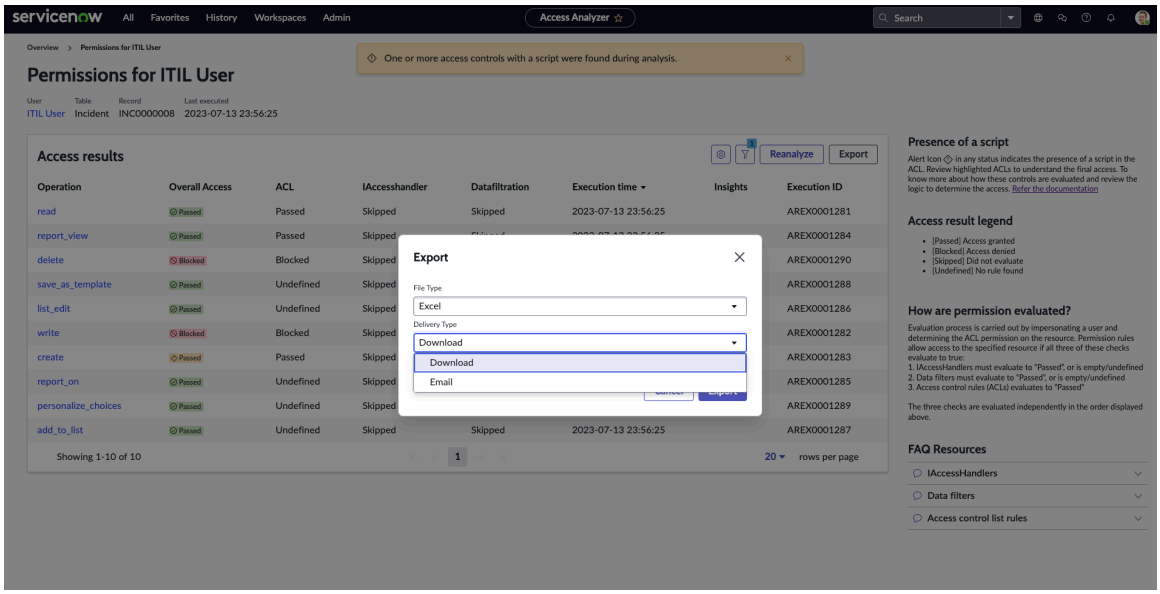
a. ファイルタイプを選択します。

利用可能なファイルタイプは、Excel、CSV、JSON、PDF です。



b. [配送タイプ] を選択します。

利用可能な配送タイプは、ダウンロードとメールです。



ユーザーレコードの比較

ユーザーレコードを比較して、2 人のユーザー間のアクセスを把握します。

始める前に

必要なロール:admin、access_analyzer_admin

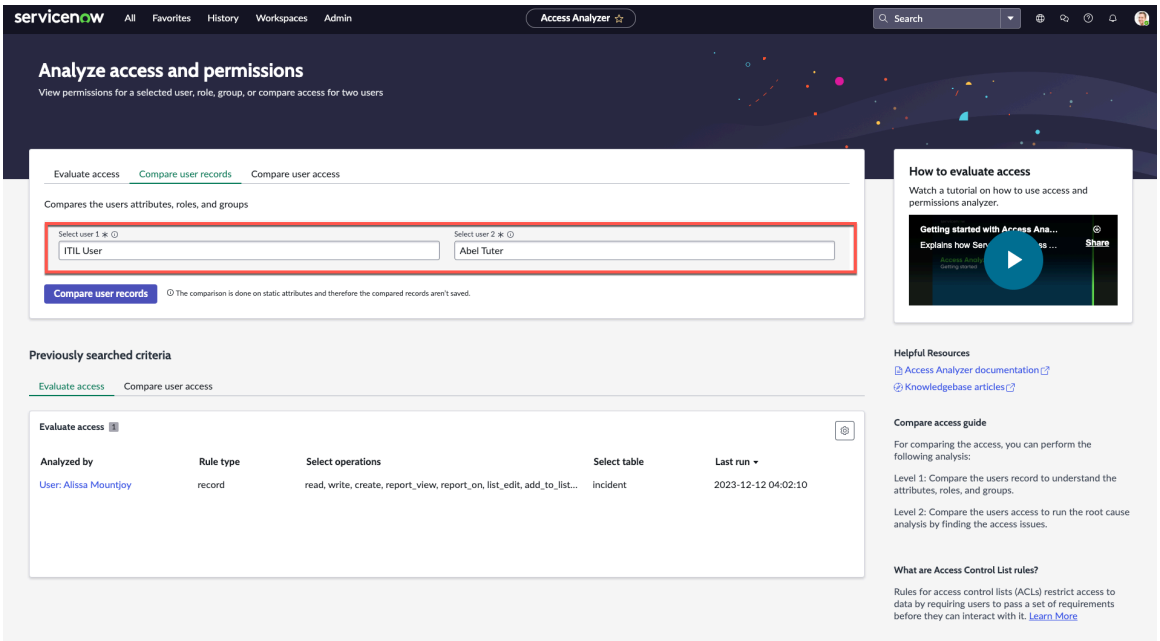
以下の手順では、アクセスアナライザーを使用してユーザーレコードを比較する方法について説明します。

注: アクセスアナライザーは ServiceNow® Store 製品です。

手順

1. 移動先 すべて > アクセスアナライザー > 権限を分析.
   [アクセスと権限の分析 (Analyze access and permissions)] ホームページが表示されます。
2. [ユーザーレコードの比較 **(Compare user records)]** タブを選択します。
3. ユーザー **1** とユーザー **2** を選択して比較します。

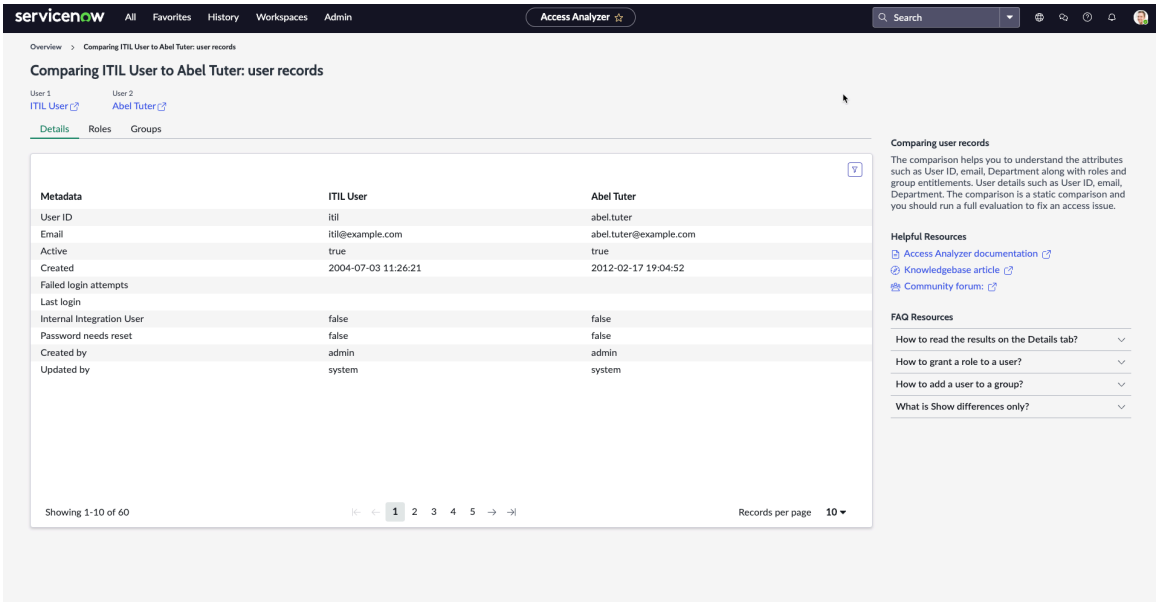
   たとえば、ITIL User をユーザー **1**、Abel Tuter をユーザー **2** とします。



4. [ユーザーレコードの比較 **(Compare user records)]** を選択します。

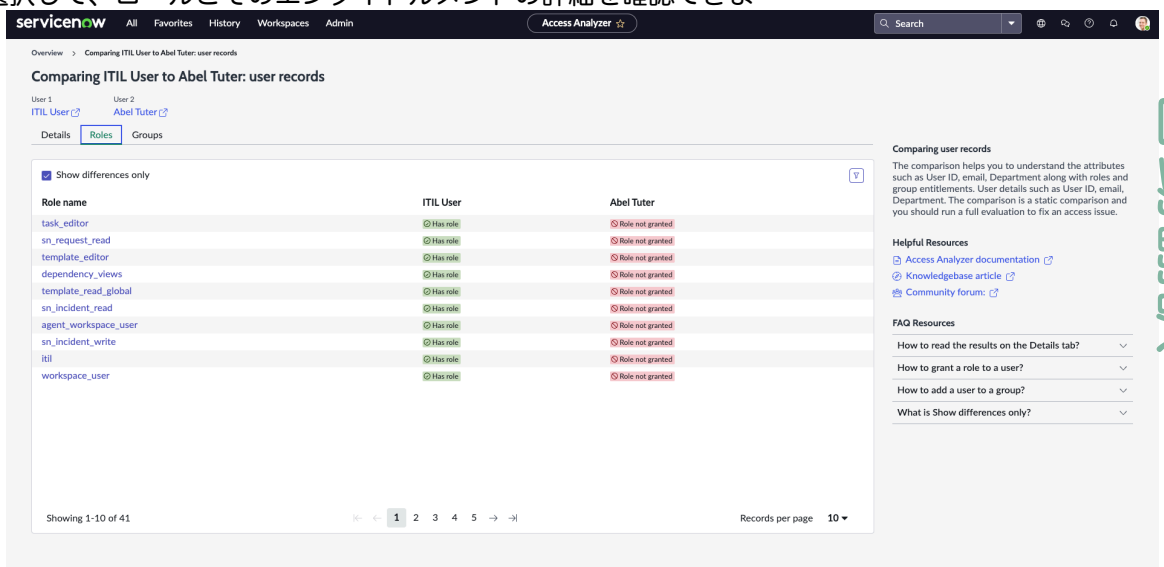
   結果は次のタブに表示されます。

- 詳細：ユーザーのメタデータを表示しま



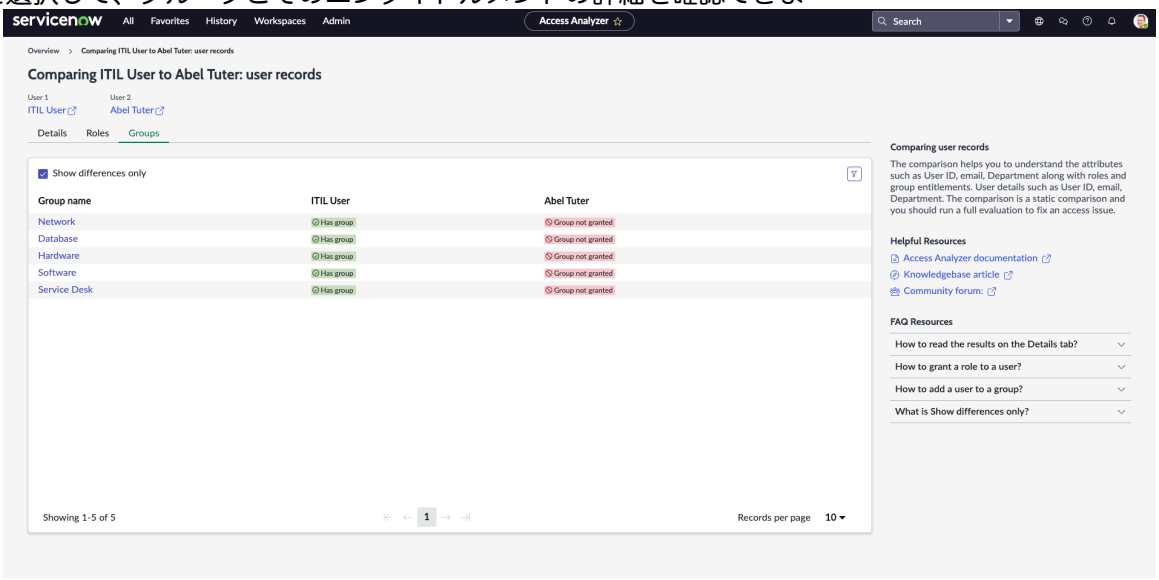
す。

- ロール：ユーザーに割り当てられているロールを表示します。ロールを選択して、ロールとそのエンタイトルメントの詳細を確認できま



す。

- グループ：ユーザーに割り当てられているグループを表示します。グループを選択して、グループとそのエンタイトルメントの詳細を確認できます。



同様に、ServiceNow インスタンス内のさまざまなユーザーを比較して、ユーザーに割り当てられているアクセス権を把握できます。

ユーザーアクセスの比較

アクセスアナライザーを使用して、ユーザーのアクセス制御を比較します。

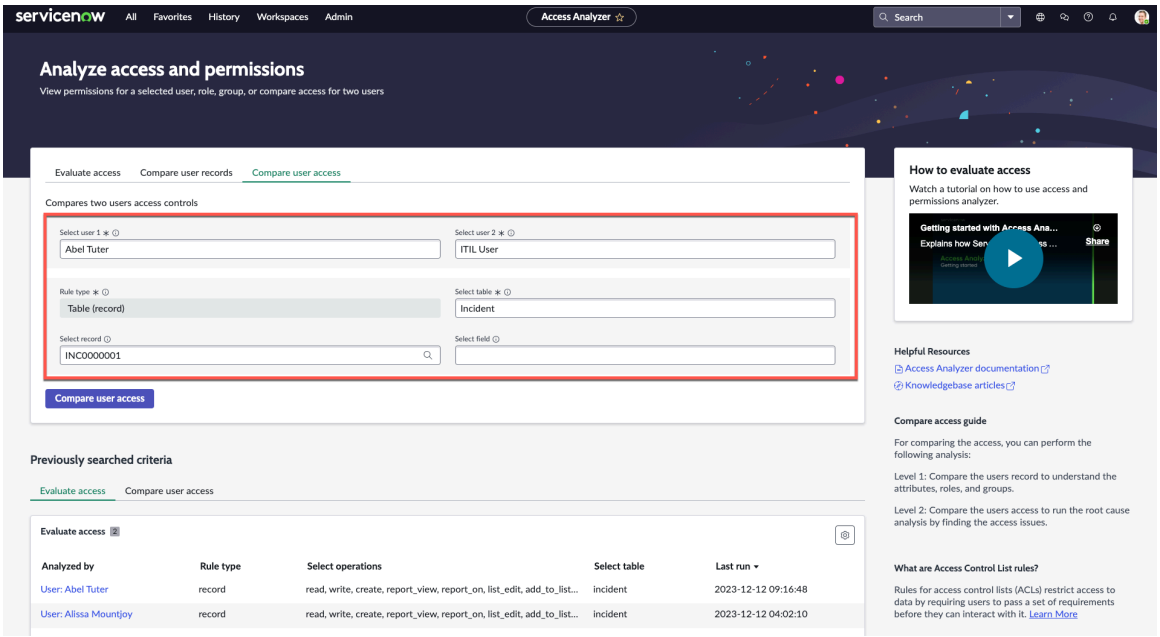
始める前に
必要なロール:admin、access_analyzer_admin

以下の手順では、アクセスアナライザーを使用してユーザー間でアクセス制御を比較する方法について説明します。

注: アクセスアナライザーは ServiceNow® Store 製品です。

手順

1. 移動先 すべて > アクセスアナライザー > 権限を分析.
   [アクセスと権限の分析 (Analyze access and permissions)] ホームページが表示されます。
2. [ユーザーアクセスの比較 **(Compare user access)]** タブを選択します。
3. 次のフィールドに入力します。



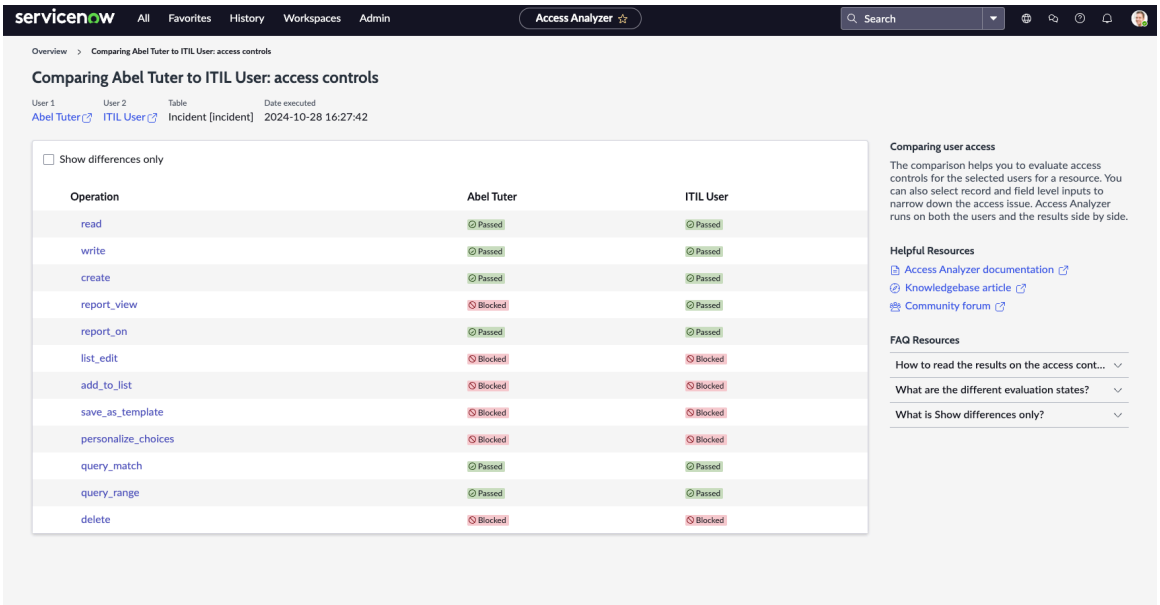
ユーザーアクセスの比較

フィールド	説明
ユーザー 1* を選択	リストから選択するユーザー名を指定して比較します。
ユーザー 2* を選択	リストから選択するユーザー名を指定してユーザー 1 と比較します。
ルールタイプ*	テーブルのアクセス許可を分析します。 **注:** ユーザーアクセスの比較で使用できるのは、テーブルのアクセス権限のみです。
テーブルを選択*	リストから選択するテーブル名を指定します。
レコードを選択	リストから選択するレコード名を指定します。
フィールドを選択	リストから選択するフィールド名を指定します。

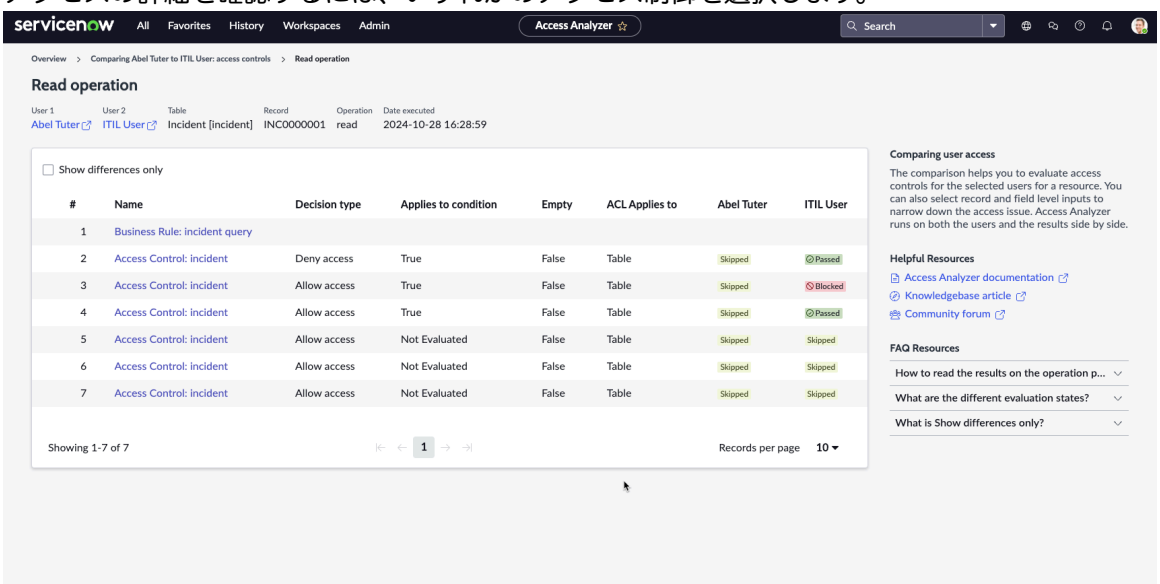
4. [ユーザーアクセスの比較 **(Compare user access)**] を選択します。

   選択されたユーザーのユーザーアクセスの比較結果が表示されます。

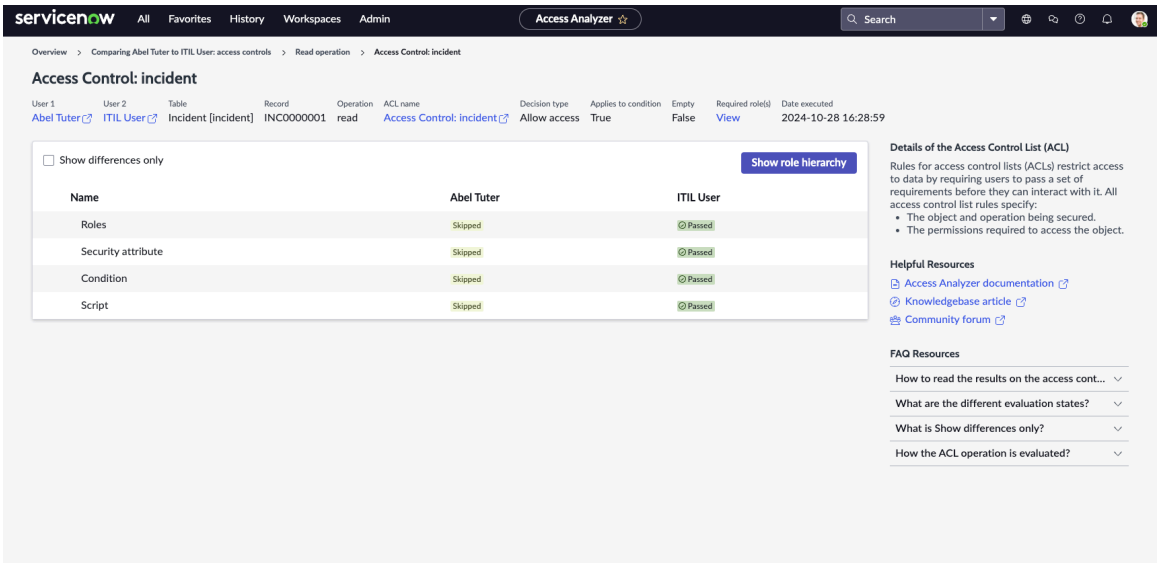
   ユーザーアクセスの比較結果には、ユーザーの操作とアクセス評価ステータスが表示されます。たとえば、Abel Tuter、ITIL User などです。



5. [操作] を選択して、権限の評価とユーザーに割り当てられているロールの詳細を確認します。たとえば、読み取り操作です。
6. アクセスの詳細を確認するには、いずれかのアクセス制御を選択します。

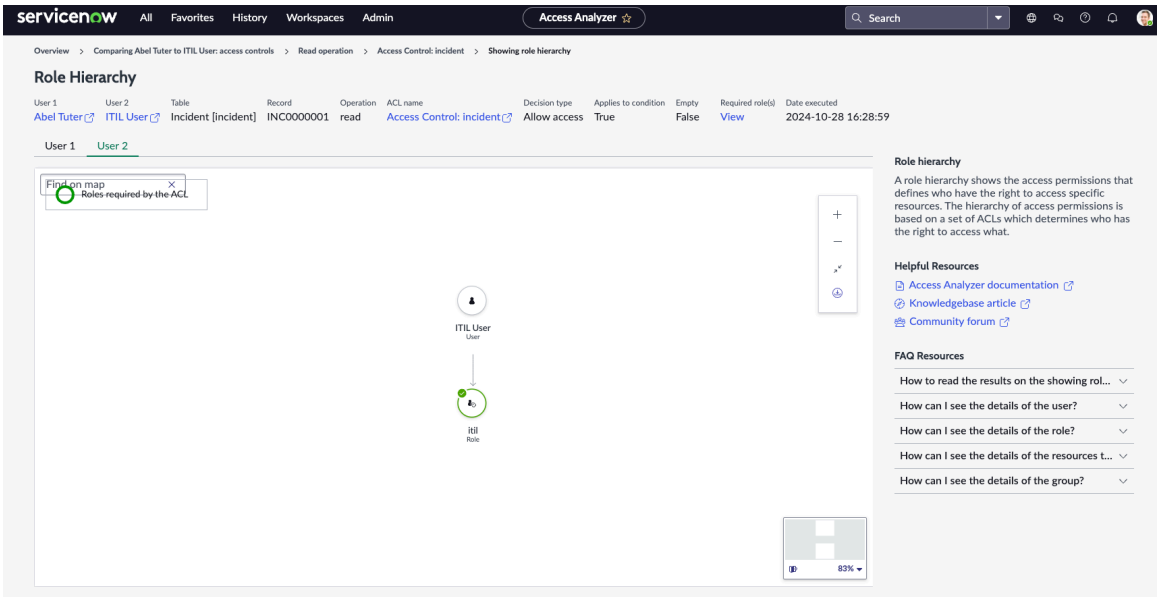


ロール、セキュリティ属性、条件、スクリプト評価ステータスなどのアクセス制御の詳細が表示されます。



7. [ロール階層の表示 **(Show role Hierarchy)]** を選択して、ユーザーに割り当てられているロールとグループの詳細を確認し、両方のユーザーを比較します。

   ロール階層に基づいて、リソース (テーブル) にアクセスするために必要なロールとグループのアサインをユーザーにアサインできます。



この例では、 **Abel Tuter** には itil がアサインされていません。ロール階層を調べることで、ユーザーに必要なロールとグループの割り当てを決定できます。

ノードを選択すると、ロール、ロールがアクセスできるリソース、またはグループの詳細を確認できます。

アクセスアナライザーで以前に検索した条件の表示

アクセスアナライザーで以前に検索した条件を表示します。

始める前に
必要なロール:admin、access_analyzer_admin

注: アクセスアナライザーは ServiceNow® Store 製品です。

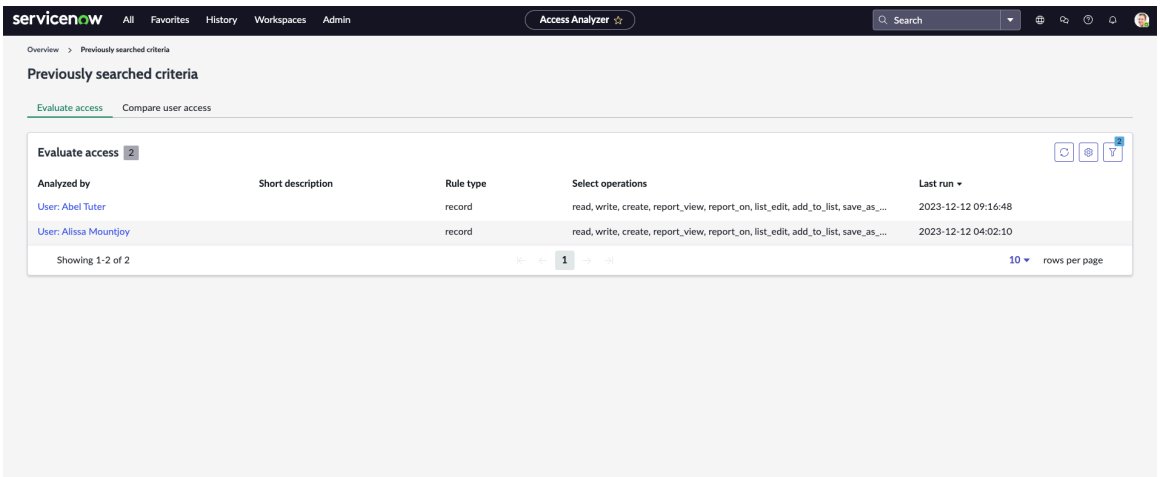
手順

1. 移動先 すべて > アクセスアナライザー > アクセスアナライザークエリ.

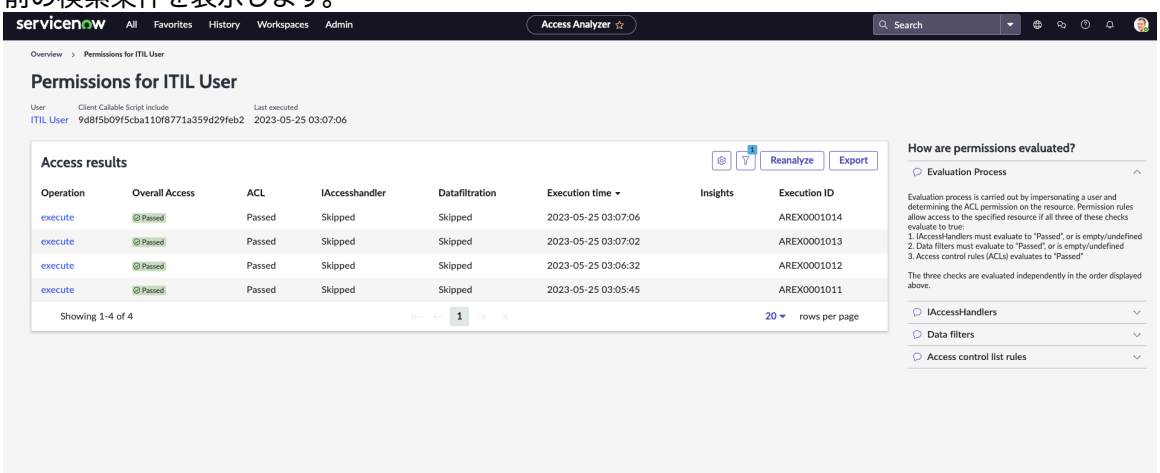
   以前に検索された条件には、次のセクションがあります。

   - アクセスの評価：アクセスの評価機能によって実行されたクエリに基づいて結果を表示します。
   - ユーザーアクセスの比較：ユーザーアクセスの比較機能によって実行されたクエリに基づいて結果を表示します。

   注: ユーザーレコードの比較機能を使用する場合、以前に検索した条件は保存されません。

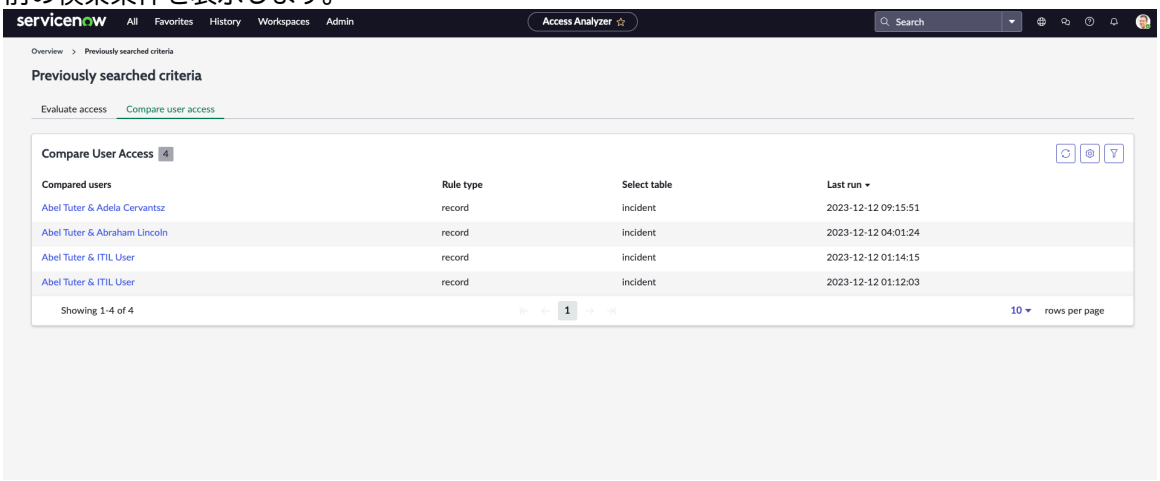


2. [分析者 **(Analyzed by)]** リンクを選択して、[アクセスを評価 (Ecaluate access)] セクションで以前の検索条件を表示します。



ユーザーのアクセスについて [再分析する] を選択できます。[エクスポート] オプションを使用して詳細をエクスポートします。

3. [ユーザーの比較 **(Compared users)**] リンクを選択して、ユーザーアクセスの比較セクションで以前の検索条件を表示します。



権限の評価

Access Analyzer を使用する場合の権限の評価基準です。

評価階層

選択したユーザー、グループ、ロールの権限は、次の階層で評価されます。

- ビジネスルール：ビジネスルールとは、レコードが表示、挿入、更新、または削除されるとき、またはテーブルに対してクエリが実行されるときに実行する、サーバー側スクリプトです。
- アクセスハンドラー：プラットフォーム上の非表示のソースコードを使用する内部システムチェックです。
- データフィルタリング：データフィルターは、インスタンスの既存のアクセス制御ルール (ACL) と連携して機能するように設計されたアクセス制御の形式です。データフィルターでは読み取り操作のみがサポートされます。
- アクセス制御リスト (ACL)：アクセス制御リスト (ACL) のルールは、まず要件のセットをユーザーに要求し、その後でユーザーとやり取りできるようにすることで、データへのアクセスを制限します。ACL 内では、次の階層が評価されます。
  - ロール
  - セキュリティ属性
  - 条件
  - スクリプト

Access Analyzer を使用して、選択したユーザー、ロール、またはグループのアクセス権と権限を分析できます。権限は、次のルールタイプに基づいて評価されます。

- テーブルレベルの評価 **(Table Level Evaluation)**：テーブルレベルの評価には、ロールとセキュリティ属性の ACL が使用されます。
- レコードまたはフィールドレベルの評価 **(Record or Field level Evaluation)**：レコードまたはフィールドレベルの評価には、ロール、セキュリティ属性、条件、およびスクリプトレベルの ACL が使用されます。
- **UI** ページ：準備完了操作のみをサポートします。読み取りレベルの ACL のみが評価されます。
- **REST** エンドポイント **(REST Endpoint)**：実行操作のみをサポートします。実行レベルの ACL のみが評価されます。

アクセス結果の重要なフィールドの詳細は次のとおりです。

- スクリプトの存在
- アクセス結果の凡例
- 評価プロセス
- IAccessHandler
- データフィルター
- アクセス制御リストのルール

スクリプトの存在

ステータスのアラートアイコンは ACL にスクリプトが存在することを示します。ハイライト表示された ACL を確認して、最終的なアクセス権を把握します。これらの制御を評価する方法の詳細を把握し、アクセスを決定するロジックを確認するには、「Access Analyzer のデバッグログ」を参照してください。

Access Analyzer の凡例

アクセスと権限を分析する際、評価プロセスの一部として凡例が表示されます。凡例は次のとおりです。

- [合格] アクセスが許可されました
- [ブロック済み] アクセスが拒否されました
- [スキップ] 評価しませんでした
- [未定義] ルールが見つかりませんでした

評価プロセス

評価プロセスは、ユーザーの代理操作を行い、リソースに対するアクセス制御リスト (ACL) 権限を決定することによって実行されます。次のチェックが true と評価された場合、権限ルールにより、指定されたリソースへのアクセスが有効になります。

- IAccessHandler は「合格」と評価されるか、空または未定義である必要がある
- データフィルターは「合格」と評価されるか、空または未定義である必要がある
- アクセス制御ルール (ACL) が「合格」と評価される

IAccessHandler

プラットフォーム上の非表示のソースコードを使用する内部システムチェック。IAccessHandler は、ACL を評価せずにリソースへのアクセスを許可または拒否できます。IAccessHandler が無視されると、ACL が評価されます。

IAccessHandler チェックは変更できません。たとえば、IAccessHandler 実装は読み取りアクセスなどのアプリケーションリソースのアクセスチェックに使用されます。

データフィルター

データフィルターは、インスタンスの既存のアクセス制御ルール (ACL) と連携して機能するように設計されたアクセス制御の形式です。

アクセス制御リストのルール

アクセス制御リスト (ACL) のルールは、まず要件のセットをユーザーに要求し、その後でユーザーとやり取りできるようにすることで、データへのアクセスを制限します。

よく寄せられる質問

アクセスアナライザーの使用時によく寄せられる質問。

Evaluate Access

アクセスアナライザーの Evaluate Access 機能の使用時によく寄せられる一部の質問は次のとおりです。

よく寄せられる質問

質問	説明
アクセスアナライザーによって表示される評価結果の読み方を教えてください。	各行は個々のアクセス制御リスト (ACL) を表します。結果のシーケンス (#) は ACL が評価される順序です。ステータスは、全体的なアクセスが許可 (合格) されるのか、拒否 (ブロック) されるのかを示します。
ACL はどのように評価されますか？	テーブルレベルでは、ACL はロールとセキュリティ属性に対してのみ評価され、条件とスクリプトは評価されません。 ロールが最初に評価されます。ロールがブロックされている場合、条件とスクリプトはスキップされます。詳細については、「ACL ルールの構成」を参照してください。
アクセスアナライザーにはどのような凡例がありますか？	アクセスと権限を分析する際、評価プロセスの一部として凡例が表示されます。凡例は次のとおりです。 • [合格] アクセスが許可されました • [ブロック済み] アクセスが拒否されました • [スキップ済み] 評価されませんでした • [未定義] ルールが見つかりませんでした
アクセス結果のアラートアイコンは何を意味しますか？	ステータスのアラートアイコンは ACL にスクリプトが存在することを示します。ハイライト表示された ACL を確認して、最終的なアクセス権を把握します。これらの制御を評価する方法の詳細を把握し、アクセスを決定するロジックを確認するには、「Access Analyzer のデバッグログ」を参照してください。

よく寄せられる質問 (続く)

質問	説明
IAccesshandler とは何ですか？	プラットフォーム上の非表示のソースコードを使用する内部システムチェック。これは変更できないシステムセキュリティチェックです。**IAccessHandler** は、ACL を評価せずにリソースへのアクセスを許可または拒否できます。 **IAccessHandler** が無視されると、ACL が評価されます。どのような方法でも **IAccessHandler** チェックは変更できません。たとえば、**IAccessHandler** 実装は読み取り専用アクセスなどのアプリケーションリソースのアクセスチェックに使用されます。
データフィルターとは何ですか？	データフィルターは、インスタンスの既存のアクセス制御ルール (ACL) と連携して機能するように設計されたアクセス制御の形式です。
ACL ルールとは何ですか？	アクセス制御リスト (ACL) のルールは、ユーザーがデータを操作する前に、まず要件のセットをユーザーに要求することで、データへのアクセスを制限します。

結果に影響する可能性がある、ユーザーに対する時間制限付きロールの割り当てが見つかりました。ユーザーに割り当てられている時間制限付きロールは、ここで確認できます。

ユーザーレコードの比較

アクセスアナライザーのユーザーレコードの比較機能の使用時によく寄せられる一部の質問は次のとおりです。

よく寄せられる質問

質問	説明
[詳細] タブで結果を読むにはどうすればよいですか？	[詳細] タブには、ユーザー 1 とユーザー 2 に関連付けられたメタデータが表示されます
ユーザーにロールを付与するには？	[ユーザー] タブで、ユーザーに付与する必要があるロールを確認し、そのロールを割り当てることができます。
グループにユーザーを追加するには？	[グループ] タブで、ユーザーを追加する必要があるグループを確認し、ユーザーをグループに追加できます。
[差異のみを表示 (Show difference only)] とは何ですか？	[差異のみを表示 (Show difference only)] チェックボックスをオンにすると、ユーザー 1 とユーザー 2 の間で異なるロールまたはグループのみが表示されます。

ユーザーアクセスの比較

アクセスアナライザーのユーザーアクセスの比較機能の使用時によく寄せられる一部の質問は次のとおりです。

よく寄せられる質問

質問	説明
アクセス制御比較ページで結果を読む方法は？	アクセス制御比較ページには、さまざまな ACL 操作の評価状況が表示されます。
評価状況にはどのようなものがありますか？	ユーザー間でアクセス制御を比較する場合のさまざまな評価状態は次のとおりです。 • 合格 • ブロック済み
[差異のみを表示 (Show difference only)] とは何ですか？	[差異のみを表示 (Show difference only)] チェックボックスをオンにすると、ユーザー 1 とユーザー 2 の間で異なる操作の評価状態のみが表示されます。
ACL 操作はどのように評価されますか？	アクセス制御リスト (ACL) のルールは、まず要件のセットをユーザーに要求し、その後でユーザーとやり取りできるようにすることで、データへのアクセスを制限します。ACL 内では、次の階層が評価されます。 • ロール • セキュリティ属性 • 条件 • スクリプト
表示されているロール階層ページで結果を読むには？	表示されているロール階層ページには、ユーザー 1 とユーザー 2 に割り当てられているロールが表示されます。特定の ACL 操作でユーザーに必要なロールを把握できます。
ユーザーの詳細を確認するにはどうすればよいですか？	選択できます ユーザー (ノード) > その他のアクション > ユーザーを表示 ユーザーの詳細を確認します。
ロールの詳細を確認するにはどうすればよいですか？	選択できます ロール (ノード) > その他のアクション > 表示ロール ロールの詳細を確認します。
ロールでアクセスできるリソースの詳細を確認するには？	選択できます ロール (ノード) > その他のアクション > ロールがアクセスできるすべてのリソースを表示する ロールがアクセスできるリソースを確認します。

よく寄せられる質問 (続く)

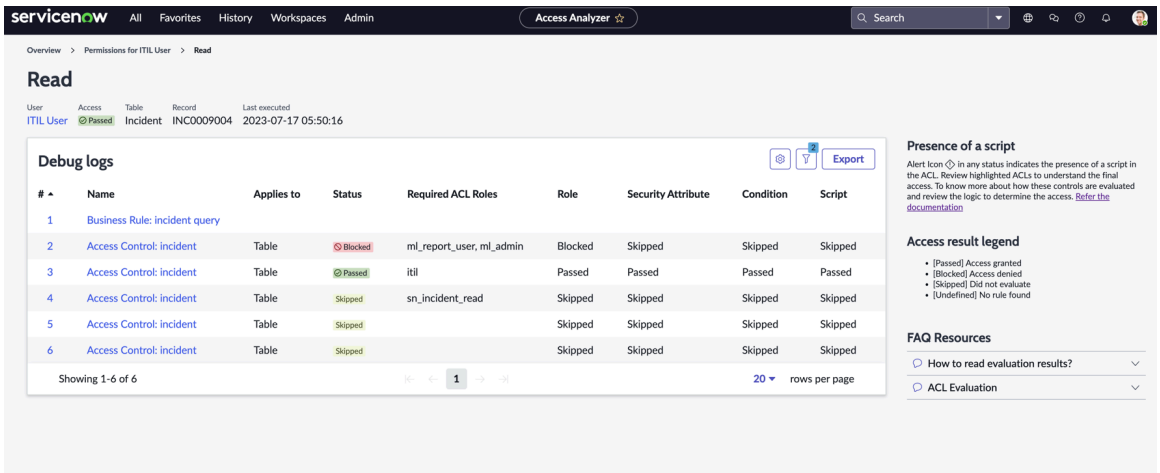
質問	説明
グループの詳細を確認するにはどうすればよいですか？	選択できます グループ (ノード) > その他のアクション > グループを表示 グループの詳細を確認します。

Access Analyzer のデバッグログ

デバッグログには、アクセス結果の選択操作の詳細が表示されます。

デバッグログのフィールド

アクセスアナライザーのデバッグログには、選択した操作に関する情報が表示され、操作に関連付けられた権限、ビジネスルール、および ACL を把握できます。



デバッグログのフィールドとその説明は次のとおりです。

デバッグログ

フィールド	説明
名前	ビジネスルールまたは ACL に関する詳細。ACL のビジネスルールを選択すると、詳細を確認できます。
適用先	フィールド、レコード、またはテーブルレベルでの ACL の適用に関する詳細
ステータス	関連するロールと権限の ACL のステータス
ACL が必要 (Requires ACL)	フィールド、レコード、またはテーブルへのアクセスに必要なロール
ロール	アクセス制御に対するブロック済み、合格、スキップ済みのロールに関する詳細
セキュリティ属性	アクセス制御に対するブロック済み、合格、スキップ済みのセキュリティ属性に関する詳細
条件	アクセス制御に対するブロック済み、合格、スキップ済みの条件に関する詳細

デバッグログ (続く)

フィールド	説明
スクリプト	アクセス制御に対するブロック済み、合格、スキップ済みのスクリプトに関する詳細
カスタマイズ済み	アクセス制御に対してカスタマイズされた ACL がある場合はその詳細
アプリケーション	アプリケーションのステータス。グローバルまたはストア

評価階層

選択したユーザー、グループ、ロールの権限は、次の階層で評価されます。

- ビジネスルール：ビジネスルールとは、レコードが表示、挿入、更新、または削除されるとき、またはテーブルに対してクエリが実行されるときに実行する、サーバー側スクリプトです。
- アクセスハンドラー：プラットフォーム上の非表示のソースコードを使用する内部システムチェックです。
- データフィルタリング：データフィルターは、インスタンスの既存のアクセス制御ルール (ACL) と連携して機能するように設計されたアクセス制御の形式です。データフィルターでは読み取り操作のみがサポートされます。
- アクセス制御リスト (ACL)：アクセス制御リスト (ACL) のルールは、まず要件のセットをユーザーに要求し、その後でユーザーとやり取りできるようにすることで、データへのアクセスを制限します。ACL 内では、次の階層が評価されます。
  - ロール
  - セキュリティ属性
  - 条件
  - スクリプト

アクセス制御リストの評価

操作の ACL は、次の順序で評価されます。

- ロール
- セキュリティ属性
- 条件
- スクリプト

スクリプトの存在

ステータスのアラートアイコンは ACL にスクリプトが存在することを示します。ハイライト表示された ACL を確認して、最終的なアクセス権を把握します。

注: Access Analyzer のクエリでは、ビジネスルールが最初に実行され、次にアクセス制御リストが実行されます。

実行順序

さまざまなシナリオにおけるアクセス結果の実行順序は次のとおりです。

- 継承された **ACL** またはワイルドカード **ACL** の存在：実行順序では、継承された ACL が最初に評価され、次にワイルドカード ACL が評価されます。
- **1** つの **ACL** が合格すると他の **ACL** はスキップされる：権限の実行と評価では、1 つの ACL が合格すると他の ACL の実行と評価はスキップされます。ID のフィールド、レコード、またはテーブルにアクセスするには、選択した操作の全体的な権限に 1 つの ACL が必要であるためです。
- フィールドレベルの **ACL** とテーブルレベルの **ACL** の実行：実行では、ID のアクセスを分析するときに詳細な結果を提供するため、フィールドレベルの ACL が最初に実行され、次にテーブルレベルの ACL が実行されます。
- スクリプト化された **ACL** が存在する場合の評価：スクリプトが存在する場合、ACL 内のスクリプトを示すアラートアイコンとともに操作の全体的なアクセスが合格します。

アクセスシミュレーター

アクセスシミュレーターでは、ロールやグループがユーザーにアサインされたり削除されたりした場合に、指定したテーブルへのアクセスがどのように変化するかをシミュレートできます。

探索	使用方法
	
アクセスシミュレーターの機能とビジネス価値について説明します。	アクセスシミュレーターの使用方法について説明します。

自動翻訳

構成

アクセスシミュレーターの構成方法について説明します。

よく寄せられる質問

アクセスシミュレーターに関するよくある質問の詳細を確認します。

アクセスシミュレーターの詳細

アクセスシミュレーターでは、ロールやグループがユーザーにアサインされたり削除されたりした場合に、指定したテーブルへのアクセスがどのように変化するかをシミュレートできます。

アドミニストレーターは、アクセスシミュレーターを使用して、指定したテーブルでのユーザーのアクセス要件をシミュレートし、ユーザーのアクセス制御がどのように変化するかを理解できます。これは、ユーザーの現在のアクセス権をすばやく把握し、ユーザーにロールを提供または削除した場合の影響を理解するのに役立ちます。

https://player.vimeo.com/video/988627830?badge=0&autopause=0&player_id=0&app_id=58479

アクセスシミュレーターを使用して、次のようなシナリオでユーザーのアクセス権をシミュレートできます。

- ユーザーへのロールの追加
- ユーザーからのロールの削除
- グループへのユーザーの追加
- グループからのユーザーの削除

アクセスシミュレーターの設定の詳細と「アクションの実行」を有効にする方法については、「アクセスシミュレーターの構成 (アクションの実行)」を参照してください。

アクセスシミュレーターの構成 (アクションの実行)

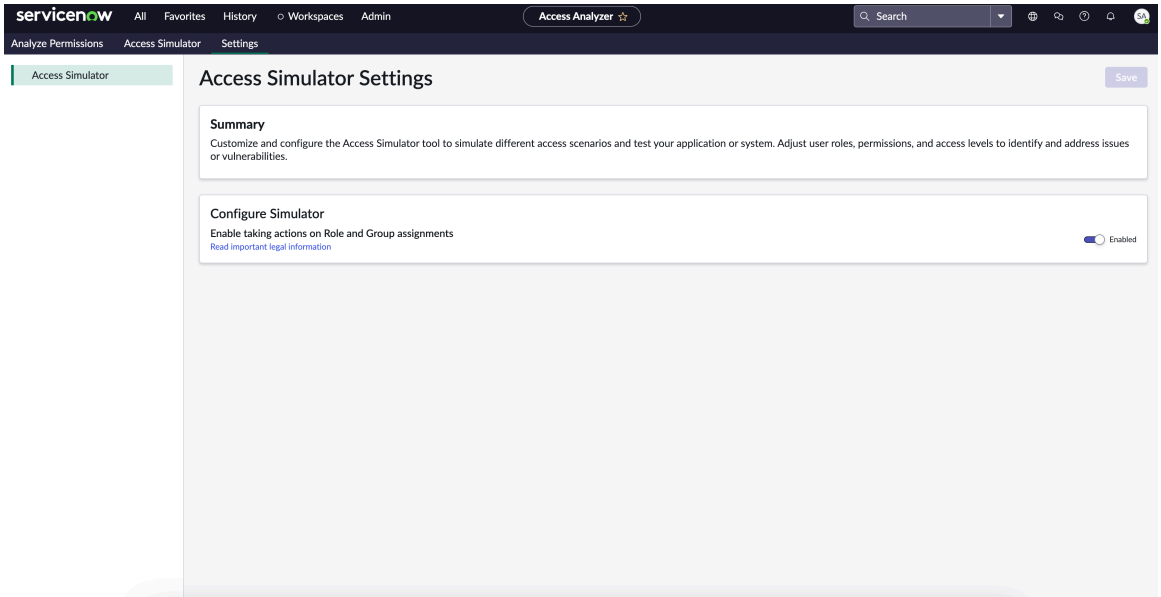
「アクションの実行」オプションを構成して、シミュレーション後に追加するロールとグループをアサインします。

始める前に

必要なロール:admin、access_analyzer_admin

手順

1. 移動先 すべて > アクセスアナライザー > 設定.



2. 有効にする切り替えスイッチを選択します。
   アクセスシミュレーター (アクションの実行) を有効にする前に、法務情報に目を通します。
3. [承認] を選択します。
4. [保存] を選択して構成を保存します。

アクセスシミュレーターの使用

アクセスシミュレーターを使用すると、ロールやグループをアサインまたは削除した後に、指定したテーブルへのユーザーのアクセス権がどのように変化するかをシミュレートできます。

始める前に

必要なロール:admin、access_analyzer_admin

「アクションの実行」を有効にします。詳細については、「アクセスシミュレーターの構成 (アクションの実行)」を参照してください。

手順

1. 移動先 すべて > アクセスアナライザー > アクセスシミュレーター.
2. 以下のシナリオに基づいて、各セクションの下の [シミュレート] オプションを選択します。
   - ユーザーへのロールの追加
   - ユーザーからのロールの削除
   - グループへのユーザーの追加
   - グループからのユーザーの削除

ユーザーへのロールの追加

リソース (テーブル) に対するユーザーのアクセス変更をシミュレートするには、[ロールの追加をシミュレート **(Simulate Add Role)]** を使用します。

始める前に
必要なロール:admin、access_analyzer_admin

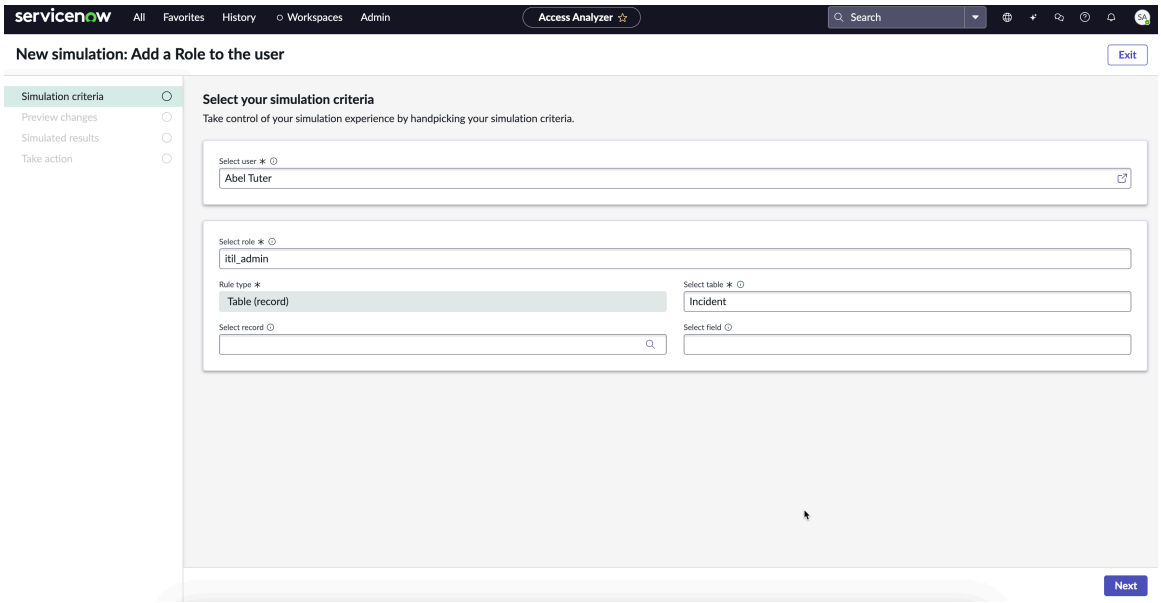
「アクションの実行」を有効にします。詳細については、「アクセスシミュレーターの構成 (アクションの実行)」を参照してください。

手順

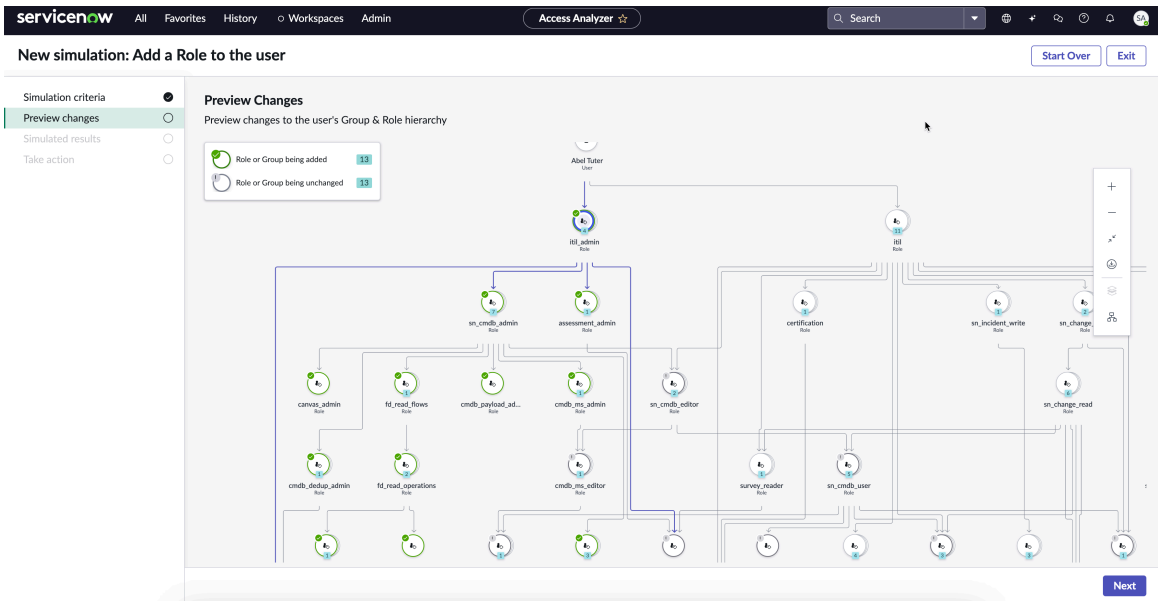
1. 移動先 すべて > アクセスアナライザー > アクセスシミュレーター.
2. [ロールをユーザーに追加 (Add a Role to the user)] セクションから [シミュレート] を選択します。
3. シミュレーション基準として以下のフィールドを指定します。

そのユーザーにロールを追加

フィールド	説明
ユーザーを選択 *	リストから選択するユーザー名を指定します。**Abel Tuter** など。
ロールを選択 *	リストから選択するロールを指定します。この例では、**itil_admin** です。
ルールタイプ *	[ルールタイプ] は自動入力され、変更することはできません。
テーブルを選択 *	リストから選択するテーブル名を指定します。この例では、[インシデント] です。
レコードを選択	リストから選択するレコード名を指定します (オプション)。
フィールドを選択	リストから選択するフィールド名を指定します。このフィールドは、フィールドレベルで権限を分析するためにも使用できます。たとえば、アクティブ、作成者などです。

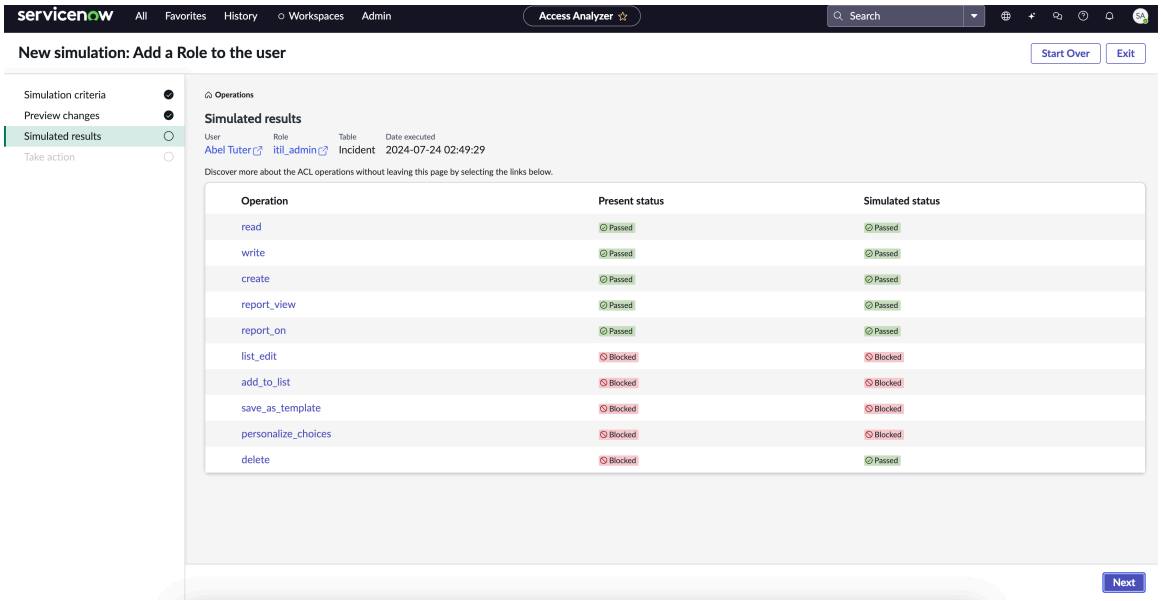


4. [次へ] を選択します。
5. 変更をプレビューします。
   新しくアサインされるロールが、[変更のプレビュー] でシミュレートされます。次のステップに進む前に、ユーザーに追加されるロールと子ロールを検証できます。



   itil_admin ロールの一部として追加される新しいロールが、ユーザーの既存のロールとともに表示されます。

6. [次へ] を選択します。
7. [現在のステータス **(Present status)**] と [シミュレーションのステータス **(Simulated status)**] を確認して、シミュレートされたユーザーに対してアクセスが [合格] するか [ブロック済み] になるかを検証できます。

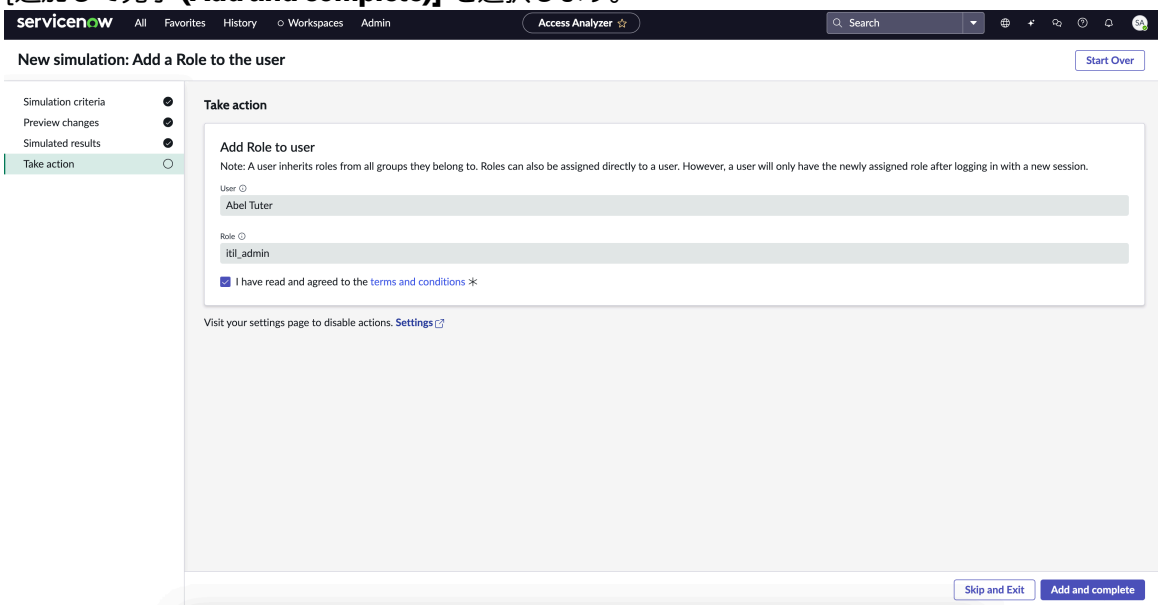


itil_admin を追加すると、ユーザーはテーブル (インシデント) に対して削除操作を実行できるようになります。

注:

- ACL (操作) について詳しく知りたい場合は、各レコードの操作リンクを選択します。
- 別のロールでシミュレーションを再開する場合は、[やり直します] を選択します。
- シミュレーションを終了するには、[終了] を選択します。

8. [次へ] を選択します。
9. [追加して完了 **(Add and complete)**] を選択します。



注:

- アクセスシミュレーターが有効になっていない場合は、シミュレーションを実行できません。有効にするには、 [アクションを有効化 **(Enable actions)]** を選択し、法的情報を承認します。
- シミュレーションを非表示にするには、[アクションを非表示] を選択します。アクションを再表示して有効にするには、設定に移動します。詳細については、「アクセスシミュレーターの構成 (アクションの実行)」を参照してください。
- シミュレーションを終了するには、[スキップして終了 **(Skip and Exit)]** を選択します。

ロールがユーザーに正常に追加されました。アクセスアナライザーを使用してアクセスを検証できます。ユーザーのアクセスを検証する方法の詳細については、「アクセスアナライザー」を参照してください。

ユーザーからのロールの削除

リソース (テーブル) に対するユーザーのアクセス変更をシミュレートするには、 [ロールの削除をシミュレート **(Simulate Remove Role)]** を使用します。

始める前に

必要なロール:admin、access_analyzer_admin

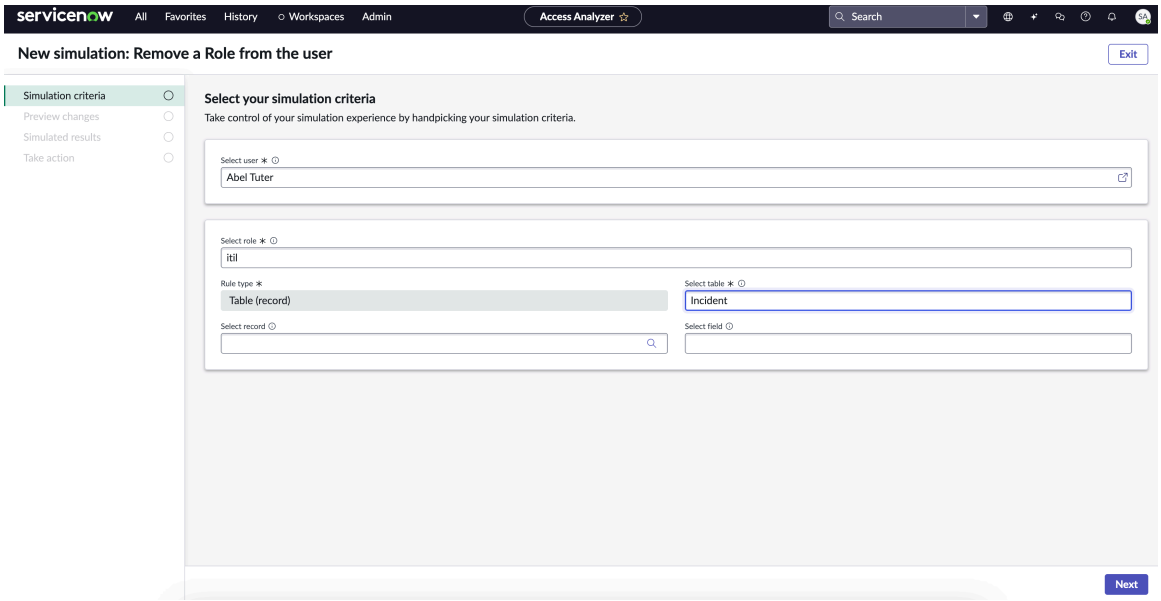
「アクションの実行」を有効にします。詳細については、「アクセスシミュレーターの構成 (アクションの実行)」を参照してください。

手順

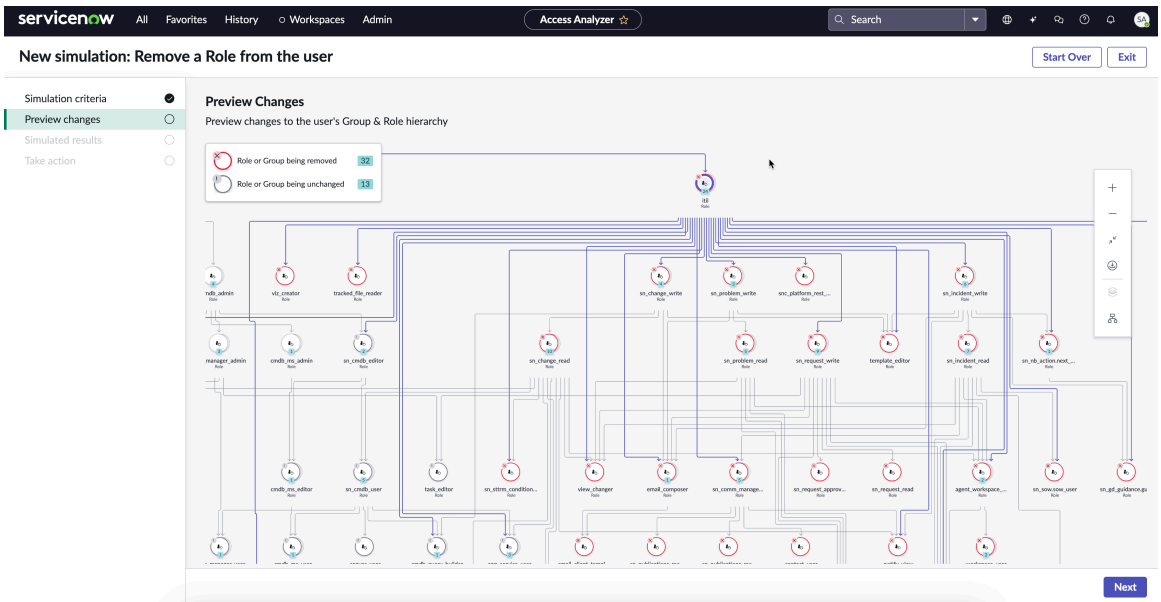
1. 移動先 すべて > アクセスアナライザー > アクセスシミュレーター.
2. [ロールをユーザーから削除 (Remove a Role from the user)] セクションから [シミュレート] を選択します。
3. シミュレーション基準として以下のフィールドを指定します。

ロールをユーザーから削除

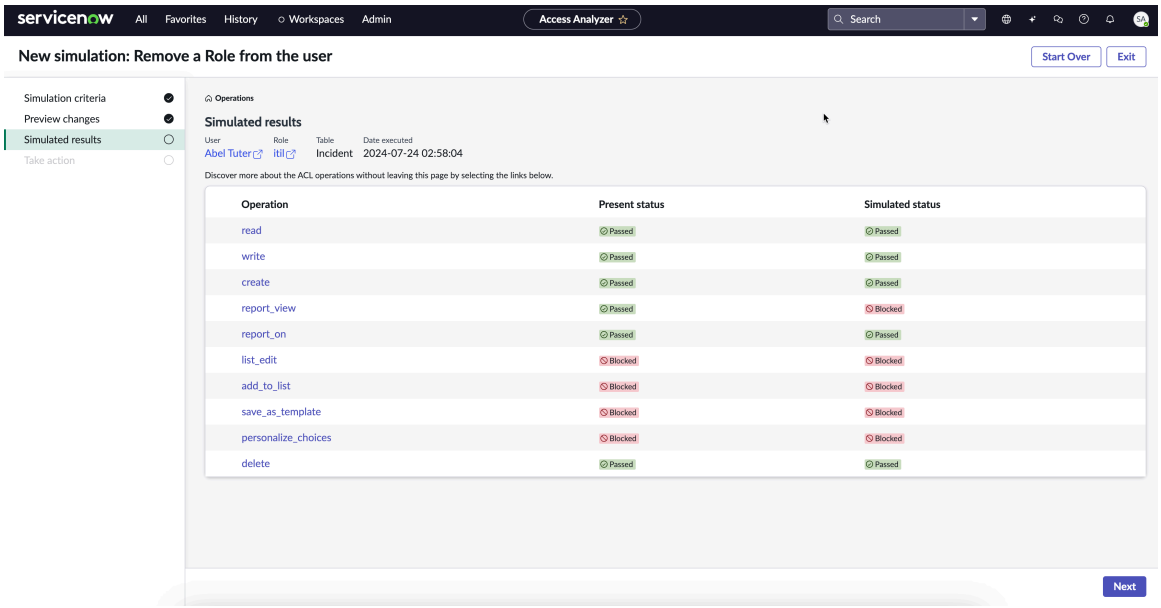
フィールド	説明
ユーザーを選択 *	リストから選択するユーザー名を指定します。**Abel Tuter** など。
ロールを選択 *	リストから選択するロールを指定します。この例では、**itil** です。
ルールタイプ *	[ルールタイプ] は自動入力され、変更することはできません。
テーブルを選択 *	リストから選択するテーブル名を指定します。この例では、[インシデント] です。
レコードを選択	リストから選択するレコード名を指定します。
フィールドを選択	リストから選択するフィールド名を指定します。このフィールドは、フィールドレベルで権限を分析するためにも使用できます。たとえば、アクティブ、作成者などです。



4. [次へ] を選択します。
5. 変更をプレビューします。
   削除されるロールが、[変更のプレビュー] でシミュレートされます。次のステップに進む前に、ユーザーから削除されるロールと子ロールを検証できます。



6. [次へ] を選択します。
7. [現在のステータス **(Present status)**] と [シミュレーションのステータス **(Simulated status)**] を確認して、シミュレートされたユーザーに対してアクセスが [合格] するか [ブロック済み] になるかを検証できます。



注:

- ACL (操作) について詳しく知りたい場合は、各レコードの操作リンクを選択します。
- 別のロールでシミュレーションを再開する場合は、[やり直します] を選択します。
- シミュレーションを終了するには、[終了] を選択します。

8. [次へ] を選択します。
9. [削除して完了 **(Remove and complete)]** を選択します。

注:

- アクセスシミュレーターが有効になっていない場合は、シミュレーションを実行できません。有効にするには、 [アクションを有効化 **(Enable actions)]** を選択し、法的情報を承認します。
- シミュレーションを非表示にするには、[アクションを非表示] を選択します。アクションを再表示して有効にするには、設定に移動します。詳細については、「アクセスシミュレーターの構成 (アクションの実行)」を参照してください。
- シミュレーションを終了するには、[スキップして終了 **(Skip and Exit)]** を選択します。

ロールがユーザーから正常に削除されました。アクセスアナライザーを使用してアクセスを検証できます。ユーザーのアクセスを検証する方法の詳細については、「アクセスアナライザー」を参照してください。

グループへのユーザーの追加

ユーザーがグループに追加されたときの、リソース (テーブル) に対するユーザーのアクセス変更をシミュレートするには、[グループへの追加をシミュレート **(Simulate Add to Group)]** を使用します。

始める前に
必要なロール:admin、access_analyzer_admin

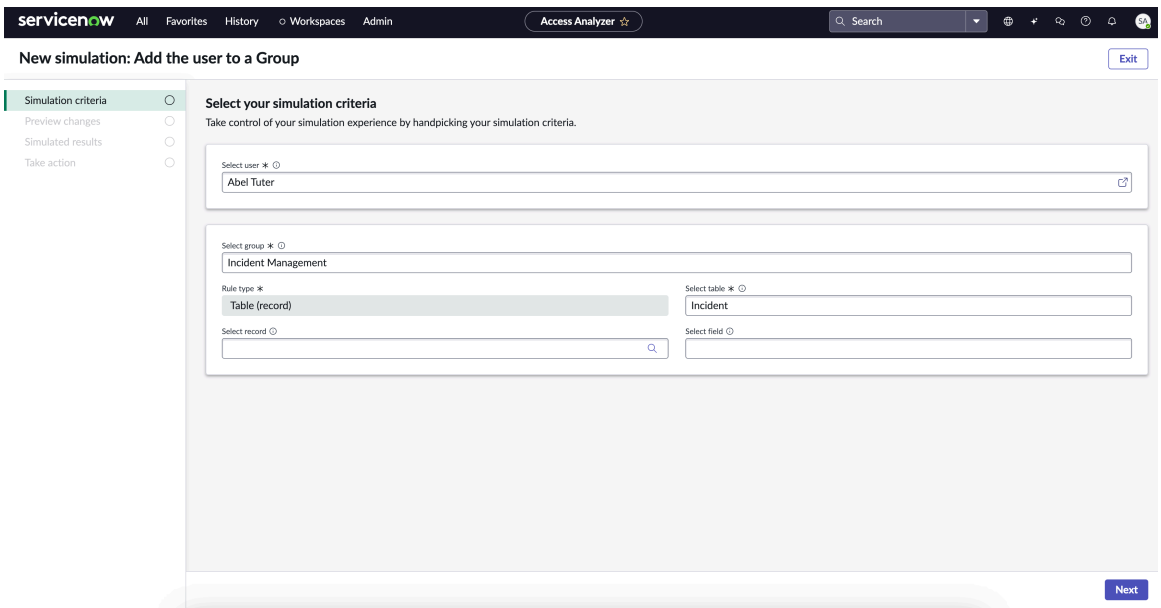
「アクションの実行」を有効にします。詳細については、「アクセスシミュレーターの構成 (アクションの実行)」を参照してください。

手順

1. 移動先 すべて > アクセスアナライザー > アクセスシミュレーター.
2. [そのユーザーをグループに追加 (Add the user to a Group)] セクションから [シミュレート] を選択します。
3. シミュレーション基準として以下のフィールドを指定します。

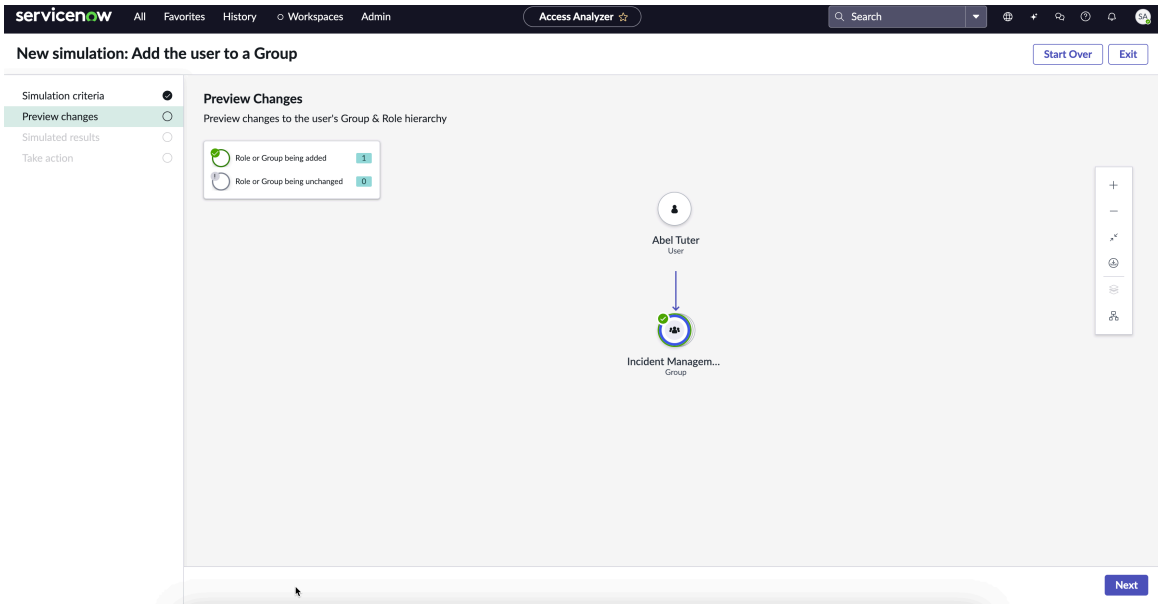
グループにユーザーを追加する

フィールド	説明
ユーザーを選択 *	リストから選択するユーザー名を指定します。**Abel Tuter** など。
グループを選択 *	リストから選択するグループを指定します。この例では、[インシデント管理] です。
ルールタイプ *	[ルールタイプ] は自動入力され、変更することはできません。
テーブルを選択 *	リストから選択するテーブル名を指定します。この例では、[インシデント] です。
レコードを選択	リストから選択するレコード名を指定します。
フィールドを選択	リストから選択するフィールド名を指定します。このフィールドは、フィールドレベルで権限を分析するためにも使用できます。たとえば、アクティブ、作成者などです。

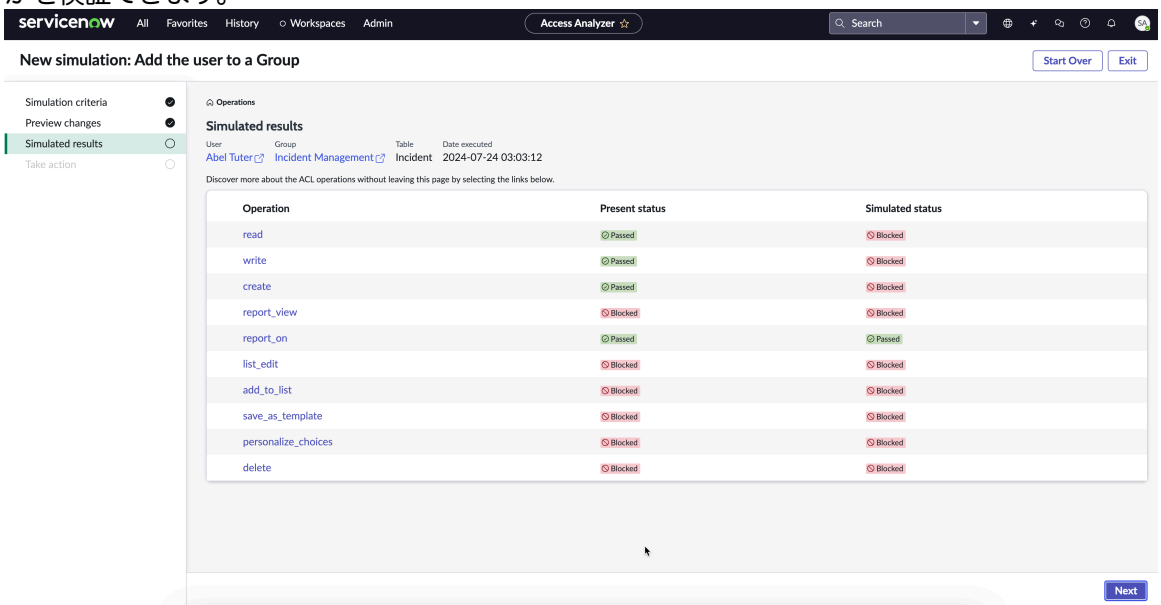


4. [次へ] を選択します。
5. 変更をプレビューします。

   ユーザーがアサインされるグループが、[変更のプレビュー] でシミュレートされます。次のステップに進む前に変更を検証できます。



6. [次へ] を選択します。
7. [現在のステータス **(Present status)**] と [シミュレーションのステータス **(Simulated status)**] を確認して、シミュレートされたユーザーに対してアクセスが [合格] するか [ブロック済み] になるかを検証できます。



> **注:**
>
> - ACL (操作) について詳しく知りたい場合は、各レコードの操作リンクを選択します。
> - 別のロールでシミュレーションを再開する場合は、[やり直します] を選択します。
> - シミュレーションを終了するには、[終了] を選択します。

8. [次へ] を選択します。
9. [追加して完了 **(Add and complete)**] を選択します。

注:

- アクセスシミュレーターが有効になっていない場合は、シミュレーションを実行できません。有効にするには、 [アクションを有効化 **(Enable actions)]** を選択し、法的情報を承認します。
- シミュレーションを非表示にするには、[アクションを非表示] を選択します。アクションを再表示して有効にするには、設定に移動します。詳細については、「アクセスシミュレーターの構成 (アクションの実行)」を参照してください。
- シミュレーションを終了するには、[スキップして終了 **(Skip and Exit)]** を選択します。

ユーザーがグループに正常に追加されました。

グループからのユーザーの削除

ユーザーがグループから削除されたときの、リソース (テーブル) に対するユーザーのアクセス変更をシミュレートするには、[グループからの削除をシミュレート **(Simulate Remove from Group)]** を使用します。

始める前に

必要なロール:admin、access_analyzer_admin

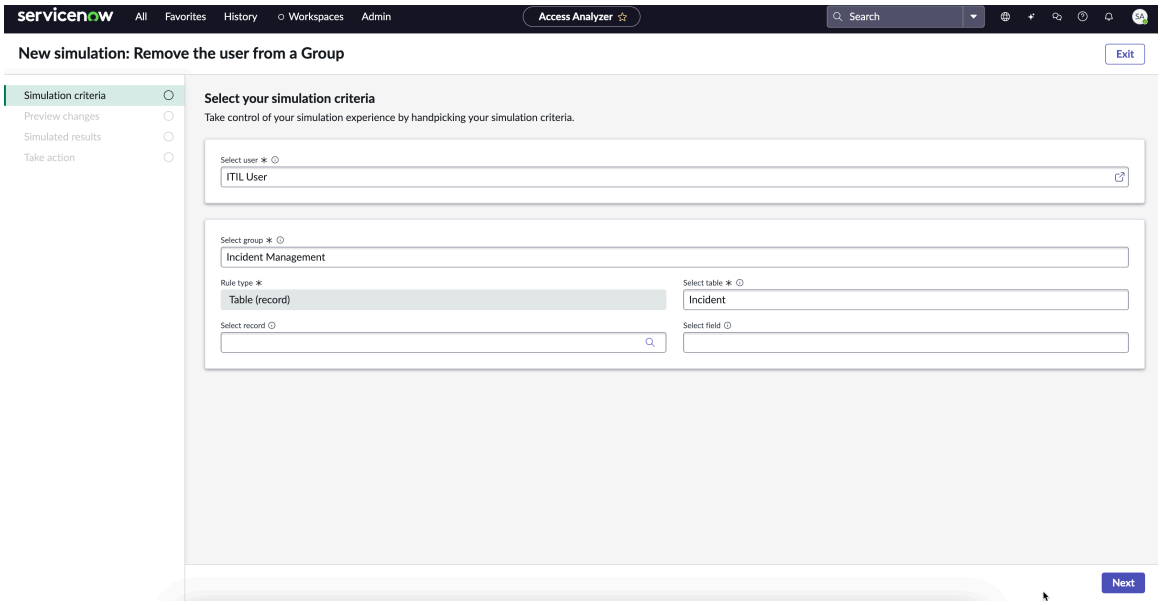
「アクションの実行」を有効にします。詳細については、「アクセスシミュレーターの構成 (アクションの実行)」を参照してください。

手順

1. 移動先 すべて > アクセスアナライザー > アクセスシミュレーター.
2. [グループからユーザーを削除 (Remove the user from a Group)] セクションから [シミュレート] を選択します。
3. シミュレーション基準として以下のフィールドを指定します。

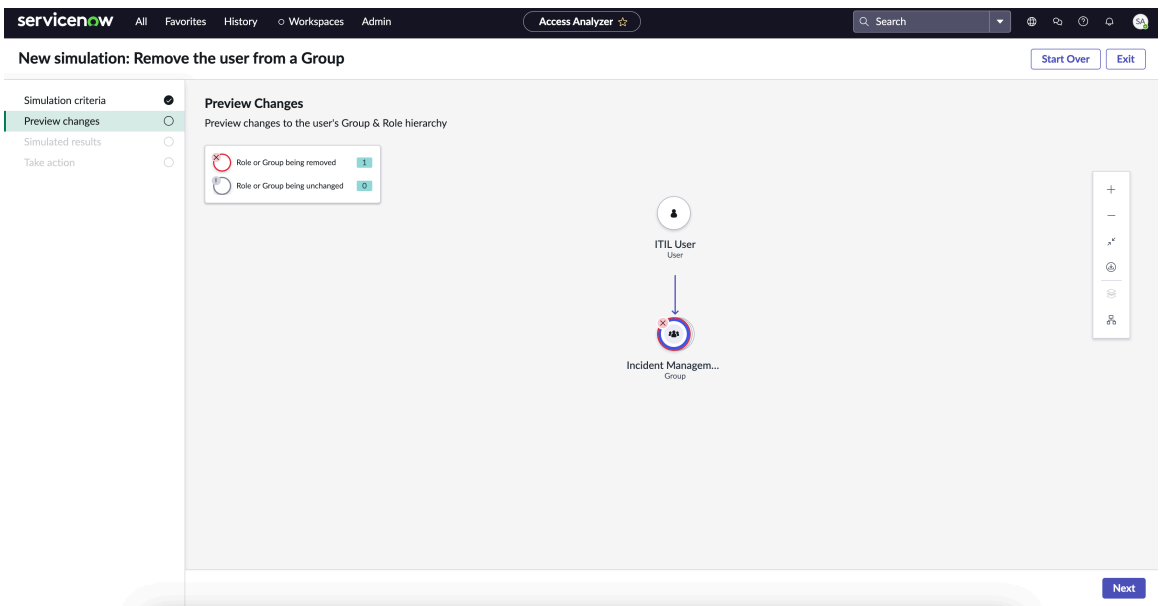
グループからユーザーを削除

フィールド	説明
ユーザーを選択 *	リストから選択するユーザー名を指定します。この例では、**[ITIL** ユーザー **(ITIL User)]** です。
グループを選択 *	リストから選択するグループを指定します。この例では、[インシデント管理] です。
ルールタイプ *	[ルールタイプ] は自動入力され、変更することはできません。
テーブルを選択 *	リストから選択するテーブル名を指定します。この例では、[インシデント] です。
レコードを選択	リストから選択するレコード名を指定します。
フィールドを選択	リストから選択するフィールド名を指定します。このフィールドは、フィールドレベルで権限を分析するためにも使用できます。たとえば、アクティブ、作成者などです。

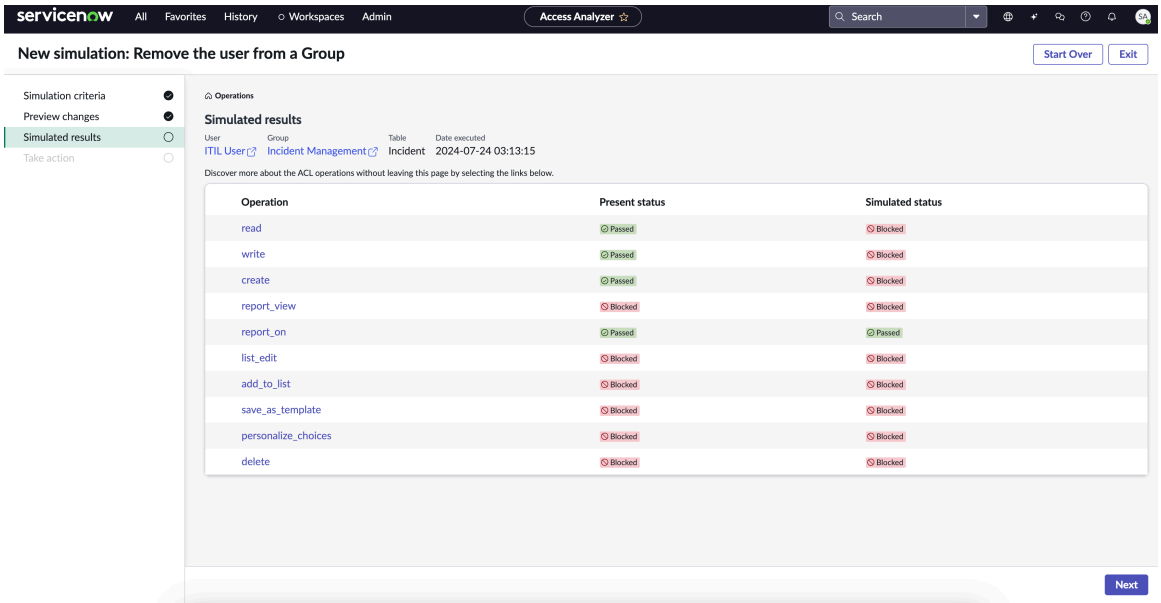


4. [次へ] を選択します。
5. 変更をプレビューします。

   ユーザーが削除されるグループが、[変更のプレビュー] でシミュレートされます。次のステップに進む前に変更を検証できます。



6. [次へ] を選択します。
7. [現在のステータス **(Present status)**] と [シミュレーションのステータス **(Simulated status)**] を確認して、シミュレートされたユーザーに対してアクセスが [合格] するか [ブロック済み] になるかを検証できます。



注:

- ACL (操作) について詳しく知りたい場合は、各レコードの操作リンクを選択します。
- 別のロールでシミュレーションを再開する場合は、[やり直します] を選択します。
- シミュレーションを終了するには、[終了] を選択します。

8. [次へ] を選択します。
9. [削除して完了 **(Remove and complete)]** を選択します。

注:

- アクセスシミュレーターが有効になっていない場合は、シミュレーションを実行できません。有効にするには、 [アクションを有効化 **(Enable actions)]** を選択し、法的情報を承認します。
- シミュレーションを非表示にするには、[アクションを非表示] を選択します。アクションを再表示して有効にするには、設定に移動します。詳細については、「アクセスシミュレーターの構成 (アクションの実行)」を参照してください。
- シミュレーションを終了するには、[スキップして終了 **(Skip and Exit)]** を選択します。

ユーザーがグループから正常に削除されました。

よく寄せられる質問

アクセスシミュレーターの使用に関してよく寄せられる質問です。

よく寄せられる質問

質問	説明
シミュレーターを使い始めるには？	さまざまな状況で、ユーザーに対してシミュレートし、手順を確認する [シミュレート] オプションを選択できます。
シミュレーターはどのように機能しますか？	アクセスシミュレーターは、選択したユーザーに対する指定されたテーブルへのアクセスをシミュレートし、アクセスプロビジョニングに

よく寄せられる質問 (続く)

質問	説明
	関する決定を下すためにアクセスの変更をプレビューします。
シミュレーターでアクションを有効または無効にする方法は？	アクセスシミュレーターのアクションを有効または無効にするには、[設定] に移動するか、シミュレーションプロセスの「アクションの実行」ステップを使用します。

アクセスインサイト

アクセスインサイトにより、ロールまたはグループのアサインを確実に確保できます。

探索

Access Insights の機能とビジネス価値について説明します。

構成

アクセスインサイトの構成方法について説明します。

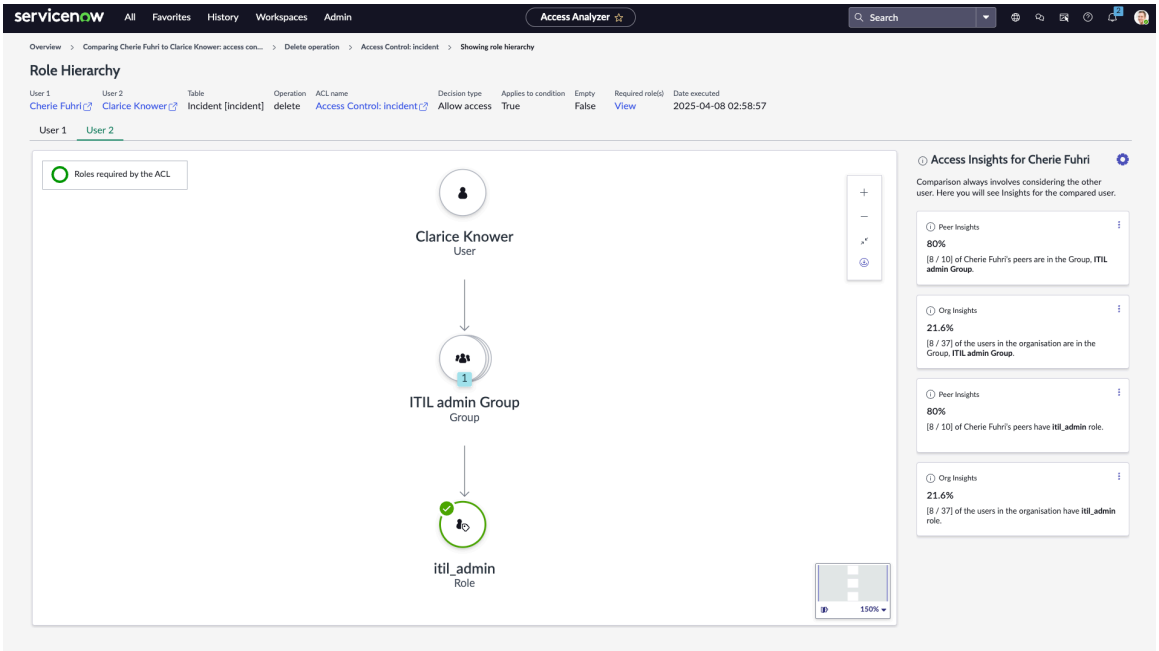
使用方法	
	
アクセスインサイトの 使用方法を理解します。	

アクセスインサイトの詳細

アクセスアナライザーのアクセスインサイトは、ロールまたはグループのアサインを確実にするのに役立ちます。

アクセスインサイトには、リソースへのユーザーのピアおよび組織レベルのアクセスの統計情報が表示されます。これにより、追加のエンタイトルメントについて現在レビューしているユーザーにロールまたはグループをアサインできるかどうかを確認できます。同じピアグループ内のロールとグループのアサインを比較するのに役立ちます。

アクセスインサイトは、アクセスアナライザーのユーザーアクセスの比較機能を使用してユーザーのアクセスを比較しているときにのみ表示されます。Access Insights の使用の詳細については、「アクセスインサイトの使用」を参照してください。



統計情報に基づいて、ピア内で比較しているユーザーに適切なロールまたはグループの割り当てを提供することを決定できます。

注:

- アクセスインサイトは、アクセスアナライザー V4 で使用できます。
- アクセスインサイト機能内の統計情報を表示しながら、ピアレベル (同じ組織、同じ部門、同じマネージャー) でユーザーを比較する必要があります。

アクセスアナライザーでアクセスインサイトを有効にするには、[設定] に移動する必要があります。詳細については、「アクセスインサイトの構成」を参照してください。

アクセスインサイトの構成

アクセスアナライザーでアクセスインサイト機能を有効にします。

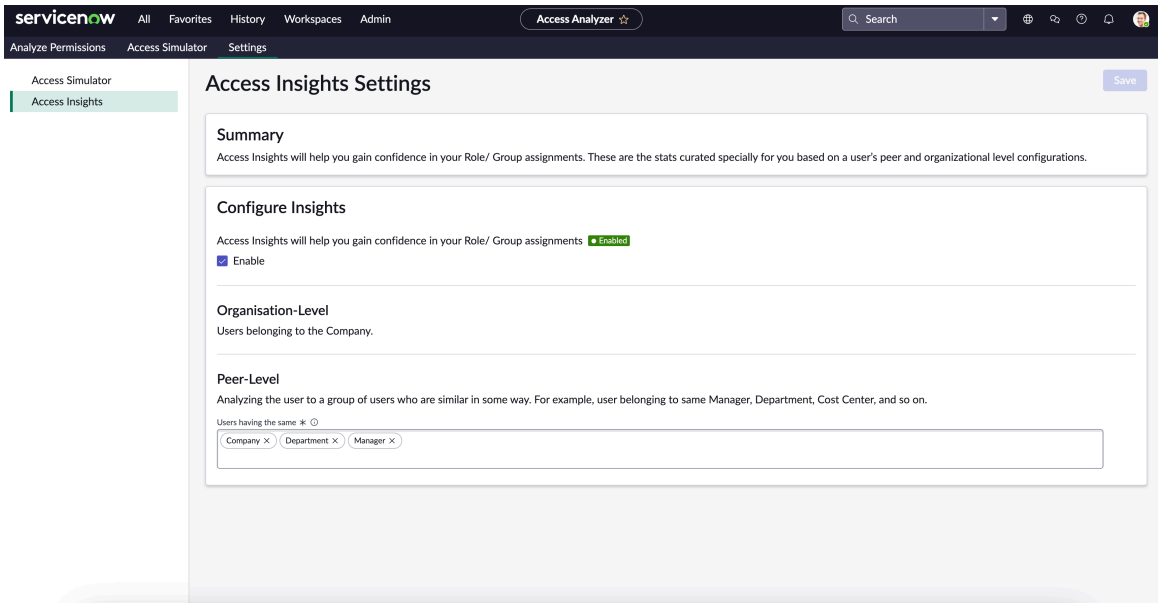
始める前に

必要なロール:admin、access_analyzer_admin

次の手順では、アクセスインサイトを有効にする手順について説明します。

手順

1. 移動先 すべて > アクセスアナライザー > アクセスの分析.
2. [設定] タブを選択します。
3. [アクセスインサイト] を選択します。
4. アクセスインサイトを有効にします。



注: デフォルトでは、[ピアレベル] フィールドは、[会社]、[部門]、[マネージャー] とともに選択されています。[コストセンター]、[国コード]、[場所]、[タイトル] などのフィールドをさらに追加できます。

5. **[Save** (保存)] を選択します。

結果
アクセスインサイト機能は、リソースへのユーザーアクセスの比較中に ロール階層 を表示しているときに有効になり、表示されます。Access Insights の使用の詳細については、「 アクセスインサイトの使用」を参照してください。

アクセスインサイトの使用

アクセスインサイトを使用して、選択したリソースへのユーザーのピアレベルのアクセスを把握します。

始める前に
必要なロール:admin、access_analyzer_admin

注: アクセスインサイト機能の詳細を表示するには、 ユーザーアクセスの比較 機能を使用する必要があります。

次の手順では、アクセスインサイト機能内でピアレベルのロールまたはグループの割り当ての統計を表示する手順について説明します。

手順

1. 移動先 すべて > アクセスアナライザー > 権限を分析.
   [アクセスと権限の分析 (Analyze access and permissions)] ホームページが表示されます。
2. [**Compare user access**] タブを選択します。
3. 次のフィールドに入力します。

ユーザーアクセスを比較

フィールド	説明
ユーザー 1* を選択	リストから選択するユーザー名を指定して比較します。
ユーザー 2* を選択	リストから選択するユーザー名を指定してユーザー 1 と比較します。
ルールタイプ*	テーブルのアクセス許可を分析します。 注: ユーザーアクセスの比較で使用できるのは、テーブルのアクセス権限のみです。
テーブルを選択*	リストから選択するテーブル名を指定します。
レコードを選択	リストから選択するレコード名を指定します。
フィールドを選択	リストから選択するフィールド名を指定します。

4. [ユーザーアクセスの比較 **(Compare user access)]** を選択します。

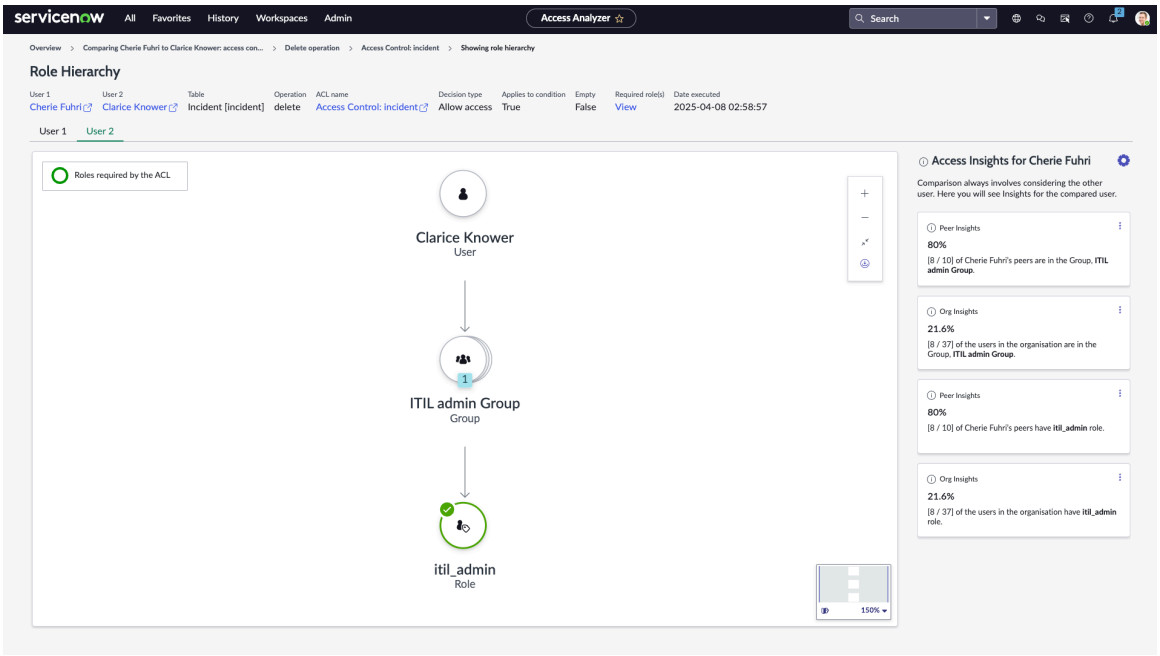
   選択されたユーザーのユーザーアクセスの比較結果が表示されます。

   ユーザーアクセスの比較結果には、ユーザーの操作とアクセス評価ステータスが表示されます。たとえば、Charlier Fuhri や Charlie Knower などです。

5. 操作を選択します。
   たとえば、読み取り操作です。
6. アクセス制御のいずれかを選択して、アクセスの詳細を確認します。
7. [ロール階層を表示] を選択して、ピアレベルのアクセスでアクセス比較を表示します。

   アクセスインサイトは次のように表示されます。

   - ピアインサイト:ピア内 (同じマネージャーまたは部門) 内で同じロールが割り当てられているユーザーの数を表示します。
   - 組織インサイト:組織内で同じロールが割り当てられているユーザーの数を表示します。



この情報に基づいて、比較したユーザーにアクセス権を付与するか取り消すかを選択できます。

Global Identity

Global Identity は、複数のインスタンス間で一意のユーザーを識別できるようにする ServiceNow® 製品です。

Global Identity は、複数の ServiceNow® インスタンスでユーザー、その属性、およびその他の基盤データを複数の ServiceNow インスタンス間で管理するのに役立ちます。

自動翻訳

フェデレーション ID の詳細

フェデレーション ID の主な機能とビジネス価値について学びます。

フェデレーション ID へのアクセス

フェデレーション ID にアクセスします。

ID フィールドの更新

フェデレーション ID を
更新して設定します。

フェデレーション **ID** の探索

ユーザー名とメールに基づいて複数のインスタンス間でユーザーを特定し、インスタンス全体でユーザーに一意の ID (フェデレーション ID) を提供します。

フェデレーション ID は、複数の ServiceNow® インスタンス間でユーザーを識別するために使用されます。フェデレーション ID に基づいてユーザーを識別し、複数のインスタンス間で正確なユーザー数を判断できます。詳細については、フェデレーション ID を参照してください。

注: ユーザー名はフェデレーション ID を生成するために必要です。

フェデレーション ID は、ServiceNow® インスタンス全体でハッシュ関数を使用する ID の一意の識別子です。

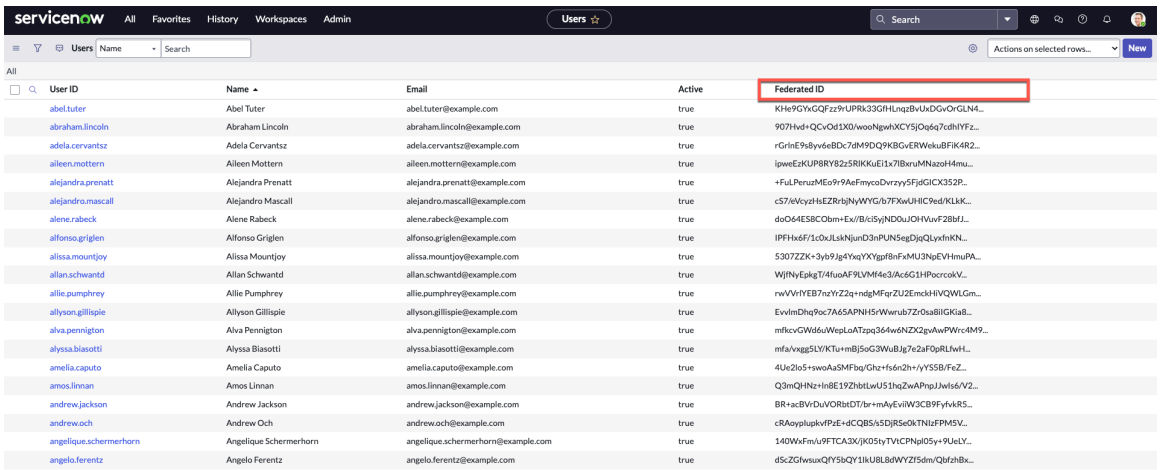
インスタンス全体でユーザーの [ユーザー **ID**] と [メール] を使用することで、フェデレーション ID が作成され、sys_user テーブルに表示されます。

Zurich リリースにアップグレードすると、Federated ID Generation (com.glide.identity.globalid) プラグインがすべてのインスタンスに自動インストールされます。

自動翻訳

注:

- ユーザー名はフェデレーション ID を生成するために必要です。
- ユーザー名とメールはフェデレーション ID を生成するためにデフォルトで使用されます。要件に基づいてフェデレーション ID を生成するためのフィールドを更新するには、「ID フィールドの更新」を参照してください。
- 構成を更新するには、**Iamsync_admin** ロールが必要です。
- ユーザー名とメールが重複しているユーザーがいる場合、フェデレーション ID は 1 人のユーザーに対してのみ生成されます。ユーザー名が null または空の場合、フェデレーション ID は null です。



プラグインをインストールした後のスキーマの変更は次のとおりです。

- sys_user テーブルに新しい列フェデレーション ID が作成されます。
- 新しいテーブル iamsync_type には、sys_user テーブルのデフォルト構成が自動的に入力されます。

フェデレーション ID は sys_user テーブルと sys_user テーブルを拡張するすべてのテーブルでのみサポートされています。

Zurich リリースにアップグレードすると、Federated ID Generation (com.glide.identity.globalid) プラグインがすべてのインスタンスに自動インストールされます。

フェデレーション ID 基準へのアクセス

フェデレーション ID 基準にアクセスして、フェデレーション ID の生成に使用された ID フィールドについて確認します。

始める前に
必要なロール：iamsync_admin

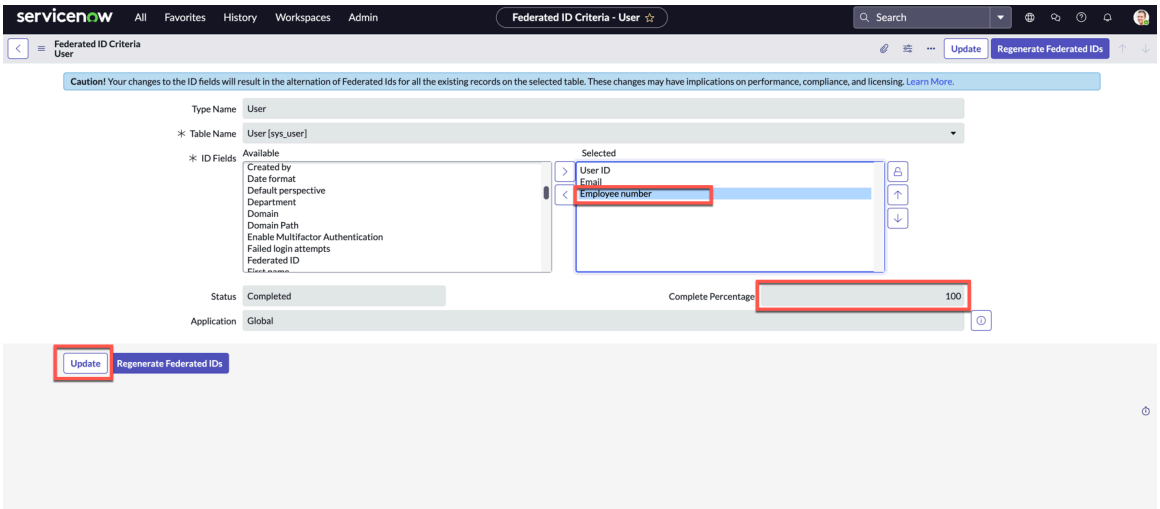
手順

1. 移動先 すべて > フェデレーション **ID** を管理 > フェデレーション **ID** 基準.
2. [フェデレーション ID 基準 (Federated ID Criterias)] ページには、次の詳細を含むレコードが表示されます。

- タイプ名：ユーザー
- テーブル名：ユーザー **[sys_user]**
- ID フィールド：**user_name** (ユーザー ID)、email (フェデレーション ID の生成に使用されるデフォルトフィールド)。
- ステータス：完了 (フェデレーション ID 生成ステータス)。利用可能なステータス：準備完了、実行中、完了、エラー

注:

- ユーザー名はフェデレーション ID を生成するために必要です。
- ユーザー名とメールはフェデレーション ID を生成するためにデフォルトで使用されます。



注: ID フィールドのみを更新して既存のレコードの新しいフェデレーション ID を生成できます。詳細については、「ID フィールドの更新」を参照してください。

ID フィールドの更新

ID フィールドを更新し、更新されたフィールドに基づいてフェデレーション ID を再生成します。

始める前に
必要なロール：iamsync_admin

注: ID フィールドを変更すると、選択したテーブルのすべての既存レコードのフェデレーション ID が変更されます。これらの変更は、パフォーマンス、コンプライアンス、およびライセンスに影響を与える可能性があります。

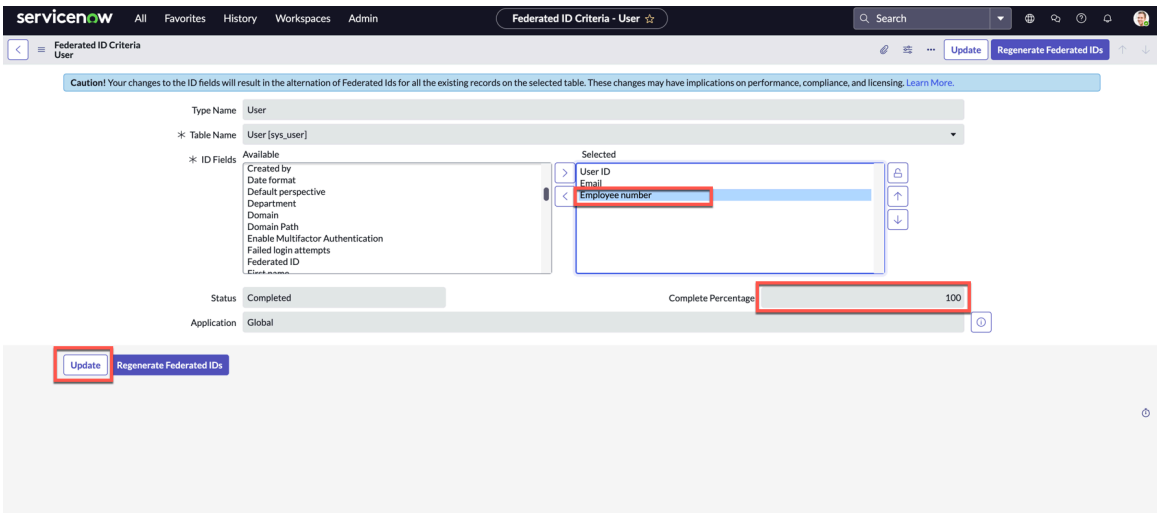
手順

1. 移動先 すべて > フェデレーション **ID** を管理 > フェデレーション **ID** 基準.
2. タイプ名 (ユーザー) を選択します。
3. [フェデレーション ID 基準ユーザー (Federated ID Criteria User)] ページで、矢印ボタンを使用して [利用可能] から [選択済み] に追加する新しい ID フィールドを選択します。

   例：[従業員番号]

注:

- ユーザー名はフェデレーション ID を生成するために必要です。
- ユーザー名とメールはフェデレーション ID を生成するためにデフォルトで使用されます。
- ユーザー名とメールが重複しているユーザーがいる場合、フェデレーション ID は 1 人のユーザーに対してのみ生成されます。ユーザー名が null または空の場合、フェデレーション ID は null です。



選択した [従業員番号] は、フェデレーション ID を生成するための別の属性になります。

4. [更新] を選択してフェデレーション ID を生成します。

注: 最初に [更新] を選択し、別の更新を開始する前に完了率 (100) を確認します。

ステータスの割合は、インスタンス全体のすべての ID のフェデレーション ID の生成を示します。

注:

- 前の更新ジョブが完了するまで、フィールドを変更しないでください。
- 更新されるフィールドは文字列タイプである必要があります。
- ID フィールドとして選択できないフィールドは次のとおりです。
  - システムレベルフィールド
  - エッジ暗号化フィールド
  - パスワードフィールド

5. sys_user テーブルに移動して、ID フィールドの更新によって生成された新しいフェデレーション ID を表示します。

注: XML インポート、低レベルのデータベース更新を介してユーザーが作成または更新され、インスタンスが正しく機能しない場合は、[フェデレーション **ID** を再生成 **(Regenerate Federated IDs)]** を選択します。[フェデレーション **ID** を再生成 **(Regenerate Federated IDs)]** を選択すると、現在の ID フィールド基準を使用して、すべてのユーザーの ID が再生成されます。

ID とアクセスの監査

ID とアクセスの監査を使用して、ユーザー、グループ、ロール、および ACL の変更を把握します。

探索

ID とアクセスの監査の機能とビジネス価値について説明します。

構成

ID とアクセスの監査を構成する方法を理解します。

監査結果

自動翻訳

ID とアクセスの監査の結果を表示します。

ID とアクセスの監査について

ID とアクセスの監査を使用して、ユーザー、グループ、ロール、および ACL の変更を把握します。

ID とアクセスの監査は、ユーザーアカウント、グループ、およびロールで誰が、何を、どこで、いつ変更したかに関する重要な情報を把握するのに役立ちます。

悪意のあるユーザーを検出し、ServiceNow® インスタンス内の異常なアクティビティを追跡し、アクセスの変更を追跡できるコンプライアンス標準に準拠するのに役立ちます。

ID とアクセスの監査 (ID セキュリティ監査) はプラグイン (com.glide.security.audit) で、自動インストールされます。

監査機能は、glide.identity.security.audit.enabled システムプロパティを切り替えることでオンまたはオフにできます。デフォルトでは、プロパティは true に設定されています。

ID とアクセスの監査により、以下が可能になります。

- 過去 30 日間にユーザー、グループ、ロール ACL 属性、ロールメンバーシップ、グループメンバーシップ、および ACL ロールに対して行われた変更を表示します。
- ServiceNow インスタンスの変更を追跡します。
- 潜在的なセキュリティおよび規制上のリスクを軽減するのに役立ちます。
- 組織内のさまざまなグループの監査人によるコンプライアンスを実証します。
- 組織が、ユーザーグループの可視化の欠如とロールの変更に関連する脅威に対して脆弱ではないことを示します。

ID アクセスと監査のユーザーペルソナ

ID とアクセスの監査のさまざまなユーザーペルソナを次に示します。

- アドミン：監査レコードと構成を表示します。
- **Security Admin**：これらの監査証跡を表示します。構成を変更して、特定のテーブルの監査を有効または無効にしたり、監査対象のフィールドを変更したりします。

監査テーブル

次のテーブルは、[ID とアクセスの監査] を使用して監査できます。

- グループ [sys_user_group]
- ロール [sys_user_role]
- アクセス制御 [sys_security_acl]
- ユーザー [sys_user]
- グループロール [sys_group_has_role]
- ユーザーロール [sys_user_has_role]
- アクセスロール [sys_security_acl_role]

自動翻訳

- 含まれているロール [sys_user_role_contains]
- グループメンバー [sys_user_grmember]☒

ID とアクセスの監査のモジュール

ID とアクセスの監査には、ServiceNow インスタンスで次のモジュールがあります。

モジュール	説明
監査結果	ServiceNow インスタンスで発生した監査を表示します。
テーブルとフィールドを構成	ID とアクセスの監査で利用可能なフィールドを使用して、システムテーブルとフィールドを構成します。
保存期間を構成	監査対象データの保持期間を設定します。設定できる最大期間は 30 日間です。
ユーザー記録	ユーザーの監査を表示します。
グループ記録	グループの監査を表示します。
ロール記録	ロールの監査を表示します。
ACL 記録	ACL の監査を表示します。

ID 監査結果

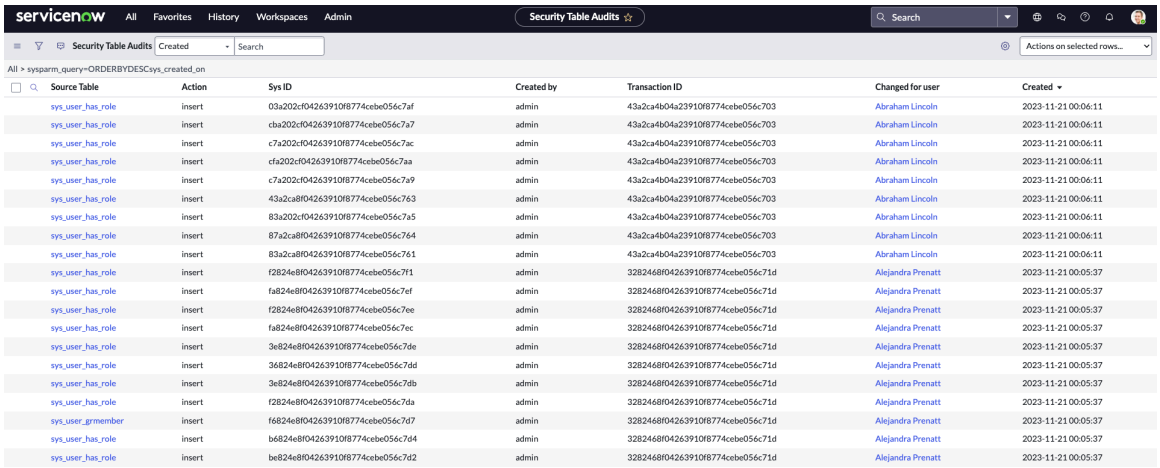
ServiceNow® インスタンスで発生した監査を表示します。

監査結果には、ServiceNow インスタンス内のユーザー、グループ、ロール、および ACL に加えられた変更が表示されます。

監査結果にアクセスするには、 すべて > システムセキュリティ > **ID** とアクセスの監査 > 監査結果. 次の情報を含むセキュリティテーブル監査ページが表示されます。

セキュリティテーブル監査

列名	説明
ソーステーブル	監査が発生したソーステーブルの詳細。
アクション	監査アクションの記述。
Sys_id	監査対象レコードの sys_id に関する詳細。
作成者	作成された監査の詳細。
トランザクション ID	実行された特定の監査のすべてのアクションを表す一意の ID。
変更依頼者	監査が実行されたときのユーザーの名前。
作成日時	監査が実施された日時。



ユーザー記録

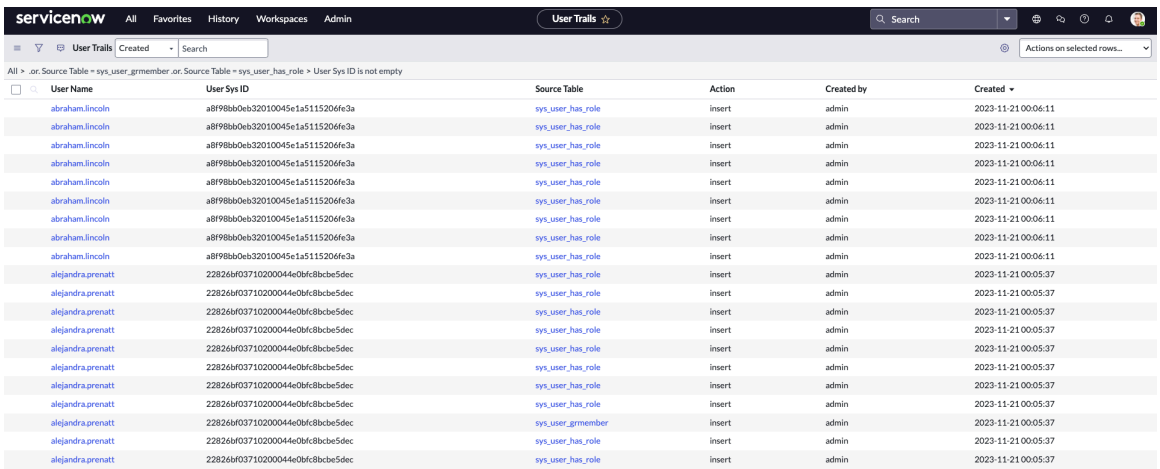
ServiceNow® インスタンス内のユーザの監査を表示します。

ユーザー記録には、ユーザの ID 属性の変更、ロールメンバーシップの変更、グルー プメンバーシップの変更が表示されます。

ユーザー記録にアクセスするには、 すべて > システムセキュリティ > **ID** とアクセスの監査 > ユーザー記録. 以下の情報がユーザー記録ページで表示されます。

ユーザー記録

ユーザー名	ユーザーの名前。
ユーザー Sys ID	監査対象レコードのユーザー sys_id の詳細。
ソーステーブル	監査が実行されたソーステーブルの詳細。
アクション	監査アクションを記述。
作成者	変更を加えたユーザー。
作成日時	監査が実施された日時。



グループ記録

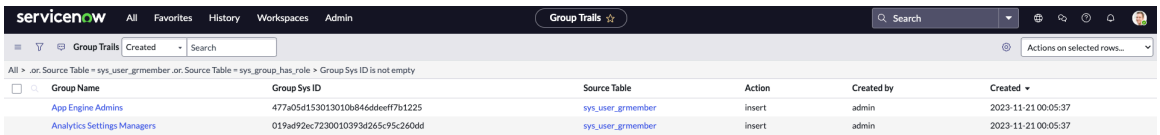
ServiceNow® インスタンス内のグループの監査を表示します。

グループ記録には、グループの属性の変更、メンバーシップの変更、およびロールの変更が表示されます。

グループ記録にアクセスするには、 すべて > システムセキュリティ > **ID** とアクセスの監査 > グループ記録. 以下の情報がグループ記録ページで表示されます。

グループ記録

グループ名	グループの名前。
グループの Sys ID	監査対象レコードのグループの sys_id の詳細。
ソーステーブル	監査が実行されたソーステーブルの詳細。
アクション	監査アクションを記述。
作成者	変更を加えたユーザー。
作成日時	監査が実施された日時。



ロール記録

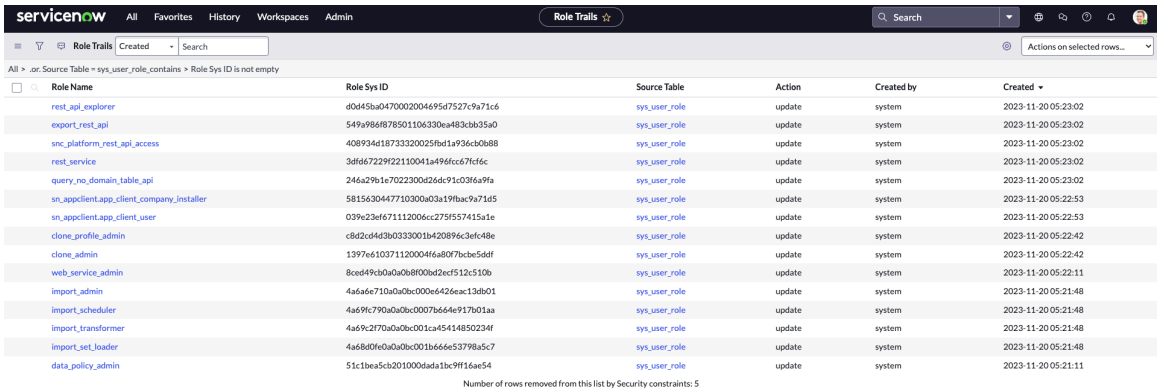
ServiceNow® インスタンス内のロールの監査を表示します。

ロール記録には、ロールの属性の変更と親子関係の変更が表示されます。

ロール記録にアクセスするには、次に移動します: すべて > システムセキュリティ > **ID** とアクセスの監査 > ロール記録. 以下の情報がロール記録ページで表示されます。

ロール記録

ロール名	ロールの名前。
ロール Sys ID	監査対象レコードのロール sys_id の詳細。
ソーステーブル	監査が実行されたソーステーブルの詳細。
アクション	監査アクションを記述。
作成者	変更を加えたユーザー。
作成日時	監査が実施された日時。



ACL 記録

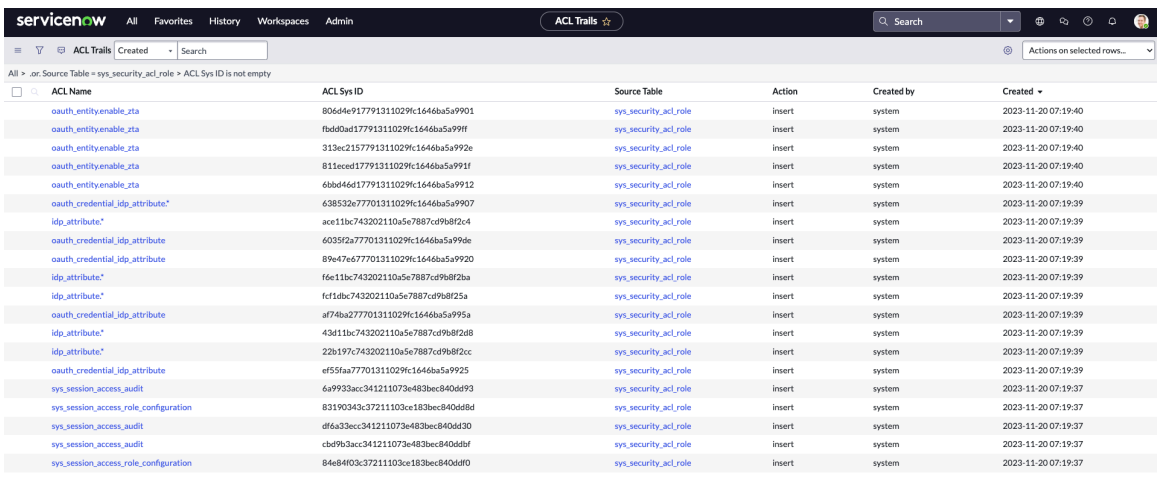
ServiceNow® インスタンス内の ACL の監査を表示します。

ACL 記録には、ACL の属性の変更と必要なロール関係の変更が表示されます。

ACL 記録にアクセスするには、 すべて > システムセキュリティ > **ID** とアクセスの監査 > **ACL** 記録. 以下の情報が ACL 記録ページで表示されます。

ACL 記録

ACL 名	ACL の名前。
ACL Sys ID	監査対象レコードの ACL sys_id の詳細。
ソーステーブル	監査が実行されたソーステーブルの詳細。
アクション	監査アクションを記述。
作成者	変更を加えたユーザー。
作成日時	監査が実施された日時。



セキュリティ監査可能フィールド

ServiceNow® インスタンスで監査されるテーブルおよびフィールドレベルの詳細を表示します。

セキュリティ監査可能フィールドでは、ServiceNow インスタンスで監査されるテーブルとフィールドの詳細が表示されます。

[セキュリティ監査可能フィールド] ページにアクセスするには、 すべて > システムセキュリティ > **ID** とアクセスの監査 > テーブルとフィールドを構成. セキュリティ監査可能フィールドページに以下の情報が表示されます。

セキュリティ監査可能フィールド

列名	説明
監査するテーブル	監査対象のテーブルの詳細。
監査ストレージの保存先	監査の詳細が保存される宛先の詳細。
フィールドリスト	リストで指定されたフィールドに対して監査が実行されます。
作成	作成操作に関連する変更を監査します。
更新	更新操作に関連する変更を監査します。
削除	削除操作に関連する変更を監査します。
アクティブ	テーブルの構成がアクティブな場合にのみ監査します。

次のテーブルは、[ID とアクセスの監査] を使用して監査できます。☒

- グループ [sys_user_group]
- ロール [sys_user_role]
- アクセス制御 [sys_security_acl]
- ユーザー [sys_user]☒
- グループロール [sys_group_has_role]
- ユーザーロール [sys_user_has_role]☒
- アクセスロール [sys_security_acl_role]
- 含まれているロール [sys_user_role_contains]
- グループメンバー [sys_user_grmember]☒

テーブルとフィールドの構成

ユーザー、グループ、ロール、および ACL に加えられた変更を把握するための ID とアクセスの監査。

始める前に
必要なロール：security_admin

ID とアクセスの監査のテーブルとフィールドを設定するには、ロールを Security Admin に昇格させる必要があります。

次のテーブルを監査用に構成できます。☒

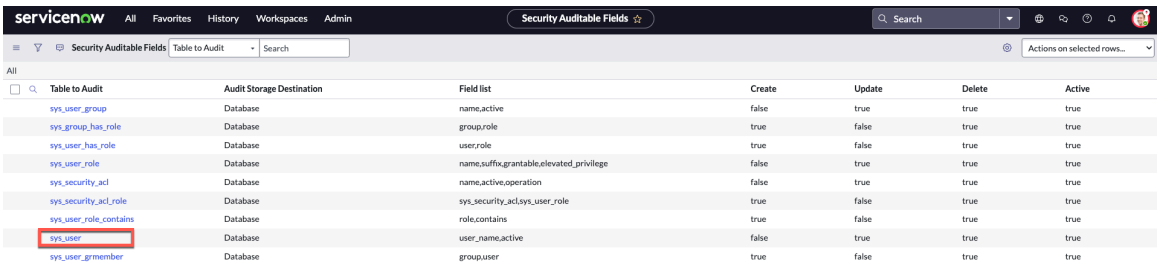
- グループ [sys_user_group]
- ロール [sys_user_role]
- アクセス制御 [sys_security_acl]
- ユーザー [sys_user]☒
- グループロール [sys_group_has_role]
- ユーザーロール [sys_user_has_role]☒
- アクセスロール [sys_security_acl_role]
- 含まれているロール [sys_user_role_contains]
- グループメンバー [sys_user_grmember]☒

注: テーブルに設定できるフィールドを把握するには、「ID アクセスと監査でサポートされているフィールドとサポートされていないフィールド」を参照してください。

手順

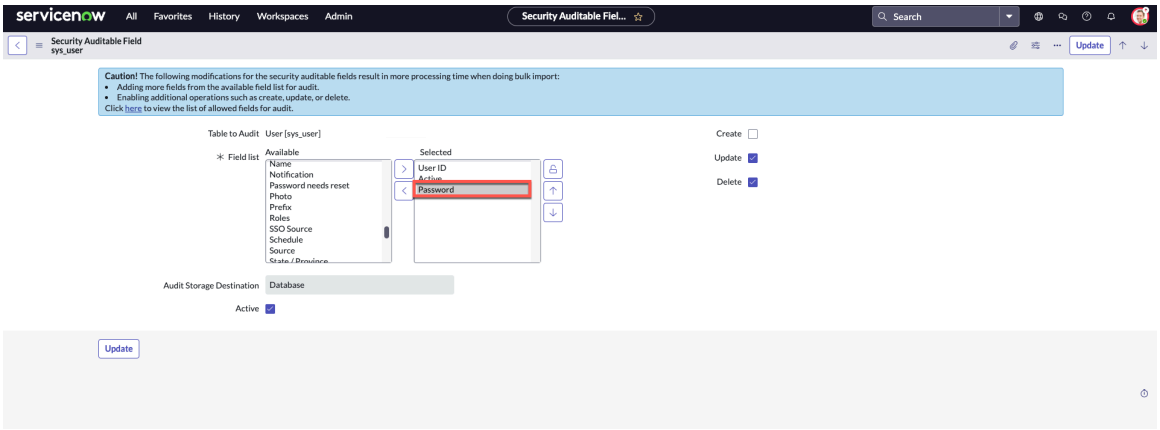
1. 移動先 すべて > システムセキュリティ > **ID** とアクセスの監査 > テーブルとフィールドを構成.
2. フィールドを監査するテーブルを選択します。

   たとえば、**sys_user** などです。



3. 監査するフィールドを追加します。

   たとえば、パスワードです。

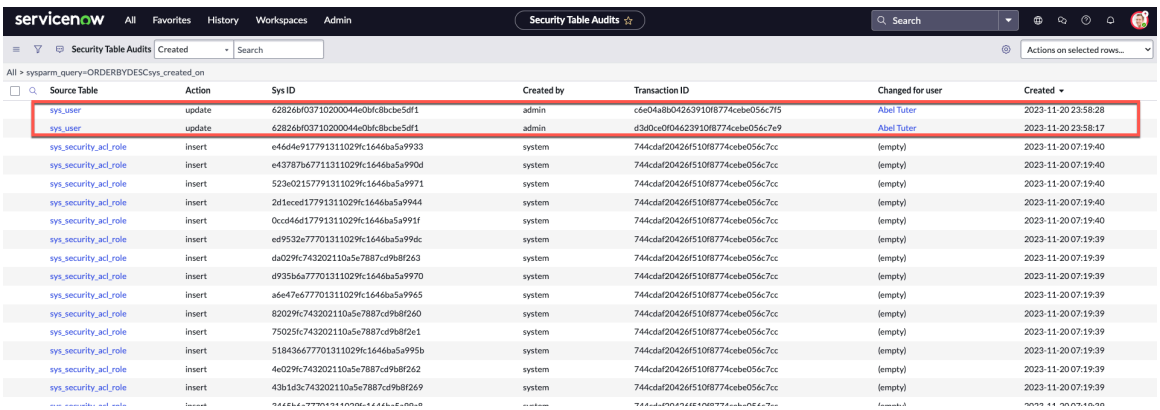


注: セキュリティ監査可能フィールドを次のように変更すると、一括インポート時の処理時間が長くなります。

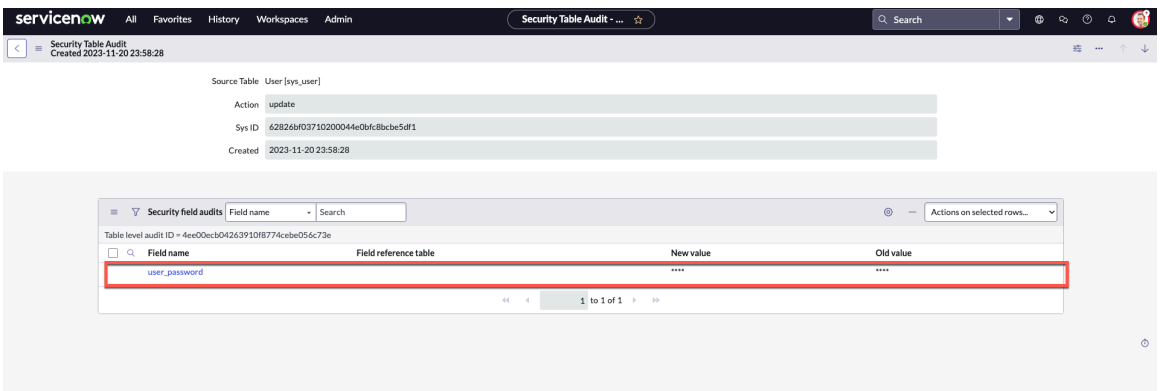
- 監査に使用可能なフィールド リストからさらにフィールドを追加します。
- 作成、更新、削除などの追加操作を有効にします。

4. レコードを更新します。

パスワードフィールドを変更すると、セキュリティテーブル監査に新しいレコードが追加されます。この例では、監査によりユーザー **Abel Tuter** の変更されたパスワードフィールドが表示されます。



作成したレコードを選択すると、変更内容の詳細が表示されます。



保存期間の構成

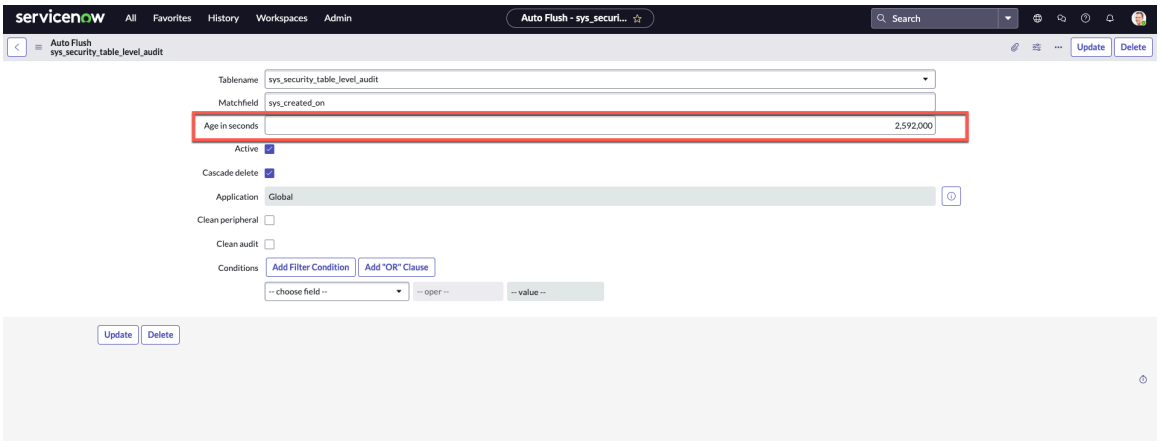
ID とアクセスの監査の保持期間を設定します。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > システムセキュリティ > **ID** とアクセスの監査 > 保存期間を構成.
2. フォームで経過時間を秒単位で変更できます。

   注: ID とアクセスの監査の最大日数は 30 日 (2,592,000 秒) です。



3. レコードを [保存] または [更新] します。

ID アクセスと監査でサポートされているフィールドとサポートされていないフィールド

監査でサポート対象のフィールドとサポート非対象のフィールドのリスト。

監査フィールドの検証では、セキュリティフィールド監査構成 (sys_sec_field_audit_config) テーブルの field_list で一部のフィールドが選択されなくなります。

監査でサポート対象のフィールドまたはサポート非対象のフィールド

テーブル	サポート対象のフィールドまたはサポート非対象のフィールド
すべてのテーブル	サポート非対象のフィールド： • 作成日時 (sys_created_on) • 作成者 [sys_created_by] • 更新者 [sys_updated_by] • 更新日 (sys_updated_on)
ユーザーにロールがある **(sys_user_has_role)**	サポート対象のフィールド： • ユーザー • ロール

監査でサポート対象のフィールドまたはサポート非対象のフィールド (続く)

テーブル	サポート対象のフィールドまたはサポート非対象のフィールド
	• 継承 • カウント
ユーザー **(sys_user)**	サポート非対象のフィールド： • 前回のログイン (last_login) • 前回のログイン時間 (last_login) • 前回のログインデバイス (last_login_device) • マルチファクター認証を有効にする (enable_multifactor_authn) • デフォルトの視点 (default_perspective) • カレンダー統合 (calendar_integration) • フェデレーション ID (federated_id) • パスワードのリセットが必要 (password_needs_reset) • 失敗した試行 (failed_attempts) • 前回のパスワード (last_password) • LDAP Server (ldap_server) • ロックアウト (locked_out) • 通知 (notification) • ロール (roles) • ドメイン (sys_domain) • ドメインパス (sys_domain_path) • 時間形式 (time_format) • ハッシュ化されたユーザー ID (hashed_user_id) • クラス名 (sys_class_name) • 変更回数 (sys_mod_count)

ID センター

ID ベースのリスクとセキュリティギャップを監視および管理し、最小限に抑えることができます。

探索

ID センターの機能とビジネス価値について説明します。

アクティブ化

ID センターをアクティブ化する方法を理解します。

アドミンの ID メトリクス

ユーザー向け ID センター

Access Analyzer に関するよくある質問の詳細を確認します。

自動翻訳

権限がどのように評価されるかを把握します。	

ID センターの概要

ID センターは、ユーザーの属性、アクセス、デバイス、ログイン履歴、セキュリティアクティビティなどのコレクションです。

ID センターは、ID ベースのリスクとセキュリティギャップを監視および管理し、最小限に抑える機能を提供します。

ID センターでは、ServiceNow の 1 つの場所で、ID ベースのリスクとセキュリティギャップを監視および管理し、最小限に抑えることができます。

ID センターを有効にするには、Identity Center (com.snc.identity_center) プラグインをインストールします。エンドユーザーは ID センターを使用して、ID センターでアクティブなセッション、ログイン履歴、および信頼できるデバイスの詳細を表示できます。詳細については、「ユーザー向け ID センター」を参照してください。

ID センターのアクティブ化

ID センターの場合、Identity Center (com.snc.identity_center) プラグインをインストールします。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > システムアプリケーション > 利用可能なすべてのアプリケーション > すべて.
2. フィルター基準と検索バーを使用して Identity Center (com.snc.identity_center) プラグインを検索します。

   名前または ID でプラグインを検索できます。プラグインが見つからない場合は、ServiceNow 担当者から要求する必要があります。
3. [インストール] を選択して、インストールプロセスを開始します。

   注: ドメインセパレーションと代理アドミンがインスタンスで有効になっている場合、管理ユーザーはグローバルドメインに含まれている必要があります。それ以外の場合、次のエラーが表示されます： 「別の操作が実行されているため、アプリケーションのインストールは利用できません： <プラグイン名> のプラグインの有効化 (Application installation is unavailable because another operation is running: Plugin Activation for <plugin name>)」

   インストールが完了するとメッセージが表示されます。プラグインとともにインストールされるコンポーネントの詳細については、「 アプリケーションとともにインストールされているコンポーネントの検索 」を参照してください。

ユーザー向け ID センター

ID センターを使用して、アクティブなセッション、ログイン履歴、および信頼できるデバイスの詳細を表示します。

ID センターは、ユーザーの属性、デバイス、ログイン履歴、セキュリティアクティビティなどのコレクションです。すべてのデータを 1 つのウィンドウで表示し、追加のセキュリティコントロールと通知機能を提供します。

ID センターを使用して、アクティブなセッション、ログイン履歴、および信頼できるデバイスの詳細を表示できます。

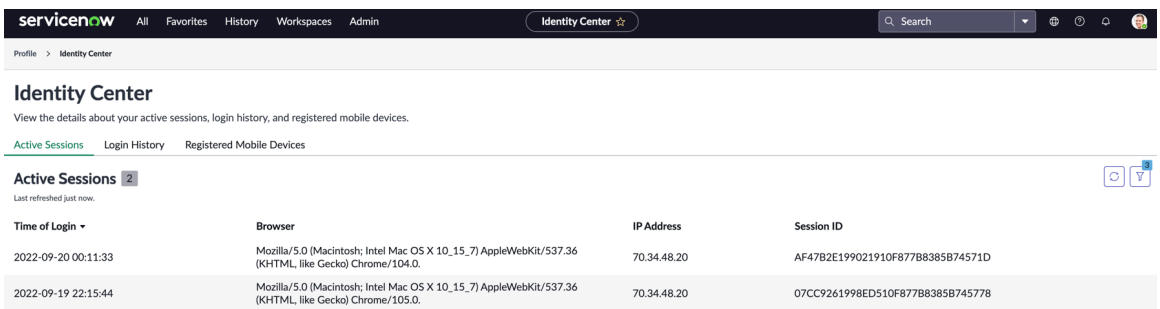
ユーザー向けの ID センターにアクセスするには、次のいずれかに移動します。

- ServiceNow AI Platformで、次の場所に移動します: すべて > セルフサービス > プロファイル をクリックし、[関連リンク] セクションの **[ID** センターの表示] を選択します。

  注: インスタンスヘッダーで自分のユーザー名を選択して、プロファイルにアクセスすることもできます。

- Now Support でプロファイルを選択し、ページの下部にある **[ID** センターを表示] を選択します。

[ID センター] ページが次のように表示されます。



ID センターには次のタブがあります。

- アクティブセッション
- ログイン履歴
- 登録済みモバイルデバイス

これらのタブを選択すると、ブラウザ、IP アドレス、セッション関連情報、ログイン情報、登録済みモバイルデバイスなどの詳細を表示できます。

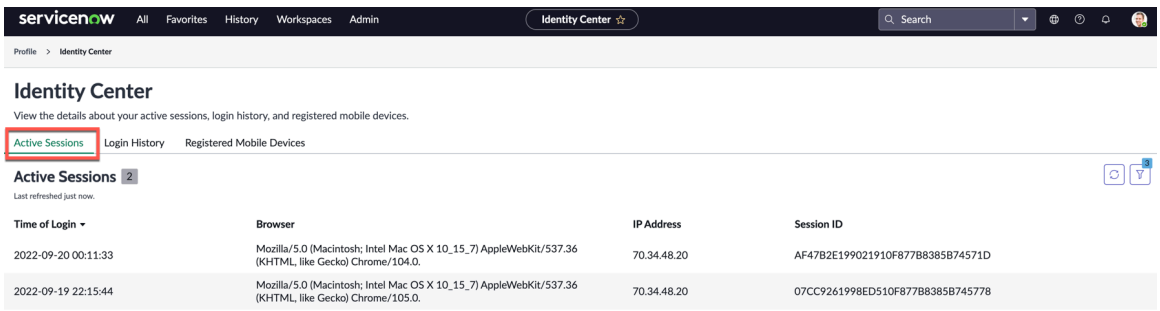
ID センターでのアクティブセッションの表示

ユーザーセッションに関する情報を表示します。

アクティブセッションは、さまざまなブラウザーまたはデバイスを使用して、現在の ServiceNow® インスタンスで開かれているセッションです。

ID センターの [アクティブセッション] タブは、ブラウザー、IP アドレス、およびセッション ID に基づいてセッションを識別するのに役立ちます。この情報を使用して、セッションの延長や終了などの必要なアクションを取ることができます。

この情報を使用すると、セッションが本物かどうか、またはセキュリティ上の問題が発生していないことを判断できます。



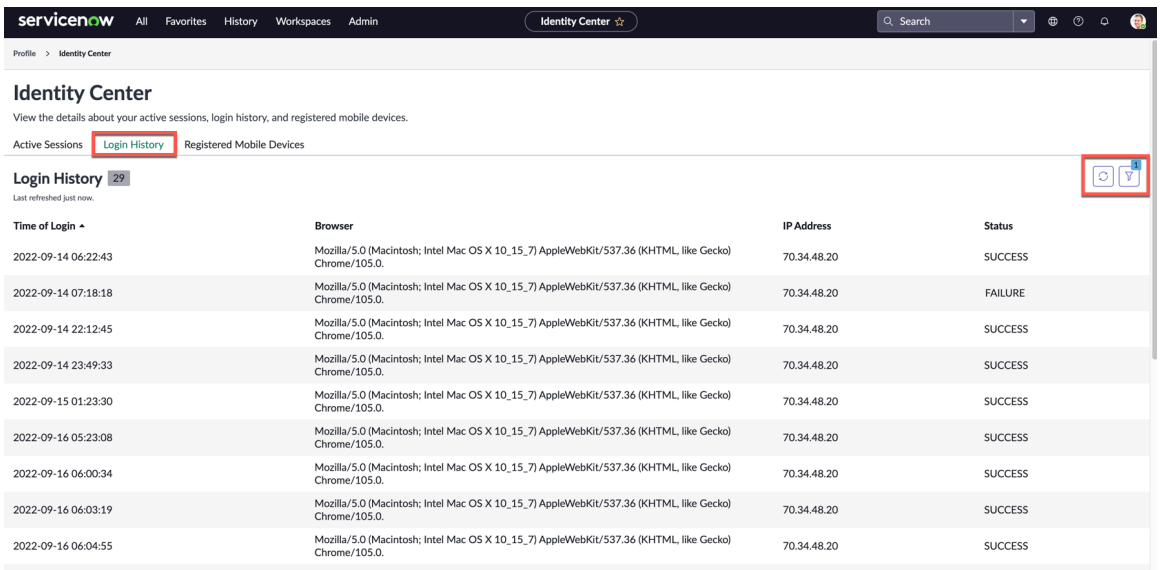
ID センターでのログイン履歴の表示

ログイン履歴の詳細を提供します。

ID センターの [ログイン履歴] タブは、ログイン情報とログインのステータスの確認に役立ちます。

フィルターを使用して、セキュリティ調査に役立つログインアクションを指定できます。フィルターは、アクティビティが本物か疑わしいかを判断し、その情報をアドミニストレーターに報告するのに役立ちます。

ログイン履歴のページネーションはデフォルトで 20 で、最大 100 に設定できます。



以下は、ログイン履歴からのその他の詳細情報の一部です。

- ログイン時刻は、モバイルデバイスを使用してインスタンスにアクセスするときに更新された時刻です。
- レコードは ID センターに 30 日間保存されます。

ID センターでの登録済みモバイルデバイスの表示

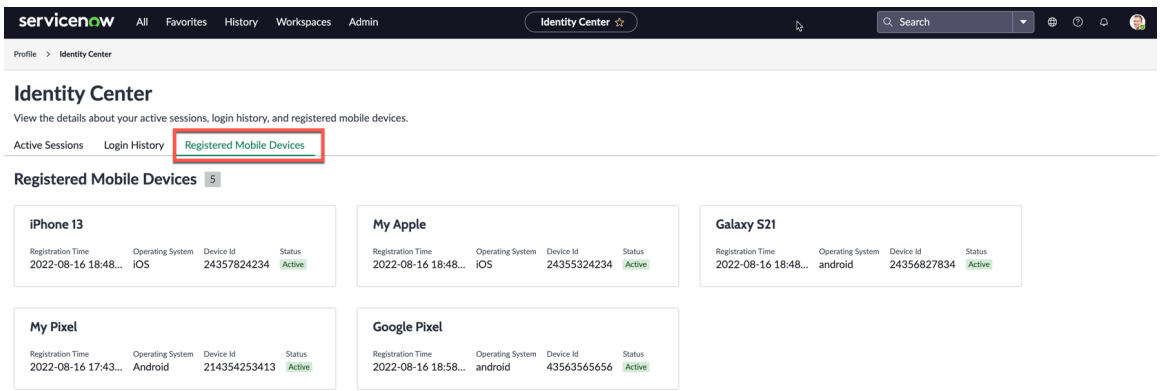
ServiceNow インスタンスの登録済みモバイルの詳細を示します。

ID センターの [登録済みモバイルデバイス (Registered Mobile Devices)] タブには、ServiceNow インスタンスへのアクセスに使用された登録済みデバイスの詳細が表示されます。

適応認証モジュールが有効になっていて、登録済みモバイルデバイスが登録されている場合、ID センターには登録済みデバイスの詳細が表示されます。

さらに、オペレーティングシステム、デバイス ID、デバイスのステータスなどのデバイスの詳細と、デバイスの登録時間も表示されます。

モバイルデバイスを登録するには、適応認証 (*com.snc.adaptive_authentication*) プラグインがインストールされ、信頼できるモバイルアプリ機能が有効になっていることを確認する必要があります。詳細については、「信頼できるモバイルアプリのアクティブ化」を参照してください。



アドミンの ID メトリクス

ServiceNow インスタンスのユーザー、特権ユーザー、アクティブなセッション、および統合されたアカウントの傾向を表示します。

アドミンの ID メトリクスには、次の傾向があります。

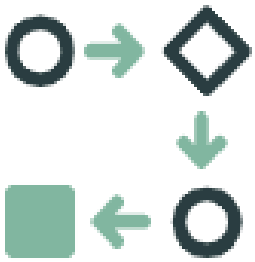
- ユーザー
- 特権ユーザー
- 統合または人間以外のアカウント
- アクティブセッション
- 非アクティブなセッション

詳細については、「セキュリティセンターのメトリクス」を参照してください。

マシン ID コンソール

ServiceNowとの統合に使用するサービスアカウントを管理します。

マシン ID は、システムやデータと対話するデジタル ID です。これらの ID は、タスクを自律的に実行するために使用されます。マシン ID コンソールは、これらの ID のセキュリティスコアを理解し、推奨事項を提供するのに役立ちます。

探索	アクティブ化	構成
		
マシン ID コンソールの機能とビジネス価値について説明します。	マシン ID コンソールをアクティブ化する方法を理解します。	マシン ID コンソールを使用してインバウンド統合を構成します。

マシン ID コンソールの詳細

ServiceNowとの統合に使用するサービスアカウントを管理します。

マシン ID コンソールを使用すると、保護された ServiceNow リソースにアクセスするために、さまざまなソフトウェアエンティティを識別、認証、および承認するために使用される人間以外の ID (NHI) を管理できます。アクセス権を取得するエンティティには、アプリケーション、ワークロード、API、ボット、自動化システムが含まれます。人間のアイデンティティとは異なり、人間以外のアイデンティティ(NHI)は同じ方法で管理されておらず、人間と直接関連付けられていません。彼らのIDと検証の方法は人間のユーザーとは異なり、標準的な人間のセキュリティ対策は適用されません。

https://player.vimeo.com/video/1101308622?h=b01d69f65a&badge=0&autopause=0&player_id=0&app_id=58479

マシン ID コンソールの概要ページに表示される機能は次のとおりです。

- マシン ID (統合) アカウントと高い権限のロールを持つアカウントの合計。
- 一意の API コール：過去 7 日間
- 使用された認証手法：過去 7 日間
- マシン ID セキュリティのスコアと ID に関連する結果。

関連トピック

- マシン ID コンソールのアクティブ化
- セキュリティの検知結果
- メトリクス
- マシン ID コンソール設定
- インバウンド統合

インバウンド統合

マシン ID コンソールのインバウンド統合を使用すると、ServiceNow API にアクセスするための外部アプリケーションを構成および管理できます。

ServiceNow のマシン ID コンソール内のインバウンド統合は、ServiceNow プラットフォームに接続する統合とアプリケーションの管理と構成に役立ちます。アプリケーション設定と API 認証情報を保存するための中央ハブとして機能し、ServiceNow と外部システム間の安全な通信を確保します。詳細については、「新しいインバウンド統合エクスペリエンス」を参照してください。

接続タイプ:

- OAuth - 認証コード許可:リソースを信頼する OAuth サーバーで直接認証して、リソースにアクセスします。詳細については、「OAuth 認証コード権限許可の構成」を参照してください。
- OAuth - クライアント認証情報の権限許可:サードパーティの OAuth クライアントから ServiceNow プラットフォームへのインバウンド統合に使用されます。詳細については、「OAuth クライアント認証情報の権限許可の構成」を参照してください。
- OAuth - JWT ベアラー権限許可:JSON Web トークン (JWT) 権限許可を使用してインスタンスを認証します。詳細については、「OAuth JSON Web トークンベアラー権限許可の構成」を参照してください。
- ID プロバイダーをサポートする OIDC によって発行されたサードパーティ ID トークン:受信 API 呼び出しを使用して、サードパーティ OIDC プロバイダーによって生成された ID トークンを受け入れるように OAuth OIDC プロバイダーを構成します。詳細については、「サードパーティ ID トークンの構成」を参照してください。
- OAuth - リソース所有者のパスワード認証情報権限許可:認証コードフローを使用して OAuth エンドポイントへのアクセスを許可します。詳細については、「OAuth リソース所有者のパスワード認証情報権限許可の構成」を参照してください。

  注: 認証コードフローの場合、ユーザーはローカルログイン、SSO、または MFA による認証を完了し、同意する必要があります。

詳細については、「」を参照してください。

関連トピック

- インバウンド統合
- OAuth 認証コード権限許可の構成
- OAuth JSON Web トークンベアラー権限許可の構成
- サードパーティ ID トークンの構成
- OAuth クライアント認証情報の権限許可の構成
- OAuth リソース所有者のパスワード認証情報権限許可の構成

自動翻訳

セキュリティの検知結果

マシン ID セキュリティスコアと結果を提供します。

マシン **ID** セキュリティスコアと結果

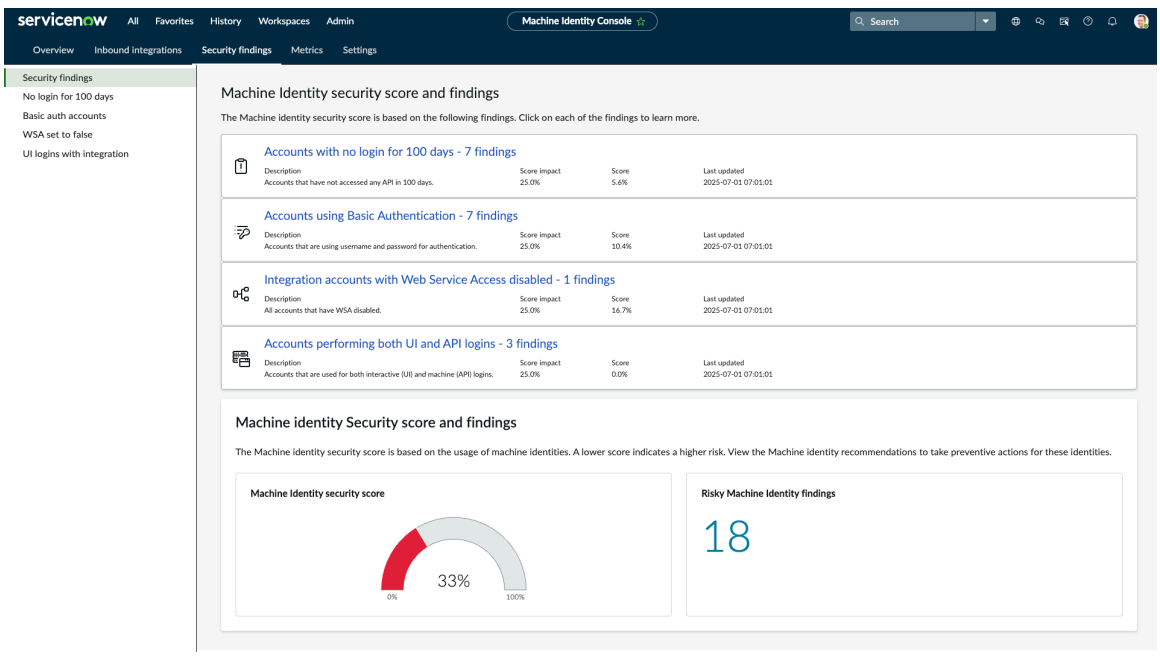
次のセキュリティ検索結果のセキュリティスコアが表示されます。各結果を選択すると、詳細を確認できます。

セキュリティの検知結果

結果	説明
100 日間ログインがないアカウント	100 日間どの API にもアクセスしていないアカウントに関する検索結果。
基本認証を使用しているアカウント	認証にユーザー名とパスワードを使用しているアカウントに関する検索結果。
Web サービスアクセスが無効になっている統合アカウント	WSA が無効になっているすべてのアカウントに関する検索結果。
UI および API の両方でログインを行っているアカウント	インタラクティブ (UI) ログインとマシン (API) ログインの両方に使用されるアカウントに関する検索結果。

マシン **ID** セキュリティスコアと結果

セキュリティスコアは、マシンIDの使用状況と方法に基づいています。スコアが低いほどリスクが高いことを示します。これらの ID に対して予防措置を講じるための推奨事項を表示できます。

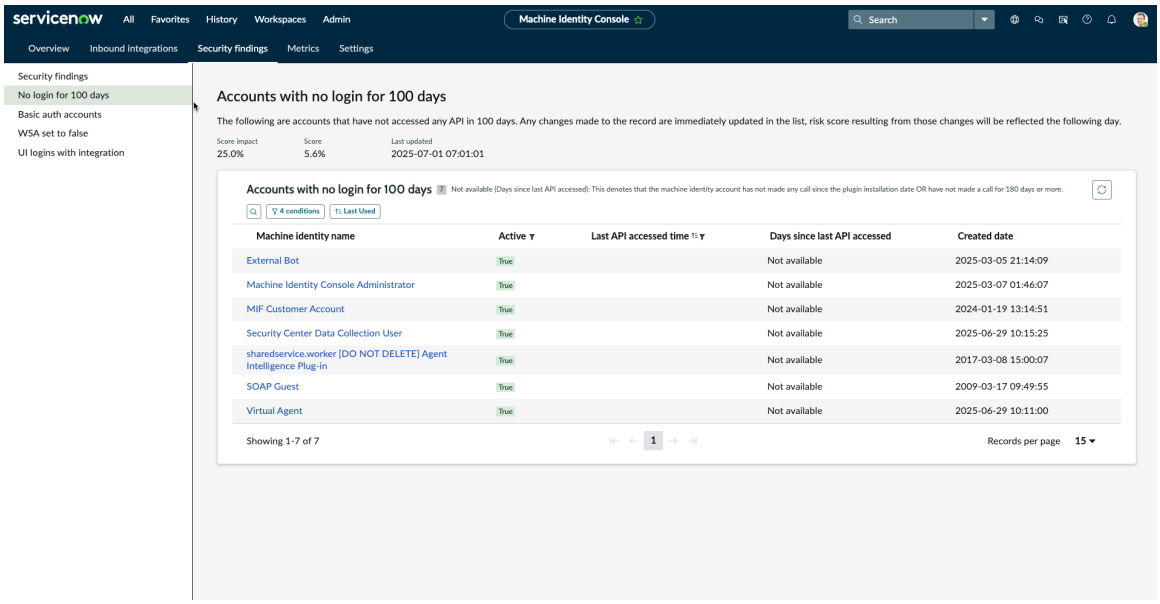


100 日間ログインがないアカウント

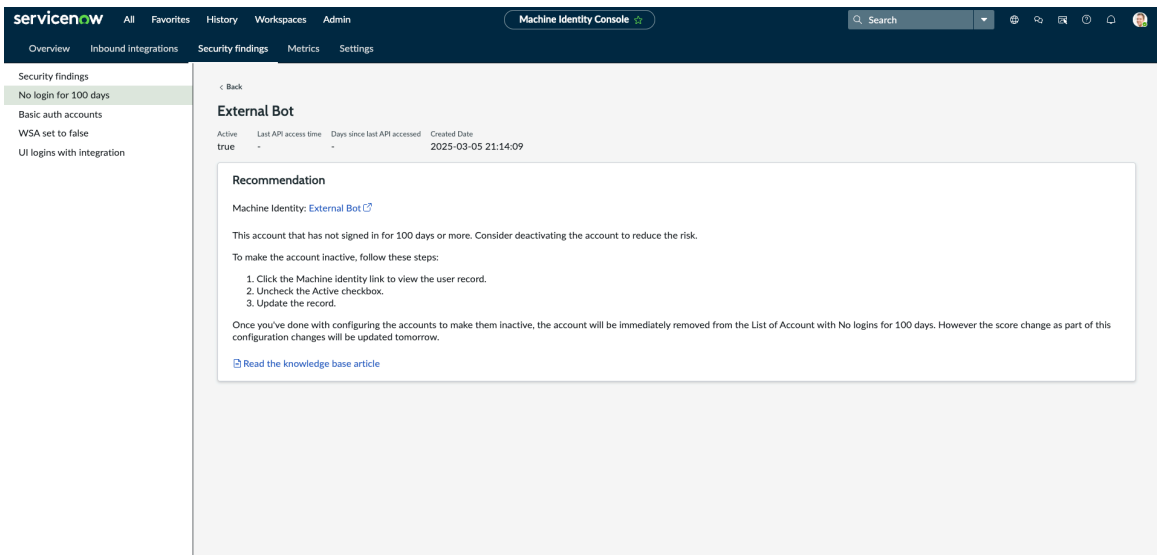
マシン ID コンソールのセキュリティ検索結果に、100 日間どの API にもアクセスしていないアカウントに関する検索結果が表示されます。

[100 日間ログインがないアカウント] には、100 日間どの API にもアクセスしていない人間以外の ID アカウントが表示されます。

注: このページに表示されているレコードに加えられた変更は、リスト内のすぐに更新され、それらの変更によって生じるリスクスコアは翌日反映されます。



マシン ID 名を選択すると、アカウントの詳細と、アカウントの適切なセキュリティ体制を維持するための推奨事項を確認できます。

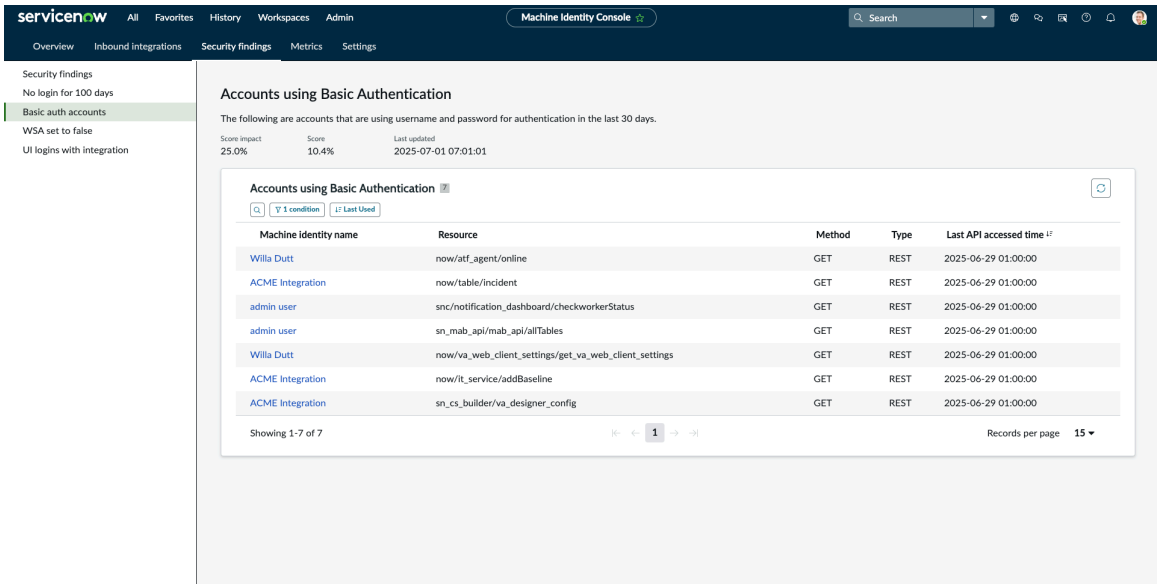


基本認証を使用しているアカウント

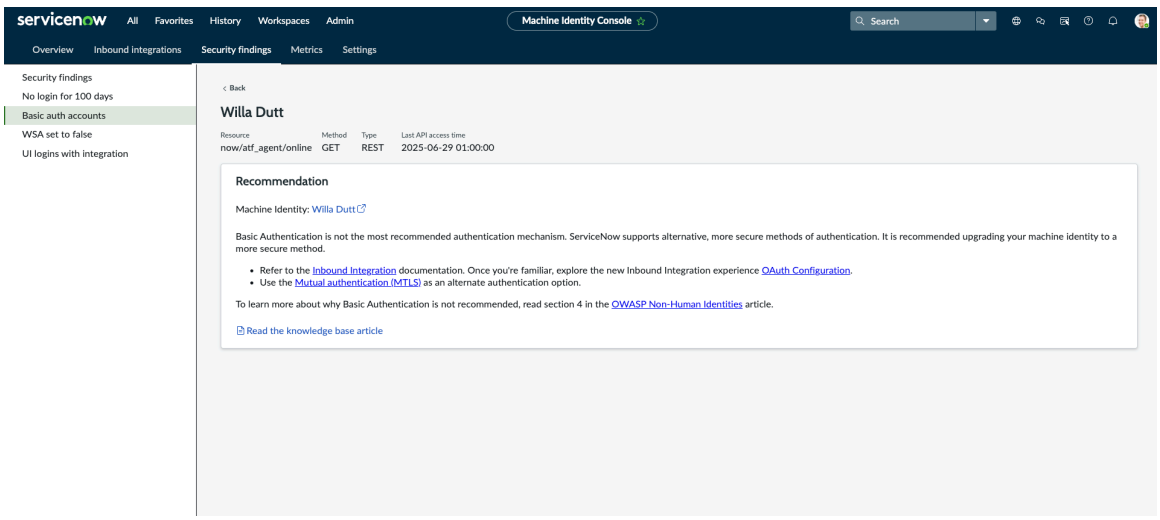
マシン ID コンソールのセキュリティ検索結果で、API の認証にベーシック認証のみを使用しているアカウントに関する検索結果を表示します。

[ベーシック認証を使用するアカウント] には、API のベーシック認証を使用して ServiceNow にログインする、人間以外の ID アカウントが表示されます。

注: ページに表示されるアカウントは、過去 30 日間に認証にユーザー名とパスワードを使用しているアカウントです。



マシン ID 名を選択すると、アカウントの詳細と、アカウントの適切なセキュリティ体制を維持するための推奨事項を確認できます。

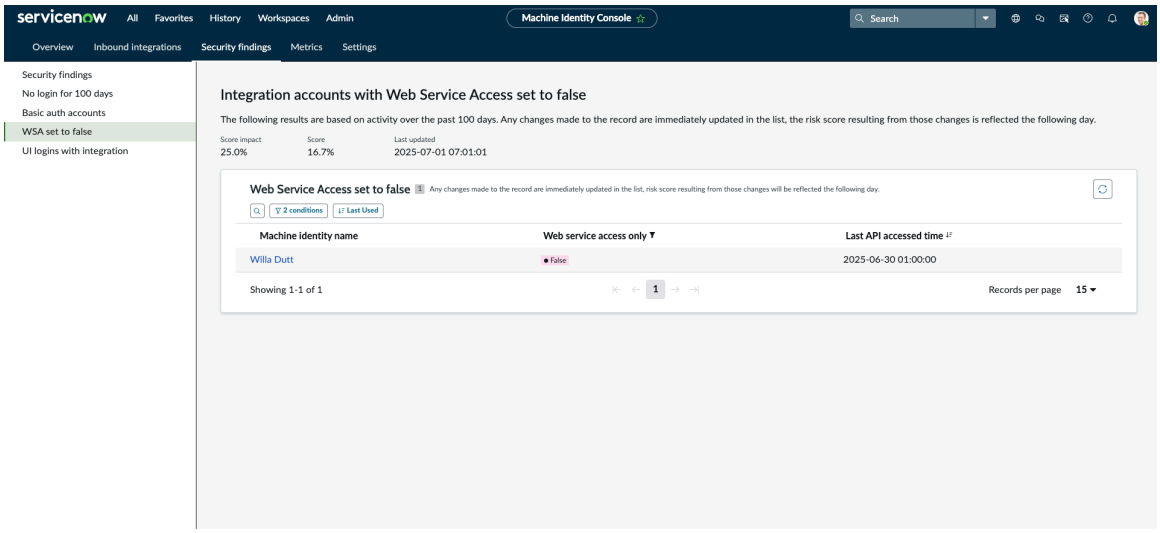


Web サービスアクセスが **false** に設定されている統合アカウント

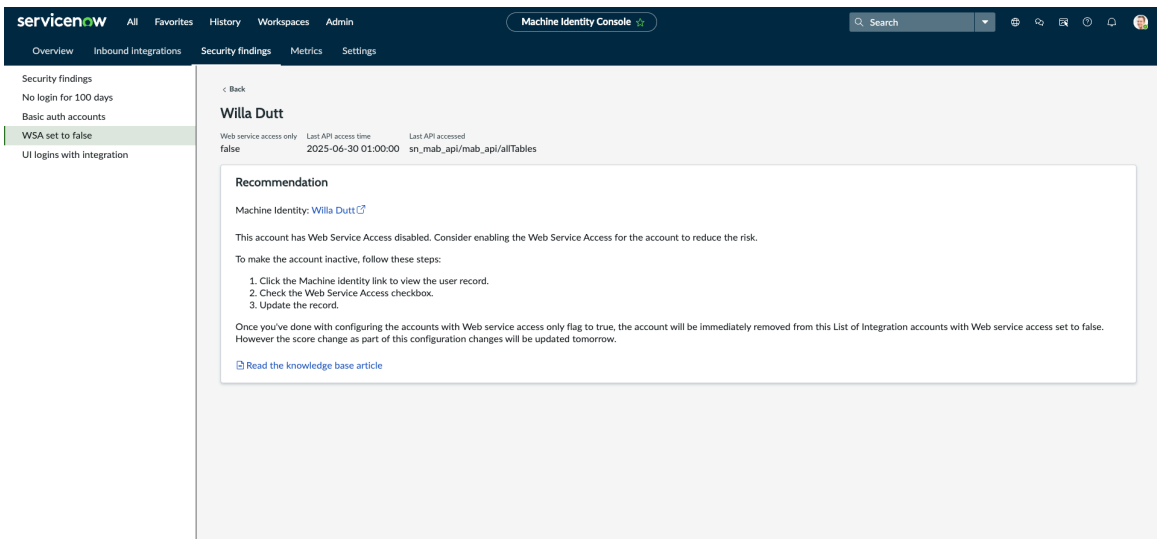
マシン ID コンソールの [セキュリティ検索結果] で、Web サービスアクセスを false に設定して ServiceNow を認証しているアカウントに関する検索結果を表示します。

Web サービスアクセスが false に設定されている統合アカウントには、Web サービスアクセスが false に設定されていない API を使用しているアカウントが表示されます。

注: このページに表示されているレコードに加えられた変更は、リスト内のすぐに更新され、それらの変更によって生じるリスクスコアは翌日反映されます。



マシン ID 名を選択すると、アカウントの詳細と、アカウントの適切なセキュリティ体制を維持するための推奨事項を確認できます。

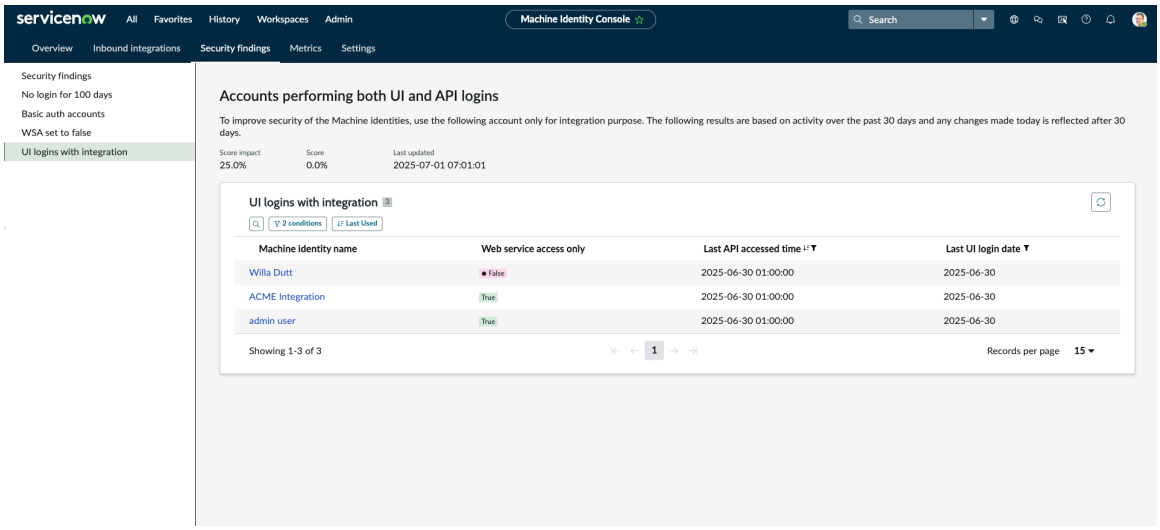


UI と **API** の両方のログインを実行するアカウント

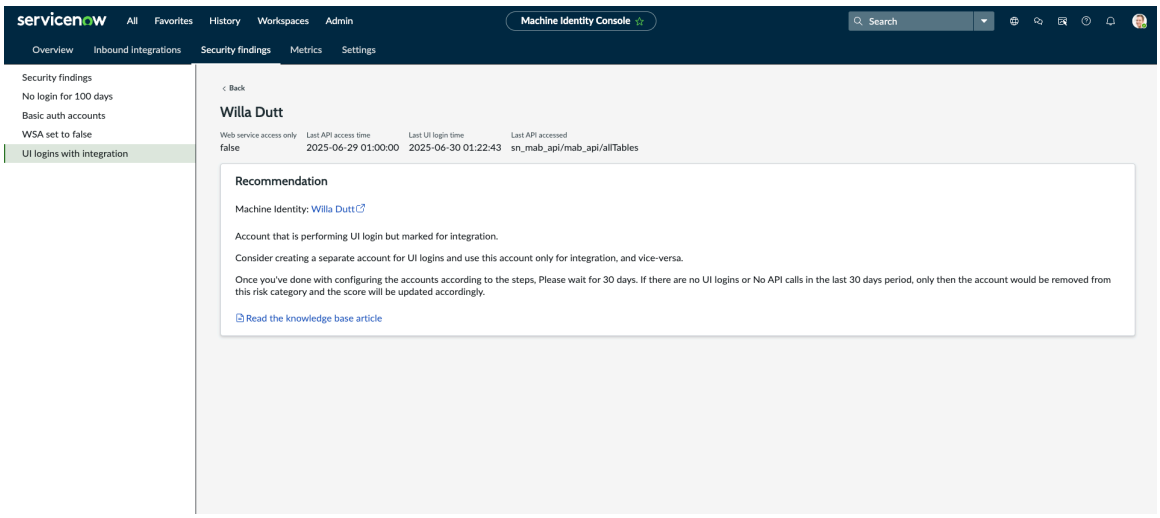
マシン ID コンソールの [セキュリティ] 検索結果の下に、UI および API ログインを実行しているアカウントに関する検索結果を表示します。

UI と API ログインの両方を実行するアカウントには、UI と API ログインを使用して ServiceNow への認証を行っているアカウントが表示されます。

> **注:** ページに表示されるアカウントは過去 30 日間のアクティビティに基づいており、今日行われた変更は 30 日後に反映されます。



マシン ID 名を選択すると、アカウントの詳細と、アカウントの適切なセキュリティ体制を維持するための推奨事項を確認できます。



メトリクス

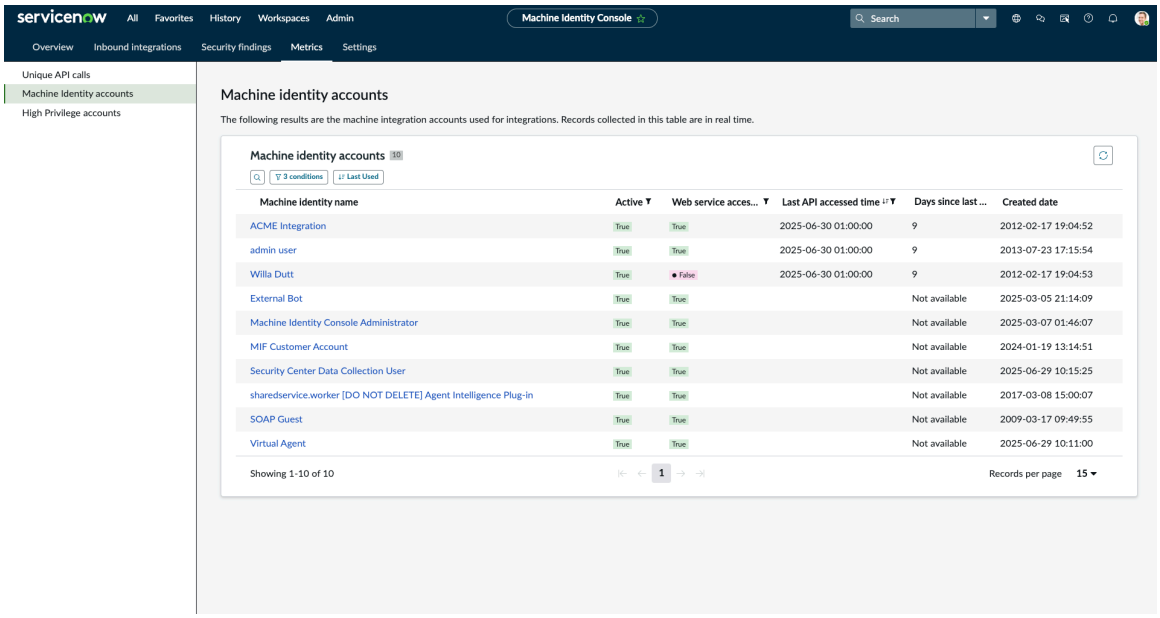
マシン ID のメトリクス。

過去 **7** 日間の一意の **API** コール

次の結果は、過去 7 日間の一意の API 呼び出しです。このテーブルに収集されるレコードはリアルタイムです。

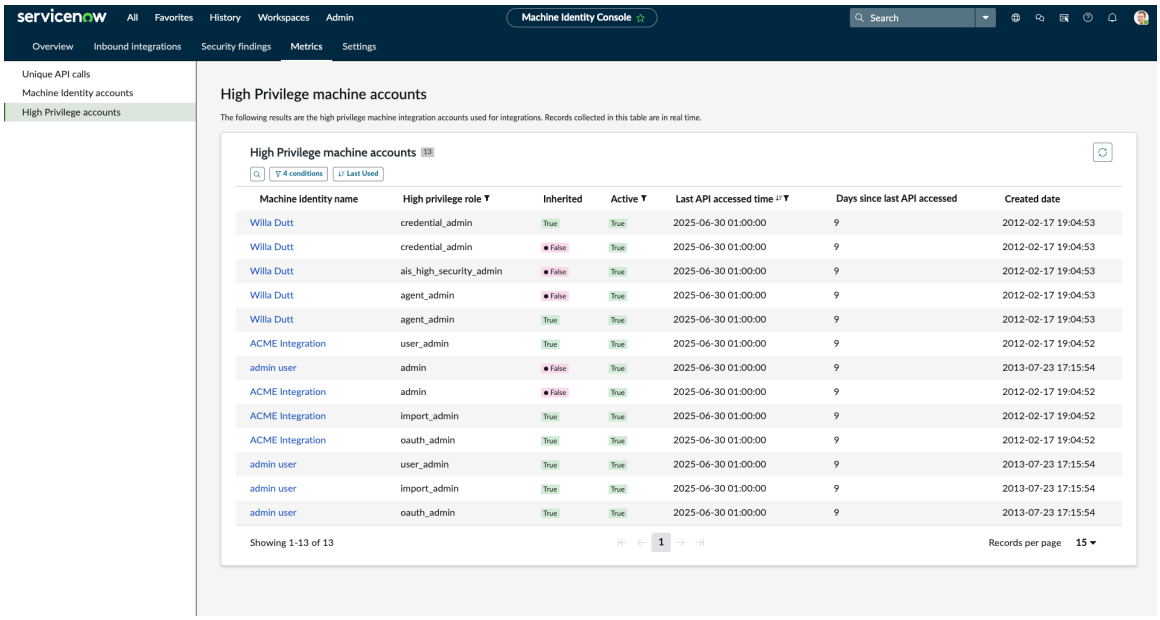
マシン **ID** アカウント

以下は、統合に使用されるマシン統合アカウントの結果です。このテーブルに収集されるレコードはリアルタイムです。



高権限マシンアカウント

以下は、統合に使用される高権限のマシン統合アカウントの結果です。このテーブルに収集されるレコードはリアルタイムです。



マシン **ID** コンソール設定

マシン ID のセキュリティスコア設定を構成します。

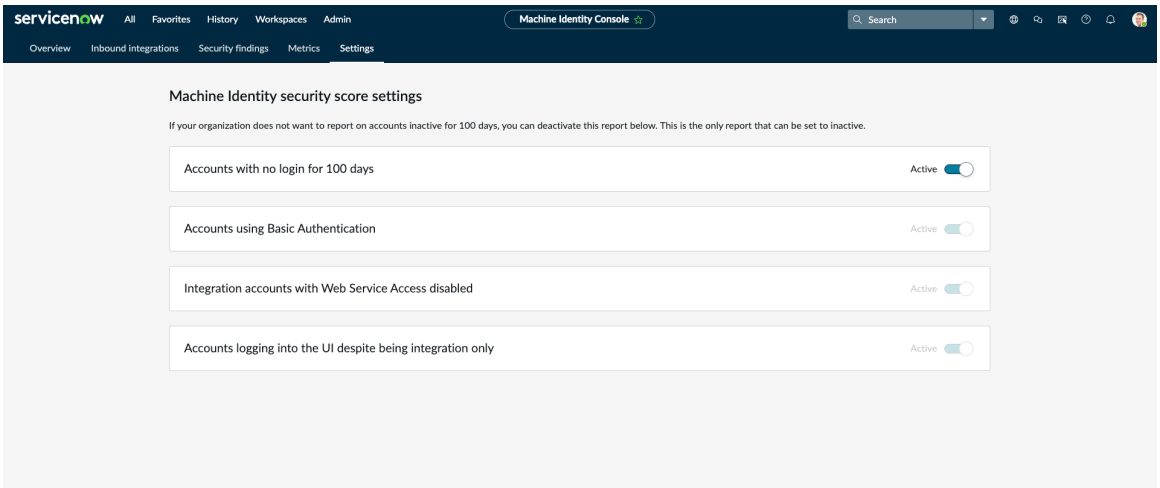
始める前に
必要なロール:mi_admin

次のタスクは、非アクティブなアカウントの処理方法に関する組織に固有のものです。

> **注:** 組織が 100 日間非アクティブなアカウントに関するレポートを希望しない場合は、以下でこのレポートを非アクティブ化できます。これは、非アクティブに設定できる唯一のレポートです。

手順

1. 移動先 すべて > マシン **ID** コンソール > 設定.
2. 要件に基づいて設定を選択します。
   - **100** 日間ログインがないアカウント
   - 基本認証を使用しているアカウント
   - **Web** サービスアクセスが無効になっている統合アカウント
   - 統合専用であるにもかかわらず **UI** にログインしているアカウント



マシン **ID** コンソールのアクティブ化

マシン **ID** 管理 com.glide.identity.machine_identity_managementを有効にして、ServiceNowとの統合に使用されるサービスアカウントを管理します。

始める前に
必要なロール:mi_admin

手順

1. 移動先 すべて > システムアプリケーション > 利用可能なすべてのアプリケーション > すべて.
2. フィルター基準と検索バーを使用して 、マシン **ID** 管理 (com.glide.identity.machine_identity_management) プラグインを検索します。

   名前または ID でプラグインを検索できます。プラグインが見つからない場合は、ServiceNow 担当者から要求する必要があります。

3. [インストール] を選択して、インストールプロセスを開始します。

   注: ドメインセパレーションと代理アドミンがインスタンスで有効になっている場合、管理ユーザーはグローバルドメインに含まれている必要があります。それ以外の場合、次のエラーが表示されます： 「別の操作が実行されているため、アプリケーションのインストールは利用できません： <プラグイン名> のプラグインの有効化 (Application installation is unavailable because another operation is running: Plugin Activation for <plugin name>)」

   インストールが完了するとメッセージが表示されます。プラグインとともにインストールされるコンポーネントの詳細については、「 アプリケーションとともにインストールされているコンポーネントの検索 」を参照してください。

マシン ID コンソールの使用

マシン ID コンソールを使用して、ServiceNow との統合に使用するサービスアカウントを管理します。

始める前に
必要なロール:mi_admin

Machine Identity Management com.glide.identity.machine_identity_management プラグインを有効にします。詳細については、「マシン ID コンソールのアクティブ化」を参照してください。

手順

1. 移動先 すべて > マシン **ID** コンソール > マシン **ID** コンソール.
   マシン ID コンソールには、次のタブがあります。
   - 概要:マシン ID アカウント全体とそのメトリクスを表示します
   - インバウンド統合:API のインバウンド統合を構成します。
   - セキュリティの検知結果:セキュリティスコアがさまざまなセキュリティ検索結果に基づいていることを表示します。
   - メトリクス:マシン ID アカウントのさまざまなメトリクスを表示します
   - 設定: マシン ID コンソールを構成します。
2. セキュリティ要件に基づいて適切なタブを選択します。

クロスドメイン ID 管理システム (SCIM)

クロスドメイン ID 管理システム (SCIM) API は、SCIM プロトコルを使用してユーザーとグループの操作を作成、読み取り、更新、および削除するためのエンドポイントを提供します。

SCIM プロバイダー

SCIM プロバイダーは、レコードの作成、更新、削除など、IdP の ID に対して行われた変更を同期します。

SCIM クライアント

SCIM クライアントは、SCIM に準拠する REST 要求をサポートするシステムで ID リソースを作成、更新、および削除する際に使用します。

SCIM プロトコルは、HTTP (RFC7230) 標準を基礎とするアプリケーションレベルの HTTP ベースプロトコルです。この API はユーザーやグループなどの ID データのプロビジョニングと管理に使用します。エンタープライズからクラウドへのサービスプロバイダーやクラウド間のシナリオなど、Web およびクロスドメイン環境で API を使用します。

この API にアクセスするには、SCIM v2 - ServiceNow® Cross-domain Identity Management (com.snc.integration.scim2) プラグインを有効にする必要があります。

SCIM API の詳細については、「クロスドメイン ID 管理システム (SCIM) API 」を参照してください。

SCIM プロバイダー

サービスプロバイダーは、SCIM API を使用してユーザーとグループをプロビジョニングします。

探索

SCIM プロバイダーについて学習します。

アクティブ化

SCIM をアクティブ化します。

SCIM のカスタマイズ

SCIM をカスタマイズする方法の詳細を取得します。

ソース定義

SCIM のソース定義について詳細を確認します。

自動翻訳

SCIM プロバイダーの概要

サービスプロバイダーは、SCIM API を使用してユーザーとグループをプロビジョニングします。

SCIM プロバイダーとして、ServiceNow スキーマは、ユーザーとグループをプロビジョニングする SCIM API をサポートしています。

SCIM プロバイダーは、レコードの作成、更新、削除など、IdP の ID に対して行われた変更を同期します。これらの変更は、SCIM プロトコルに従って自動的にプロバイダーに同期されます。また、IdP ではプロバイダーから ID を読み取って IdP ディレクトリーに追加することができます。その後、セキュリティの脆弱性を作成する可能性のあるプロバイダーの誤った値を検出することが可能です。この同期によって、エンドユーザーが最新のプロファイルと権限を使用して、自身が割り当てられたアプリケーションにシームレスにアクセスすることが可能になります。

SCIM プロバイダーの設定

SCIM プロバイダーを設定するには、次のタスクを実行します。

- **SCIM v2 - ServiceNow Cross-domain Identity Management** プラグインをアクティブ化します。詳細については、「SCIM プラグインのアクティブ化」を参照してください。
- SCIM に必要な他のプラグインをアクティブ化します。
  - OAuth 2.0
  - REST API Provider
  - REST API Access Policy
- SCIM サービスの一部として scim_admin ロールを追加します。☒

⚠ 警告: このロールは慎重に付与してください。scim_admin ロールはユーザーに admin ロールを付与することと同じで、scmin_admin は個人識別可能情報 (PII) を追加または更新することができます。

テーブル

sys_user と sys_group の 2 つのテーブルには、既存の ServiceNow テーブルにマッピングされない SCIM 属性が含まれています。テーブルの詳細については、「SCIM 固有のテーブル」を参照してください。

SCIM プラグインのアクティブ化

SCIM をアクティブ化するには、SCIM v2 - ServiceNow Cross-domain Identity Management (com.snc.integration.scim2) プラグインをインストールします。

始める前に

必要なロール：admin

手順

1. 移動先 すべて > システムアプリケーション > 利用可能なすべてのアプリケーション > すべて.
2. フィルター基準と検索バーを使用して、SCIM v2 - ServiceNow Cross-domain Identity Management (com.snc.integration.scim2) プラグインを検索します。

   名前または ID でプラグインを検索できます。プラグインが見つからない場合は、ServiceNow 担当者から要求する必要があります。
3. [インストール] を選択して、インストールプロセスを開始します。

自動翻訳

> **注:** ドメインセパレーションと代理アドミンがインスタンスで有効になっている場合、管理ユーザーはグローバルドメインに含まれている必要があります。それ以外の場合、次のエラーが表示されます： 「別の操作が実行されているため、アプリケーションのインストールは利用できません： <プラグイン名> のプラグインの有効化 (Application installation is unavailable because another operation is running: Plugin Activation for <plugin name>)」

インストールが完了するとメッセージが表示されます。プラグインとともにインストールされるコンポーネントの詳細については、「 アプリケーションとともにインストールされているコンポーネントの検索 」を参照してください。

チュートリアル：プロバイダーでユーザープロビジョニングするための **SCIM** の構成

プロバイダーのプロビジョニングサービスを使用して、ユーザーとグループが ServiceNow に自動的にプロビジョニングおよびプロビジョニング解除されるよう SCIM を構成します。

始める前に

SCIM プラグインをインストールします。

必要なロール：scim_admin

> **警告:** このロールは慎重に付与してください。scim_admin ロールはユーザーにアドミンロールを付与することと同じで、scmin_admin は個人識別可能情報 (PII) を追加または更新することができます。

このタスクについて

次の認証方法で SCIM を使用してユーザーをプロビジョニングできます。

- ベーシック認証を使用するユーザーのプロビジョニング
- OAuth を使用するユーザーのプロビジョニング

ベーシック認証を使用するユーザーのプロビジョニング

ベーシック認証でプロバイダーのプロビジョニングサービスを使用して、ユーザーとグループが ServiceNow に自動的にプロビジョニングおよびプロビジョニング解除されるよう SCIM を構成します。

始める前に

必要なロール：scim_admin

> **警告:** このロールは慎重に付与してください。scim_admin ロールはユーザーにアドミンロールを付与することと同じで、scmin_admin は個人識別可能情報 (PII) を追加または更新することができます。

SCIM プラグインをアクティブ化する必要があります。

手順

1. 移動先 すべて > システム **Web** サービス > **REST API** アクセスポリシー REST API アクセスポリシーの詳細を確認します。
2. API アクセスポリシーページで、**[SCIM API** ポリシー **(SCIM API Policy)]** レコードをクリックします。
3. [認証プロファイル] セクションで **[SCIM API** 基本認証 **(SCIM API Basic Auth)]** レコードが利用可能であることを確認します。
4. [タイプ] として [基本認証] を選択します。

自動翻訳

5. プロバイダー側で必要な構成を作成します。
6. 接続をテストして、プロバイダーが ServiceNow に接続できることを確認します。

> **注:** 接続に失敗した場合は、ServiceNow アカウントにアドミン権限があることを確認して、再試行しください。

OAuth を使用するユーザーのプロビジョニング

OAuth でプロバイダーのプロビジョニングサービスを使用して、ユーザーとグループが ServiceNow に自動的にプロビジョニングおよびプロビジョニング解除されるよう SCIM のプロバイダーを構成します。

始める前に
必要なロール：scim_admin

> **警告:** このロールは慎重に付与してください。scim_admin ロールはユーザーにアドミンロールを付与することと同じで、scmin_admin は個人識別可能情報 (PII) を追加または更新することができます。

SCIM プラグインをアクティブ化する必要があります。

手順

1. 移動先 すべて > システム **OAuth** > アプリケーションレジストリー.
2. [アプリケーションレジストリ] ページで、**[SCIM API]** レコードをクリックします。
3. **[SCIM API]** レコードの詳細を確認します。

   これらの詳細は、Azure AD で ServiceNow アプリケーションを構成するときに指定する必要があります。
4. 移動先 すべて > システム **Web** サービス > **REST API** アクセスポリシー REST API アクセスポリシーの詳細を確認します。
5. API アクセスポリシーページで、**[SCIM API** ポリシー **(SCIM API Policy)]** レコードをクリックします。
6. [認証プロファイル] セクションで **[SCIMAPIOAuthOnly]** レコードが利用可能であることを確認します。
7. 以前にアプリケーションレジストリとして構成または検証された **[SCIM API]** レコードで **[OAuth** エンティティ] フィールドが指定されているかどうかを確認します。
8. プロバイダー側で必要な構成を作成します。
9. 接続をテストして、プロバイダーが ServiceNow に接続できることを確認します。

> **注:** 接続に失敗した場合は、ServiceNow アカウントにアドミン権限があることを確認して、再試行しください。

SCIM のトラブルシューティング

SCIM と統合する一般的なエラーシナリオ。

無効な REST API URL

アクション：有効な API URL を入力します。**[REST API** エクスプローラー] で REST API URL をクロスチェックできます。

ServiceNow インスタンスにリダイレクト URL が設定されていない

アクション：ServiceNow に SCIM OAuth エンティティの有効なリダイレクト URL を入力します。ServiceNow で OAuth エンティティを構成するときにリダイレクト URL を入力します。

リダイレクト URL が要求と異なる

アクション：[認証エンドポイント **(Authorization Endpoint)]** で指定された **[redirect_url]** は、ServiceNow で構成された OAuth エンティティと同じである必要があります。

注:

このエラーは、Azure の「認証エンドポイント」と ServiceNow の「リダイレクト URL」の間に不一致がある場合に発生します。

無効なクライアントシークレットが渡された

アクション：「クライアントシークレット」に入力された値は、ServiceNow で構成された OAuth エンティティと同じである必要があります。

Azure の「認証エンドポイント」で無効な「ClientId」が渡された

アクション：「認証エンドポイント」の「client_id」パラメーターに入力された値は、ServiceNow で構成された OAuth エンティティと同じである必要があります。

SCIM のカスタマイズ

ID 管理用の SCIM プロトコルをカスタマイズします。

SCIM をカスタマイズすると、次のことができるようになります。

- 動的な拡張スキーマを生成することで、sys_user および sys_user_group テーブルのカスタムフィールドをサポートする。
- デフォルトの SCIM マッピングを上書きする機能を実現する。

SCIM アドミニストレーターは、ユーザーリソースとグループリソースのカスタム拡張スキーマを定義できます。カスタム拡張スキーマで定義された属性は、sys_user または sys_user_group テーブルのフィールドにマッピングできます。

SCIM のカスタマイズの設定

SCIM のカスタマイズでは、次のタスクを実行する必要があります。

- SCIM 拡張スキーマテーブルでユーザーとグループのカスタム拡張スキーマを定義します。詳細については、「SCIM 拡張スキーマの作成」を参照してください。
- カスタムスキーマ属性の ETL 定義にエンティティを作成します。エンティティは、sys_user または sys_user_group 属性のいずれかでマッピングされたターゲットテーブルに対して作成されます。詳細については、「SCIM ETL 定義の作成」を参照してください。
- これら 2 つのエンティティ間に RTE マッピングを作成します。詳細については、「SCIM ETL 定義の作成」の手順 5 を参照してください。
- SCIM API 要求ペイロードのデータを含むカスタムスキーマ属性を送信します。

SCIM API が、定義されたマッピングを使用して RTE エンジンを呼び出します。マッピングで定義されたターゲットテーブルの各フィールドにデータが格納されます。

SCIM のカスタマイズのプロパティとスキーマ

SCIM のカスタマイズには、次のようなプロパティ、サポート済みスキーマ、およびサポートされていないスキーマが含まれています。

プロパティ

SCIM のカスタマイズでは、次のシステムプロパティが追加されます。

プロパティ

名前	説明
com.snc.integration.scim2.max.member.count	SCIM の最大メンバー数。
com.snc.integration.scim2.resolve.external.id.conflict	外部 ID の SCIM フィルターで複数のリソースが見つかった場合は、要求元クライアントのソース定義に基づいて SCIM リソースを解決します。 注: 前提条件として、複数のソースからプロビジョニングを行う場合は、[SCIM ソース定義] テーブルでソースを定義する必要があります。ソースが定義されていない場合、またはこのプロパティが非アクティブな場合は、一致するすべてのリソースが外部 ID のフィルター応答で返されます。
com.snc.integration.scim2.user.etl.definition.id	SCIM ユーザー ETL 定義 ID。
com.snc.integration.scim2.group.etl.definition.id	SCIM グループ ETL 定義 ID。
com.snc.integration.scim2.rte.verbose.logging.enabled	SCIM ユーザーおよびグループ RTE 定義の詳細なログ記録を有効にします。
com.snc.integration.scim2.string.field.length.validation.enabled	フィールドの長さの検証。このプロパティは、フィールドを切り捨てて保存するのではなく、検証を有効にするものです。
com.snc.integration.scim2.provider.customization.script.id	SCIM 応答をカスタマイズするためのスクリプトインクルードの ID。

サポート済みスキーマ

SCIM のカスタマイズでは、次のサポート済みスキーマが追加されます。

サポート済みスキーマ

スキーマ	説明	プリフィックス	例
urn:ietf:params:scim:schemas:core:2.0:User	ユーザーのコア属性が含まれます。	none	name.middleName
urn:ietf:params:scim:schemas:extension:servicenow:2.0:User	ServiceNow に関連する属性が含まれます。	servicenow	servicenow.manager.value
urn:ietf:params:scim:schemas:extension:custom:2.0:User	コア拡張または ServiceNow 拡張スキーマの一部として	custom	custom.socialId

サポート済みスキーマ (続く)

スキーマ	説明	プリフィックス	例
	マッピングされていないカスタム属性が含まれます。		

サポートされていないスキーマ

urn:ietf:params:scim:schemas:extension:enterprise:2.0:User：ビジネスや企業に所属する、またはその組織を代表するユーザーを表す際に一般的に使用される属性が含まれます。

注: Enterprise スキーマは有効なスキーマですが、その属性はどのテーブルにもマッピングされます。データベースの永続性がサポートされていないため、Enterprise スキーマが要求本文に含まれている場合でもエラーは表示されません。

SCIM 拡張スキーマの作成

コアスキーマまたは ServiceNow 拡張スキーマの一部としてマッピングされていないフィールドにマッピングするカスタム属性を作成します。

始める前に
必要なロール：scim_admin

警告: このロールは慎重に付与してください。scim_admin ロールはユーザーにアドミンロールを付与することと同じで、scmin_admin ロールは、ビジネスロジックや ACL 保護をバイパスできる新たなレコードをテーブルに挿入することができます。

手順

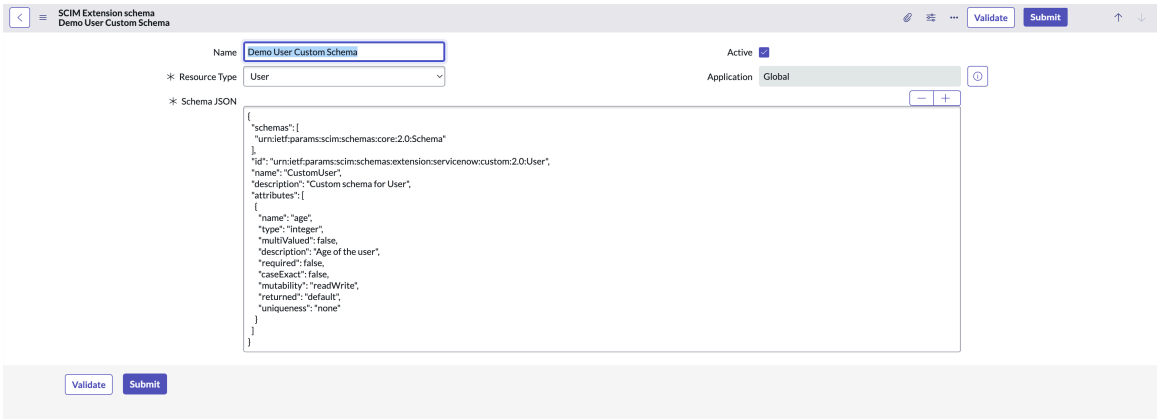
1. 移動先 すべて > **SCIM** > **SCIM** 拡張スキーマ.
2. [**New**] をクリックします。
3. フォームのフィールドに入力します。

注: [リソースタイプ] フィールドにマッピングできる拡張スキーマは 1 つだけです。たとえば、リソースタイプとしてのユーザーは、ユーザー拡張スキーマにマッピングできます。

SCIM 拡張スキーマ

フィールド	説明
名前	拡張スキーマの名前。
アクティブ	スキーマをアクティブ化するオプション。レコードをカスタム拡張スキーマと見なす必要がある場合は、このフィールドを選択します。ユーザータイプとグループタイプのどちらに対しても、特定のリソースタイプに対して一度にアクティブ化できるカスタム拡張スキーマレコードは 1 つだけです。
リソースタイプ	拡張スキーマにマッピングする必要があるリソースタイプ。オプションは次のとおりです。 ○ ユーザー ○ グループ

フィールド	説明
アプリケーション	このレコードのアプリケーションスコープ。
スキーマ JSON	JSON スキーマ内の詳細情報。属性を含む拡張スキーマの定義の詳細については、Datatracker を参照してください。



4. [検証] をクリックして属性を検証します。
5. [送信] をクリックします。

結果
ユーザーまたはグループのリソースタイプに関連するカスタム属性を持つ拡張スキーマが作成されます。SCIM ETL 定義を使用して、sys_user および sys_user_group テーブルの拡張スキーマに基づいてリソースをマッピングします。詳細については、「SCIM ETL 定義の作成」を参照してください。

SCIM ETL 定義の作成

SCIM ETL 定義を使用して、カスタム属性を sys_user または sys_user_group テーブルにマッピングします。

始める前に
必要なロール：scim_admin

警告: このロールは慎重に付与してください。scim_admin ロールはユーザーにアドミンロールを付与することと同じで、scmin_admin ロールは、ビジネスロジックや ACL 保護をバイパスできる新たなレコードをテーブルに挿入することができます。

注:

- SCIM グループと SCIM ユーザー ETL 定義は、リソースマッピングのベースシステムの一部です。同じリソースマッピングを使用して、必要に応じて基準を変更することも、新しいリソースマッピングを作成することもできます。
- SCIM マッピングでは、RTE を介した [*] フィールドはサポートされていません。

手順

1. 「抽出変換ロード (ETL) 定義を作成する」の指示に従います。
2. 新しく作成されたレコードを開いて詳細を表示します。
3. [ETL エンティティ] セクションで、[新規] をクリックしてエンティティを作成します。
   エンティティを作成する必要があるユーザーは次のとおりです。

- scim-user：SCIM のフィールド用。
- ユーザー (sys_user) またはグループ (sys_user_group) テーブル：SCIM を使用してデータベーステーブルからマッピングするフィールド用。たとえば、SCIM を使用してユーザーの詳細をカスタマイズするには、sys_user テーブルを使用します。

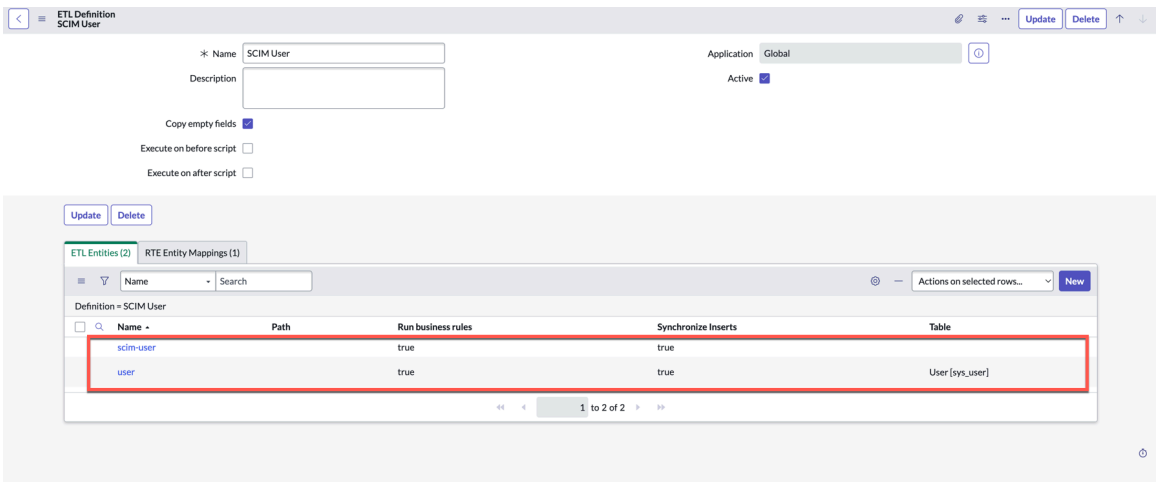
a. フォームのフィールドに入力します。

[ETL エンティティ] フォーム

フィールド	説明
名前	ETL エンティティの名前。
アプリケーション	このレコードのアプリケーションスコープ。
テーブル	ETL エンティティのターゲットテーブル。
定義	選択された ETL エンティティ。
パス	このエンティティの一意のパス。インポートセットテーブルを表すエンティティのパスを指定しないでください。エンティティがコレクションを表す場合、パスはアスタリスク (*) で終わる必要があります。この要件は、中間エントリとターゲットテーブルエンティティに適用されます。
挿入を同期する	レコード挿入を同期することで一意の結合フィールド値のある 1 つのレコードのみを保証するオプション。
ビジネスルールを実行	ビジネスルールを実行するオプション。

b. [送信] をクリックします。

scim-user および user テーブルに対して ETL エンティティが作成されます。この ETL エンティティ内に ETL エンティティフィールドを作成し、RTE エンティティマッピングを作成して両方のエンティティをマッピングする必要があります。



4. エンティティを作成してマッピングします。

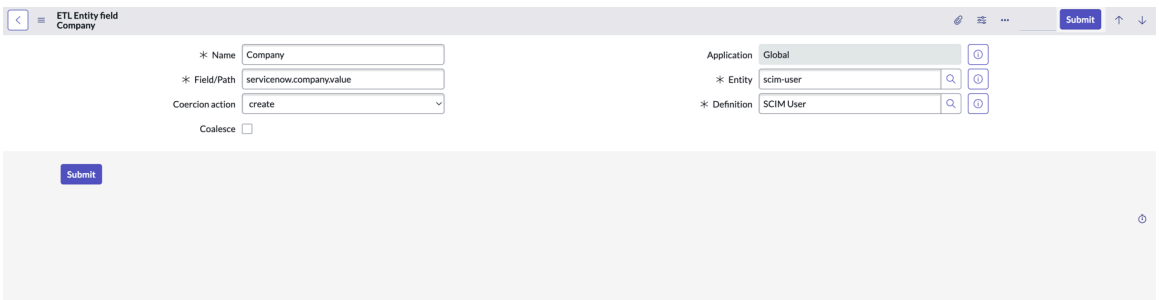
a. 送信されたレコード (scim-user と user) を開きます。

b. [ETL エンティティ] の各フィールドで、[新規] をクリックしてフィールドを追加します。

c. フォームのフィールドに入力します。

[ETL エンティティ] フォーム

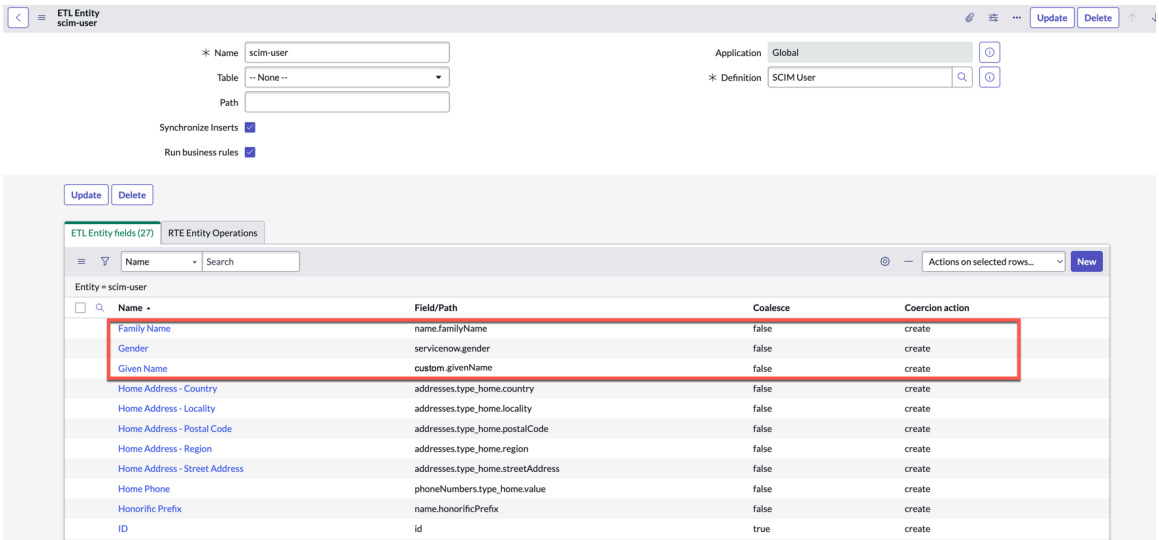
フィールド	説明
名前	[ETL エンティティ] フィールド定義の名前。
アプリケーション	このフィールド定義が属する選択された ETL エンティティ。
フィールド/パス	このフィールドは列またはパスです。 ▪ エンティティがインポートテーブルまたはターゲットテーブルの場合、フィールドは列名です。 ▪ フィールドにネストされた構造がある場合、フィールドはパスです。
エンティティ	この操作が適用されるエンティティ。検索アイコンを使用してエンティティを選択します。
強制アクション	参照または選択肢が見つからない場合にシステムで行われる必要がある操作です。オプションは次のとおりです。 ▪ 作成：新しい参照か選択肢を作成します。現在のレコードに参照または選択肢を割り当てます。 ▪ 却下：レコード全体をデータベースに保存しません。 ▪ 無視：現在の値を空に設定します。
定義	このフィールド定義が属する選択された ETL エンティティ。
結合	既存のレコードをクエリするオプション。



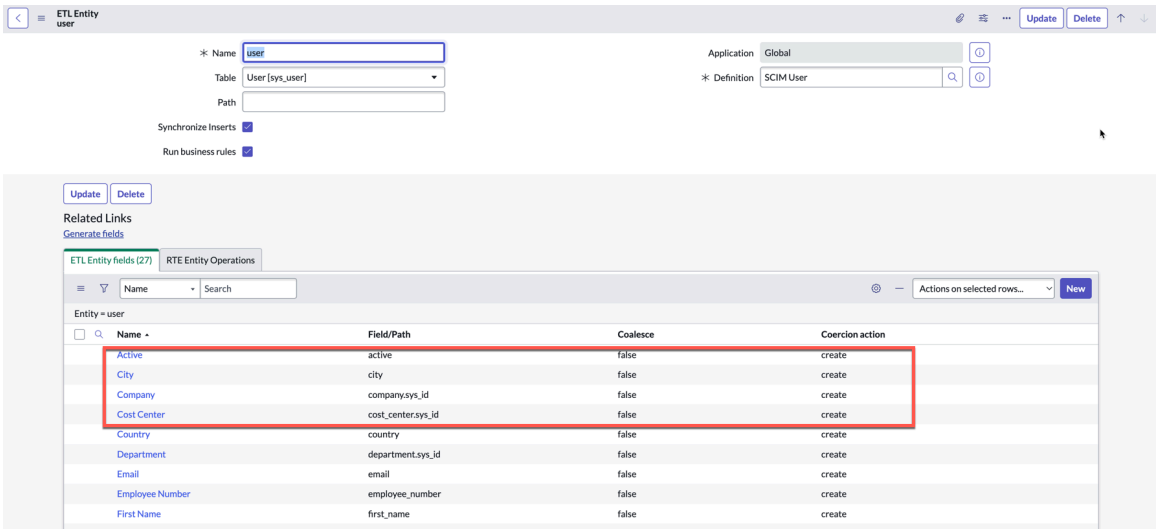
d. [送信] をクリックしてエントリを送信します。

要件に基づいて、複数のエントリを [ETL エンティティ] フィールドとして追加できます。

scim-user の ETL エンティティには、コア拡張 (user)、ServiceNow 拡張、またはカスタム拡張を持つエンティティを含めることができます。



user の ETL エンティティには、データベーステーブルのエンティティを含めることができます。たとえば sys_user テーブルなどです。



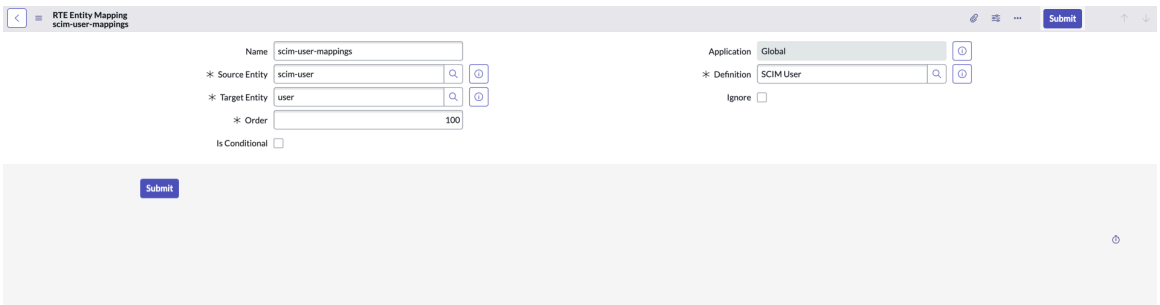
注: 受信する SCIM のフィールドにフィルターを追加するには、アンダースコア (_) を使用します。このアンダースコアは EQ フィルターに変換されます。たとえば、属性 *email.type_work.value* はメールの SCIM フィルター *[type eq "work"].value* を適用します。

scim-user と user の [ETL エンティティ] レコードの両方でフィールドを作成したら、RTE エンティティのマッピングレコードを作成する必要があります。その後、ソースとターゲットの定義を指定して、両方のフィールドをマッピングする必要があります。

5. [RTE エンティティマッピング] セクションで、[新規] をクリックしてエンティティマッピングを作成します。
6. フォームの各フィールドに入力します。

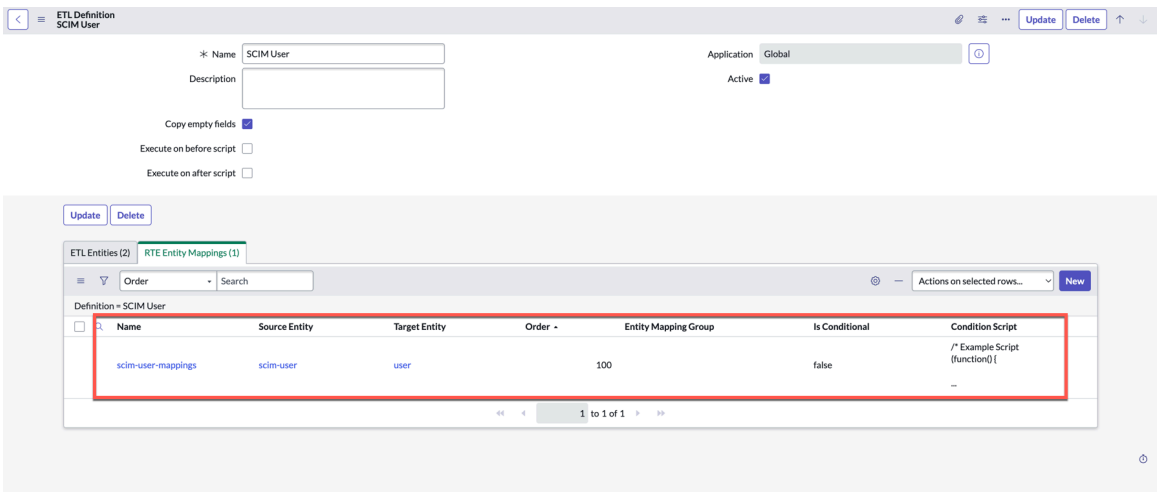
フィールド	説明
名前	マッピングの名前。

フィールド	説明
ソースエンティティ	マッピングのソースエンティティ。
ターゲットエンティティ	マッピングのターゲットエンティティ。
順番	マッピングを処理する順序。
条件つき	マッピングを条件付きとして指定するオプション。
条件スクリプト	マッピングのために満たす必要のある条件を定義するスクリプト。
アプリケーション	このレコードのアプリケーションスコープ
定義	このマッピングが属する選択された ETL エンティティ。
無視	強力なインポートセット変換 (RTE) を使用してデータを統合するときに、この ETL エンティティマッピングを無視するかどうかを指定するオプション。



7. [送信] をクリックします。

   次の例は、[ETL エンティティ] の scim-user と user の両レコードをマッピングするために作成されたレコードを示しています。



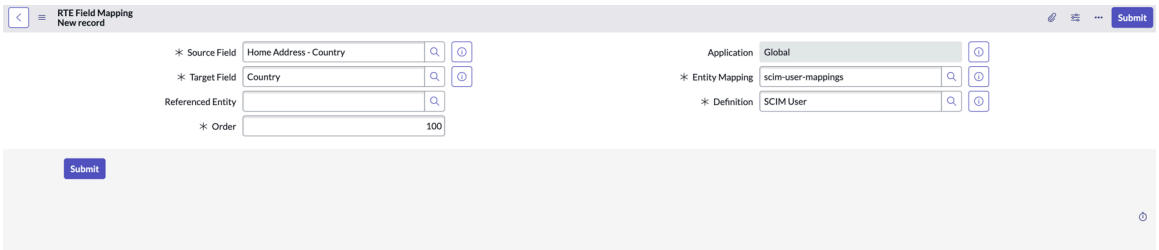
8. 送信済みのレコード (scim-user-mappings) を開き、[ETL エンティティ] の scim-user と user のレコード間のマッピングを作成します。

   a. [RTE フィールドマッピング] セクションで、[新規] をクリックします。

   b. フォームのフィールドに入力します。

ETL エンティティフィールド

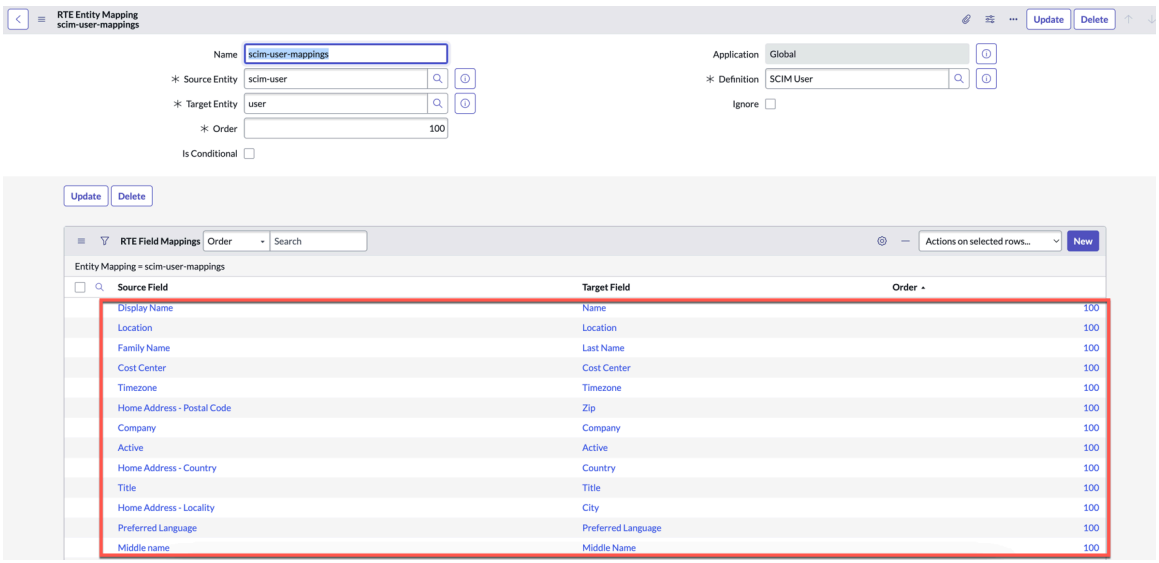
フィールド	説明
ソースフィールド	このレコードのアプリケーションスコープ。
アプリケーション	このフィールド定義が属する選択された ETL エンティティ。
ターゲットフィールド	操作が単一の出力を受け取る場合は、操作の ETL フィールドを出力します。
エンティティマッピング	この操作が適用されるエンティティマッピング。
参照エンティティ	参照されるエンティティとそれが適用される操作。
定義	このフィールド定義が属する選択された ETL エンティティ。
順番	エンティティで実行される操作の順序。



ソースフィールドの [自宅住所 - 国] (scim-user の ETL エンティティ) は、ターゲットフィールドを [国] (user の ETL エンティティ) としてマッピングします。

c. [送信] をクリックしてエントリを送信します。

要件に基づいて、複数のエントリを RTE エンティティマッピングとして追加できます。



ソースフィールドとターゲットフィールドが設定どおりにマッピングされています。SCIM を使用して CRUD 操作を実行すると、カスタマイズした値がそれぞれのテーブルで更新されます。

結果
このように ETL の定義とマッピングを設定することで、ソーステーブルからデータを抽出し、必要に応じてデータを変換して、データをターゲットテーブルにロードすることが可能になります。

関連トピック

抽出変換ロード (ETL) 定義を作成する

マップされていないフィールドの処理

SCIM のカスタマイズでは、マップされていないフィールドをさまざまな方法で処理できます。

SCIM のカスタマイズ中、sys_user および sys_user_group テーブルに含まれていないフィールドは、次の機能を実行することでマッピングできます。

SCIM のカスタマイズ (作成または更新)

SCIM クライアントを作成または更新できます。

- SCIM アドミンは、ETL 定義または RTE にマッピングされていないフィールドの *onBefore* および *onAfter* スクリプトにカスタムスクリプトを追加できます。
- SCIM アドミニストレーターは、*onBefore* および *onAfter* スクリプトにカスタムスクリプトを追加することで、RTE マッピングを上書きできます。
- RTE の *onBefore* または *onAfter* スクリプトでスクリプト可能な API を呼び出し、着信した要求にアクセスして他のテーブルやリスト、マッピングされていない属性の変換を実行できます。
- *sn_auth.SCIM2Util.getScimProviderCustomizationContext()* メソッドを使用して、*scimResource* オブジェクトを含む SCIM 要求コンテキストを提供できます。各操作におけるコンテキスト内の *scimResource* は次のものを表します。
  - **POST**：要求ペイロードで送信される SCIM リソース。
  - **PUT**：要求ペイロードで送信された SCIM リソースで置き換えられる、データベースの現在の SCIM リソース。
  - **PATCH**：パッチ操作を実行した後のデータベースの現在の SCIM リソース。

onAfter スクリプトの例を次に示します。

```
(function onAfter(source, target, importLog) {

    var ctx = sn_auth.SCIM2Util.getScimProviderCustomizationContext();
    gs.info("scim context ee: " + JSON.stringify(ctx.scimResource));

    var roles = ctx.scimResource.roles;
    if(roles) {
        var removingRolesGR = new GlideRecord("sys_user_has_role");
        removingRolesGR.addQuery("user", target.sys_user[0].sys_id);
        removingRolesGR.query();
        removingRolesGR.deleteMultiple();

    for (var i = 0; i < roles.length; i++) {
        var addingRolesGR = new GlideRecord("sys_user_has_role");
        addingRolesGR.setValue("user", target.sys_user[0].sys_id);
        addingRolesGR.setValue("role", roles[i].value);
        addingRolesGR.setValue("state", "active");
        addingRolesGR.insert();
        }
        }
```

```
    var customUserExtn = new
global.SCIMProviderCustomization().getCustomExtensionUrn('User');
    var salary = ctx.scimResource[customUserExtn].salary;
    if (salary) {
        var gr = new GlideRecord("u_user_salary");
        gr.addQuery("user", target.sys_user[0].sys_id);
        gr.query();
        if (gr.next()) {
            gr.setValue("salary", salary);
            gs.info("scim update: " + gr.update());
        } else {
        gr.setValue("salary", salary);
        gr.setValue("user", target.sys_user[0].sys_id);
        gr.insert();
        }
       }

})(source, target, importLog);
```

SCIM の応答をカスタマイズする

GET API 呼び出しの場合、SCIM クライアントへの応答は *SCIMProviderCustomization* スクリプトを拡張することでカスタマイズできます。

スクリプトを拡張する際に、作成者は *customizeUserResponse* および *customizeGroupResponse* メソッドを上書きして、ユーザーリソースとグループリソースの応答を変更できます。

com.snc.integration.scim2.provider.customization.script.id プロパティを使用すると、応答のカスタマイズに使用するスクリプトを SCIM プラグインで使用できるようになります。

ベーススクリプトの拡張例を次に示します。

```
var SCIMCustomizationScript = Class.create();
SCIMCustomizationScript.prototype = Object.extendsObject(SCIMProviderCustomization, {
   initialize: function() {
      SCIMProviderCustomization.prototype.initialize.call(this);
   },
   customizeUserResponse: function(context) {
      try {
         var rolesGR = new GlideRecord("sys_user_has_role");
         rolesGR.addQuery("user", context.scimResource.id);
         rolesGR.query();
         var i = 0;
         context.scimResource.roles = [];
         while (rolesGR.next()) {
            context.scimResource.roles[i] = {
               display: rolesGR.getElement('role.name').getValue(),
               value: rolesGR.getElement('role.sys_id').getValue()
            };
            i++;
         }
         var userGR = new GlideRecord("u_user_salary");
         userGR.addQuery("user", context.scimResource.id);
         userGR.query();
         if (userGR.next()) {
            var salary = userGR.getValue("salary");
```

```
            if (salary) {
                var customExtensionValue =
SCIMProviderCustomization.prototype.getCustomExtensionNodeValue.call(this, "User",
context);
                customExtensionValue.salary = salary;
                SCIMProviderCustomization.prototype.setCustomExtensionNodeValue.call(this,
"User", context, customExtensionValue);
            }
        }
    } catch (ex) {
        gs.error("err: " + ex);
    }
    return context;
  },
  customizeGroupResponse: function(context) {
    return context;
  },
  type: 'SCIMCustomizationScript'
});
```

注:

- *customizeUserResponse* および *customizeGroupResponse* メソッドに含まれるパラメーターは、*scimResource*という 1 つの属性を持つコンテキストオブジェクトです。この属性には、ユーザーまたはグループのリソースオブジェクトの属性がすべて含まれています。
- カスタマイズされたスクリプトインクルードは、アドミンのみが作成および表示できます。
- ユーザーリソースやグループリソースが変更された場合は、コンテキストを元に戻す必要があります。
- リソースオブジェクトに属性の変更がない場合は、*com.snc.integration.scim2.provider.customization.script.id* を空に設定するか、 null として返します。
- *onAfter* スクリプトを使用して特定の属性を保持する場合は、カスタマイズされたスクリプト内の *scimResource* オブジェクトにデータベース値を設定する必要があります。このアクションが必要なのは、システムが次の操作を実行できるようにするためです。
  - PUT と PATCH の操作中に *onAfter* スクリプトの正しい *scimResource* オブジェクトを取得する。
  - *onAfter* スクリプトによって保持された属性をクライアントへの応答に含める。

ヘルパー関数

SCIM のカスタマイズに使用するヘルパー関数の一部を次に示します。これらの関数で、さまざまなタイプの情報をフェッチしたり設定したりすることが可能です。

ヘルパー関数

関数	目的
SCIMProviderCustomization.prototype.getCustomExtensionNodeValue.call(this, "User");	カスタム拡張スキーマの値をフェッチする。
SCIMProviderCustomization.prototype.getServiceNowExtensionNodeValue.call(this, "User");	ServiceNow 拡張スキーマの値をフェッチする。

ヘルパー関数 (続く)

関数	目的
`SCIMProviderCustomization.prototype.getCustomExtensionUrn.call(this, "User", context);`	応答からカスタム拡張 URN を取得する。
`SCIMProviderCustomization.prototype.getCustomExtensionNodeValue.call(this, "User", context);`	ServiceNow テーブルの応答からカスタム拡張ノード値を取得する。
`SCIMProviderCustomization.prototype.setCustomExtensionNodeValue.call(this, "User", context, customExtensionValue);`	応答内のカスタム拡張ノード値を設定する。

ヘルパー関数の使用例を次に示します。

```
var customExtensionUrn =
SCIMProviderCustomization.prototype.getCustomExtensionUrn.call(this, "User");
var customExtensionValue =
SCIMProviderCustomization.prototype.getCustomExtensionNodeValue.call(this, "User",
 context);
customExtensionValue.age = "18";
SCIMProviderCustomization.prototype.setCustomExtensionNodeValue call(this, "User", context,
 customExtensionValue);
```

注: RTE は、sys_user および sys_user_group テーブル以外のテーブルのデータ設定をサポートしています。

ソース定義の作成

ソース定義を作成して、リソースのプロビジョニング元となる ID ソースに関する情報をキャプチャします。

始める前に
必要なロール：scim_admin

警告: このロールは慎重に付与してください。scim_admin ロールはユーザーに admin ロールを付与することと同じで、scmin_admin は個人識別可能情報 (PII) を追加または更新することができます。

このタスクについて
ソース定義を使用すると、プロビジョニング中の認証に使用する OAuth エンティティに、プロビジョニング ID ソースをマッピングできます。

ソース定義が作成されると、その ID ソースからプロビジョニングされるすべてのリソースが、対応するソース定義 ID にマッピングされます。

ソース定義では、次のような動作によって、必要なソース情報がキャプチャされます。

- リソースのプロビジョニング元の SCIM クライアントを識別する。
- 外部 ID から提供された重複情報を解決する。
  - 複数の ID ソースがプロビジョニングリソースである場合、外部 ID は ID ソースに対してのみ一意であるため、複数のリソースが同じ外部 ID 値を持つ可能性があります。
  - 外部 ID の SCIM フィルターで複数のリソースが返された場合は、要求元の ID ソースのソース定義に基づいてリソースを解決できます。

注: ソース定義は、OAuth 認証方法を使用する ID ソースに対してのみ作成できます。

自動翻訳

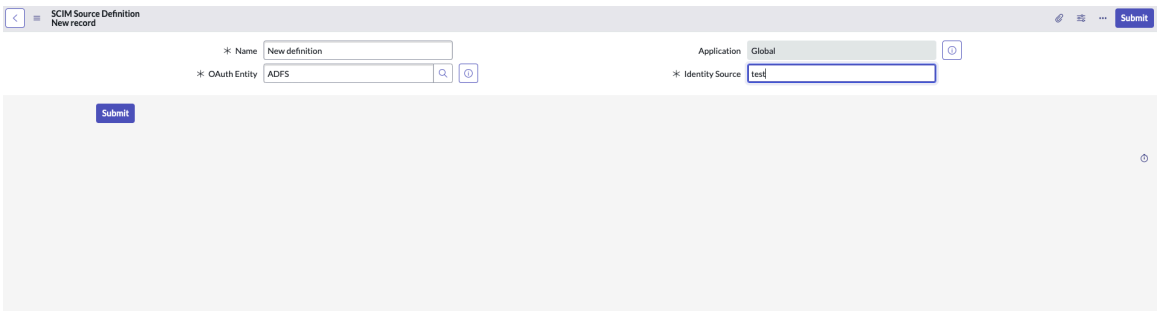
手順

1. 移動先 すべて > **SCIM** > ソース定義.
2. [SCIM ソース定義] ページで、[新規] をクリックします。
3. フォームのフィールドに入力します。

SCIM ソース定義

フィールド	説明
名前	ソース定義の名前。
アプリケーション	このレコードのアプリケーションスコープ。
OAuth エンティティ	統合ユーザーの OAuth エンティティ。このエンティティは、ID ソースプロバイダーがユーザーをプロビジョニングするために使用されます。
ID ソース	ID ソースプロバイダーの名前。例：Azure AD、Okta など

注: *com.snc.integration.scim2.resolve.externalid.conflict* プロパティを有効にすると、要求元の ID ソースによって作成された SCIM リソースのみが返されます。デフォルトでは、外部 ID フィルターを持つすべての一致するリソースが返されます。



4. [送信] をクリックします。

結果
SCIM ソース定義が作成されます。SCIM ETL 定義を使用して、sys_user および sys_user_group テーブルの拡張スキーマに基づいてリソースをマッピングします。詳細については、「SCIM ETL 定義の作成」を参照してください。

SCIM クライアント

SCIM クライアントは、外部システム上の SCIM エンドポイントで公開された CRUD 操作を通じて、ID リソースのプロビジョニングと更新を容易にするものです。

探索

SCIM クライアントについて学習します。

アクティブ化

SCIM クライアントをアクティブ化します。

SCIM クライアントプロパティ

SCIM をカスタマイズする方法の詳細を取得します。

トラブルシューティング

SCIM のソース定義について詳細を確認します。

自動翻訳

SCIM クライアントの概要

SCIM クライアントは、外部システム上の SCIM エンドポイントで公開された CRUD 操作を通じて、ID リソースのプロビジョニングと更新を容易にするものです。

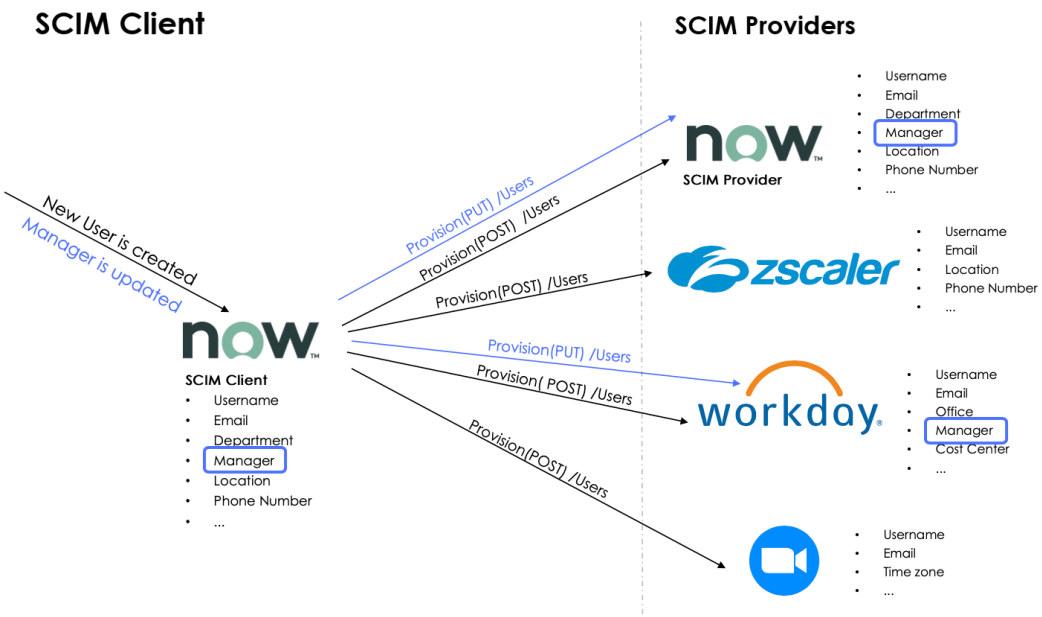
SCIM クライアントは、SCIM に準拠する REST 要求をサポートするシステムで ID リソースを作成、更新、および削除する際に使用します。このクライアントを使用して、ID ライフサイクルの管理や、ServiceNow インスタンス間または ServiceNow と他の SCIM プロバイダー間での ID 属性の同期を行います。☒

API はクライアントによって公開されるため、単一または複数の SCIM プロバイダーのリソースを作成、更新、または削除するプロセスを自動化できます。たとえば、開発者が組織に参加した場合には、Git や職場などへのアクセス権を付与する必要があります。

SCIM クライアントでは、次のアクションを実行できます。

- ユーザーまたはグループのメンバーシップの ID とアクセスをプロビジョニングする。
- ID および関連リソースを SCIM 準拠のシステムと同期する。
- 任意の SCIM プロバイダーを ServiceNow® に統合する。
- ID とアクセスをプロビジョニング解除する。

SCIM クライアントは、指定されたジョブを実行するワークフローや自動化を構築する際に統合開発者が使用できるスクリプト可能な API を提供します。スクリプト可能 API の詳細については、「SCIM2Client API」を参照してください。



SCIM クライアントの設定

SCIM クライアントを設定するには、次のタスクを実行します。

- 特定の SCIM プロバイダーのすべての発信コールに対して REST メッセージを作成します。詳細については、「REST メッセージの作成」を参照してください。
- SCIM プロバイダーを作成して、その SCIM プロバイダーから REST メッセージを使用してリソースタイプとスキーマ情報をフェッチします。SCIM プロバイダー内のリソースを更新するには、HTTP メソッド (PUT または PATCH) の設定を有効にします。詳細については、「SCIM プロバイダーの作成」を参照してください。

- 特定のリソースタイプと SCIM プロバイダーの ServiceNow 属性に対する SCIM 属性のマッピングを作成します。 詳細については、「SCIM プロバイダーリソースマッピングの作成」を参照してください。
- データベーステーブルとフィールド名で SCIM フィールドのマッピングを実行します。デフォルト値を渡すか、値をフェッチするスクリプトを記述します。詳細については、「SCIM 属性マッピングの作成」を参照してください。

SCIM クライアントプラグインのアクティブ化

SCIM クライアントをアクティブ化するには、SCIM v2 - ServiceNow Cross-domain Identity Management Client (com.snc.integration.scim2.client) プラグインをインストールします。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > システムアプリケーション > 利用可能なすべてのアプリケーション > すべて.
2. フィルター基準と検索バーを使用して、SCIM v2 - ServiceNow Cross-domain Identity Management Client (com.snc.integration.scim2.client) プラグインを検索します。

   名前または ID でプラグインを検索できます。プラグインが見つからない場合は、ServiceNow 担当者から要求する必要があります。
3. [インストール] を選択して、インストールプロセスを開始します。

   注: ドメインセパレーションと代理アドミンがインスタンスで有効になっている場合、管理ユーザーはグローバルドメインに含まれている必要があります。それ以外の場合、次のエラーが表示されます： 「別の操作が実行されているため、アプリケーションのインストールは利用できません： <プラグイン名> のプラグインの有効化 (Application installation is unavailable because another operation is running: Plugin Activation for <plugin name>)」

   インストールが完了するとメッセージが表示されます。プラグインとともにインストールされるコンポーネントの詳細については、「 アプリケーションとともにインストールされているコンポーネントの検索 」を参照してください。

SCIM クライアントのプロパティ、テーブル、スクリプト可能な API およびログ

SCIM v2 - ServiceNow Cross-domain Identity Management Client (com.snc.integration.scim2.client) プラグインには、次のシステムプロパティ、テーブル、スクリプト可能な API、およびログが含まれています。

プロパティ

SCIM クライアントでは、次のシステムプロパティが追加されます。

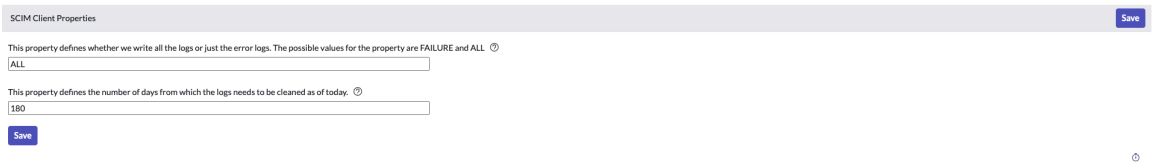
プロパティ

名前	説明
com.snc.integration.scim2.client.log.request.status	このプロパティは、すべてのログレコードを書き込むか、エラーログレコードのみを書き込むかを決定します。設定可能な値は **FAILURE** または **ALL** です。 デフォルト値：**ALL**

プロパティ (続く)

名前	説明
com.snc.integration.scim2.client.log.cleanup.duration	ログのパージ間隔、ログをクリアするまでの日数を決定します。 デフォルト値：**180**

プロパティを設定するには、次に移動します: すべて > **SCIM** > **SCIM** クライアントプロパティ.



テーブル

SCIM クライアントでは、次のテーブルが追加されます。

テーブル

名前	説明
SCIM プロバイダー (sys_scim_provider)	名前や REST メッセージリソース定義など、各 SCIM プロバイダーのデータを格納します。
SCIM プロバイダーリソースマッピング (sys_scim_provider_resource_mapping)	各プロバイダーとリソース名のプライマリテーブル情報を格納します。
SCIM 属性マッピング (sys_scim_attribute_mapping)	テーブルフィールドやスクリプトなど、各 SCIM 属性値の取得元となるソースの詳細を格納します。
SCIM クライアントログ (sys_scim_client_logs)	SCIM プロバイダーに対してトリガーされた各コールのステータスを格納します。

スクリプト可能 **API**

SCIM2Client API は、クロスドメイン ID 管理システム (SCIM) (サーバーロール) を呼び出して、サービスプロバイダー (SP) のデータを作成、更新、または削除します。SCIM クライアントのスクリプト可能 API は、システムのコンテキストで実行されているスクリプト内で使用するか、システムアドミンユーザーが使用する必要があります。

たとえば、システムユーザーとして Integration Hub ワークフローを実行しているときや、スケジュール済みジョブを実行しているときなどに、スクリプトを使用できます。

スクリプト可能 API を使用するユースケースをいくつか次に示します。

- アドミンが、バックグラウンドスクリプト、ビジネスルール、スクリプトインクルード呼び出し、ワークフローなどから ID 情報をプロビジョニングする。
- アドミンが、ID プロビジョニングのスケジュール済みジョブまたはオンデマンドジョブを実行する。
- プロビジョニングスクリプト可能 API の呼び出しを使用して、 スクリプトステップでワークフローやサブワークフローを実行する。

- プロビジョニングスクリプトをビジネスルールやスクリプトインクルードに直接追加する。スクリプトはアドミン以外のユーザーがトリガーしてもかまわない。このユースケースは、次のような状況で有効に働きます。
  - ユーザーがトークンにアクセスできる。すなわち、REST テンプレートからトークンを生成するロールを持っている。
  - ユーザーが、マッピングされたテーブルから SCIM 属性値を取得するためのアクセス権を持っている。

スクリプト可能 API の詳細については、「SCIM2Client API」を参照してください。

SCIM クライアントログ

SCIM クライアントログには、SCIM API に関するプロビジョニングステータスが表示されます。プロビジョニングステータスを表示するには、次に移動します: すべて > **SCIM** > **SCIM** クライアントログ.

REST メッセージの作成

特定の SCIM プロバイダーのすべての発信コールに対する REST メッセージを設定します。

始める前に
必要なロール：admin

手順

1. 移動先 すべて > システム **Web** サービス > アウトバウンド > **REST** メッセージ.
2. [**New**] をクリックします。
3. フォームのフィールドに入力します。

SCIM プロバイダー

フィールド	説明
名前	このメッセージのわかりやすい名前。
エンドポイント	SCIM プロバイダーのベース URL。 例：https://example.service-now.com/api/now/scim
認証タイプ	外部の SCIM プロバイダーへの接続に使用される認証のタイプ。詳細については、「送信 REST の認証」を参照してください。
HTTP ヘッダー	外部の SCIM プロバイダーから期待されるコンテンツタイプ。 たとえば、ヘッダー名は Content-type: application/scim+json という API 要求の本文のコンテンツタイプです。

4. [HTTP メソッド] 関連リストで、**[New]** をクリックします。
5. 次の HTTP メソッドを設定します。

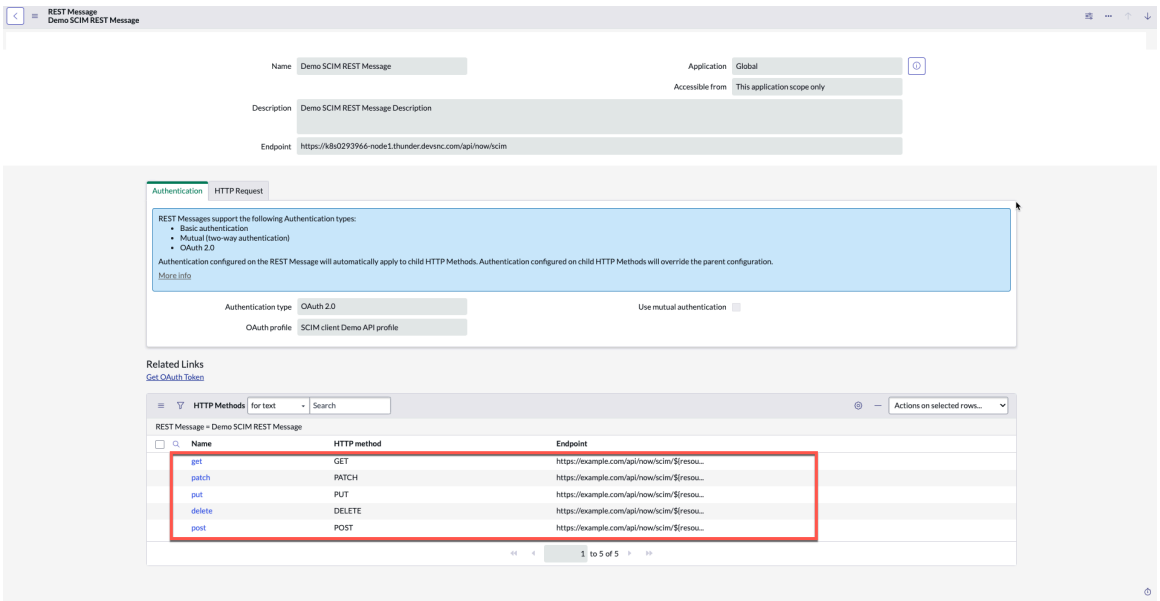
注: URL には、置き換える必要のある変数があります。

HTTP メソッド

メソッド	サンプル **URL**
GET	https://example.com/api/now/scim/${resourceName}
PATCH	https://example.com/api/now/scim/${resourceName}/${resourceId}
PUT	https://example.com/api/now/scim/${resourceName}/${resourceId}
DELETE	https://example.com/api/now/scim/${resourceName}/${resourceId}
POST	https://example.com/api/now/scim/${resourceName}/${resourceId}

注:

- SCIM クライアントの操作に使用するすべての HTTP メソッドを作成する必要があります。
- サンプル REST メッセージはベースシステムから入手できます。



6. [送信] をクリックします。

結果
REST メッセージレコードが作成されます。

次のタスク
REST メッセージを使用して SCIM プロバイダーを作成します。詳細については、「SCIM プロバイダーの作成」を参照してください。

REST メッセージの作成方法の詳細については、「REST メッセージの作成」を参照してください。

SCIM プロバイダーの作成

SCIM プロバイダーを作成して、その SCIM プロバイダーから REST メッセージを使用してリソースタイプとスキーマ情報をフェッチします。SCIM プロバイダー内のリソースを更新するには、HTTP メソッド (PUT または PATCH) の設定を有効にします。

始める前に

注: サンプルの SCIM プロバイダーはベースシステムに含まれています。これを使用して自社の要件に基づいて設定しても、新しいレコードを作成してもかまいません。

必要なロール：admin

手順

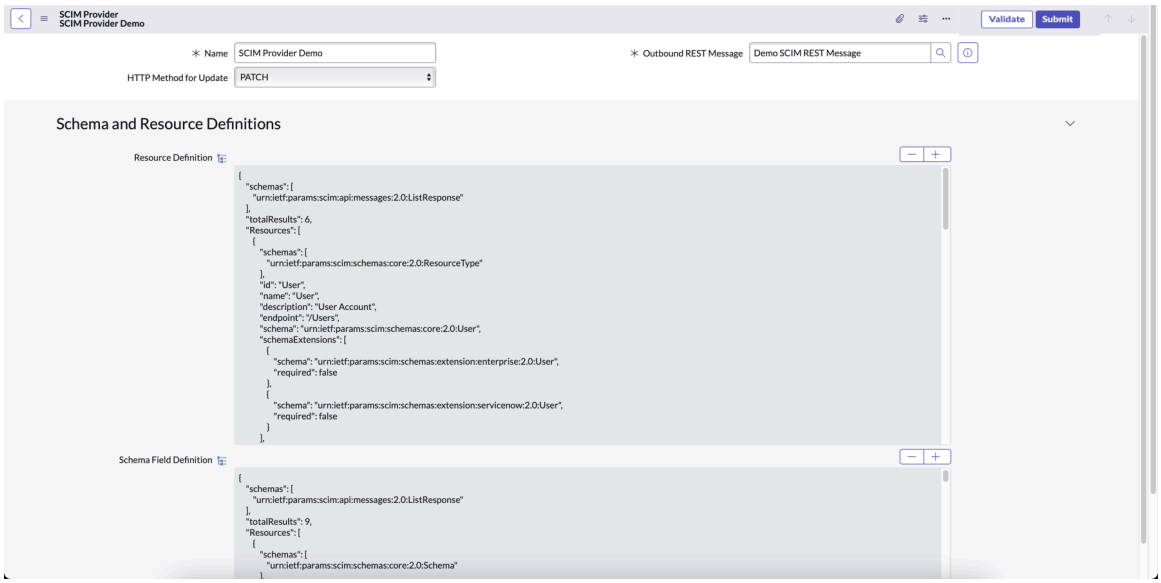
1. 移動先 すべて > **SCIM** クライアント > **SCIM** プロバイダー.
2. [SCIM プロバイダー] ページで、[新規] をクリックします。
3. フォームのフィールドに入力します。

[SCIM プロバイダー] フォーム

フィールド	説明
名前	SCIM プロバイダーの名前
送信 REST メッセージ	SCIM プロバイダーの API を呼び出すために使用するメッセージ。詳細については、「REST メッセージの作成」を参照してください。
更新のための HTTP メソッド	リソースマッピングの更新に使用する HTTPS メソッドのタイプ。PATCH または PUT メソッドは、既存のリソースのプロビジョニング中にクライアントが ID リソースを更新するときに使用可能です。

注:

- [リソース定義] および [スキーマフィールド定義] フィールドは、SCIM プロバイダーの */ResourceTypes* および */Schemas* 公開エンドポイントから取得されます。これらのフィールドは、REST メッセージの選択後に自動入力されます。
- SCIM プロバイダーで REST メッセージ、スキーマ、またはリソースタイプが更新された場合は、 [リフレッシュ] をクリックしてから [更新] をクリックし、[リソース定義] フィールドと [スキーマフィールド定義] フィールドを更新します。



4. [送信] をクリックします。

結果
SCIM プロバイダーの詳細が正常に作成されます。SCIM プロバイダーリソースマッピングを使用して、SCIM プロバイダーの詳細をユーザーやグループなどのリソースにマッピングします。詳細については、「SCIM プロバイダーリソースマッピングの作成」を参照してください。

SCIM プロバイダーリソースマッピングの作成

特定のリソースタイプと SCIM プロバイダーの ServiceNow 属性に対する SCIM 属性のマッピングを定義します。

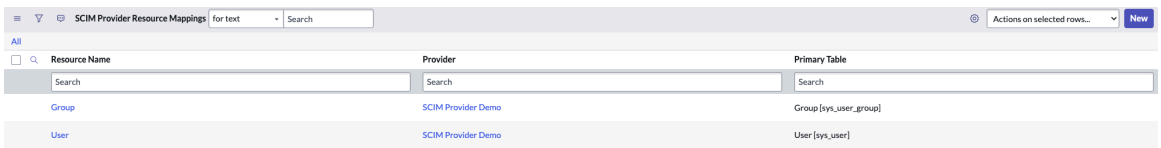
始める前に
必要なロール：admin

手順

1. 移動先 すべて > **SCIM** クライアント > **SCIM** プロバイダーリソースマッピング.

SCIM プロバイダーリソースマッピングには、デフォルトでユーザーとグループのマッピングが入っています。

> **注:** このユーザーやグループのマッピングの中に、参考として使用できるサンプルマッピングが入っています。ユーザーリソースやグループリソースに基づいてマッピングを作成することもできます。

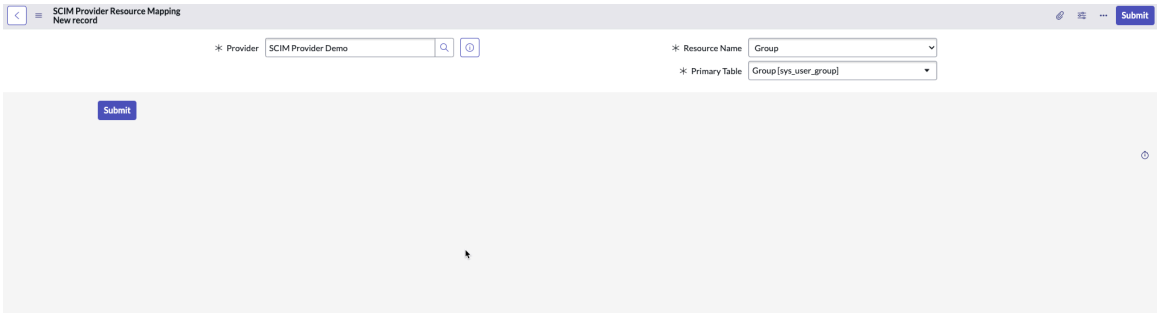


2. [新規] をクリックして、リソースマッピングを作成します。

3. フォームのフィールドに入力します。

リソースマッピング

フィールド	説明
プロバイダー	SCIM プロバイダーの名前。SCIM プロバイダーの作成時に設定されたプロバイダー名を参照します。
リソース名	マッピングを定義する必要があるリソース。
プライマリテーブル	マッピングされるリソースの sys_id を含むテーブル。



4. [送信] をクリックします。

結果
レコードが作成され、[SCIM プロバイダーリソースマッピング] ページに表示されます。SCIM 属性マッピングを使用して、スキーマから属性をさらにマッピングします。詳細については、「SCIM 属性マッピングの作成」を参照してください。

SCIM 属性マッピングの作成

SCIM 属性マッピングを作成し、それを ServiceNow テーブルのフィールドに対するリソースの単一のソースとして使用します。

始める前に
必要なロール：admin

このタスクについて
属性マッピングのタイプとその説明は次のとおりです。

属性マッピングのタイプ

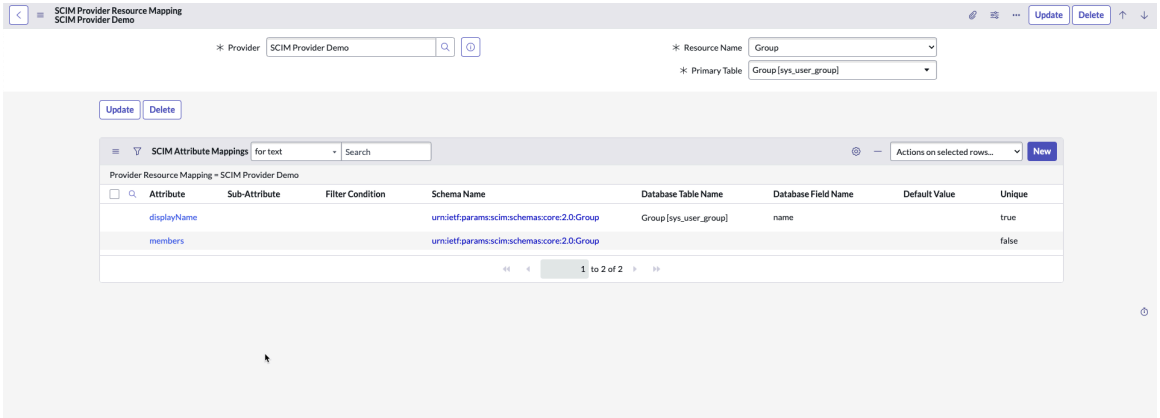
属性マッピングのタイプ	説明
直接	SCIM 属性は、[データベーステーブル名] フィールドと [データベースフィールド名] フィールドを使用して入力されます。
定数	SCIM 属性は、指定されたデフォルト値を活用して入力されます。
スクリプト/カスタム	SCIM 属性は、スクリプトの戻り値を活用して入力されます。この属性では、[スクリプトを実行] オプションを有効にする必要があります。

注:

- パスワード属性はサポートされていません。
- ユーザーリソースおよびグループリソースのサンプルの属性マッピングは、ベースシステムに入っています。これを使用して自社の要件に基づいてマッピングを設定しても、新しいレコードを作成してもかまいません。

手順

1. 移動先 すべて > **SCIM** クライアント > **SCIM** プロバイダーリソースマッピング.
2. 設定用に作成された SCIM リソースマッピングを選択しました。
3. [SCIM 属性マッピング] 関連リストで [新規] をクリックします。



4. フォームのフィールドに入力します。

[SCIM 属性マッピング] フォーム

フィールド	説明
プロバイダーリソースマッピング	プロバイダーとリソースの組み合わせに対する属性マッピング。 このフィールドが自動的に入力されない場合は、検索オプションを使用します。プロバイダーリソースマッピングのレコードを選択します。
スキーマ名	マッピングを定義する必要のある SCIM 属性のスキーマ名。 `urn:ietf:params:scim:schemas:core:2.0:User` など。
属性	マッピングを定義する必要のある属性。`userName` など。
サブ属性	マッピングを定義する必要のあるサブ属性 (あれば)。

自動翻訳

フィールド	説明
	たとえば、*name.familyName* のような複合的なタイプ属性がある場合、属性は **name** で、サブ属性は **familyName** となります。
フィールドタイプ	SCIM 属性のデータタイプ。このフィールドは、SCIM プロバイダーによって定義されたスキーマから自動的に入力されます。 **boolean** など。
複数値	属性に割り当てられる複数の値。1 つの属性に、仕事用メール、自宅用メール、その他のメールなど複数の値を設定できます。 *emails* など。 このフィールドは、SCIM プロバイダーによって定義されたスキーマを使用して入力されます。
フィルター条件	複数の値を持つ属性には、[フィルター条件]を使用して指定できる追加情報を入力することができます。フィルター条件の選択肢は、SCIM プロバイダーによって定義されたスキーマを使用して入力されます。 たとえば、*phoneNumbers* 属性には、職場、モバイル、自宅といった複数のタイプがあります。
一意	SCIM クライアントと SCIM プロバイダーのシステム間でリソースを一意に識別するオプション。複数値の属性を一意としてマークすることはできません。 たとえば、ユーザーリソースでは、*username* 属性を一意とすることができます。
データベーステーブル名	このフィールドを使用して、属性テーブル名を ServiceNow テーブル名にマッピングします。直接の属性マッピングを選択した場合は、このフィールドを定義する必要があります。 たとえば、*username* SCIM 属性は [データベーステーブル名] フィールドの [ユーザー] フィールドにマッピングできます。
データベースフィールド名	[データベースフィールド名] フィールドでは、SCIM 属性を、その SCIM 属性でマッピングされる ServiceNow フィールド名にマッピングします。直接の属性マッピングを選択した

自動翻訳

フィールド	説明
	場合は、このフィールドを定義する必要があります。 たとえば、*username* SCIM 属性は [データベースフィールド名] フィールドの [ユーザー **ID]** フィールドにマッピングできます。
デフォルト値	SCIM プロバイダーに渡されるデフォルト値 の詳細。 フィールドの直接の属性マッピングが null を返す場合、またはデフォルトを使用してハードコードされた値を返せる場合に使用できます。 ハードコードされた値の場合は、データベーステーブル名とフィールド名を [なし] にする必要があります。 たとえば、仕事用メールのプライマリサブ属性の値は **true**としてハードコードできます。
スクリプトを実行	スクリプトを介して属性の値をフェッチするオプション。 このオプションは、フィルター条件を含まない複数値の属性に必要です。複合タイプの属性の場合、スクリプトは属性またはサブ属性レベルで値を指定できます。 たとえば、グループリソースの [メンバー] 属性にはフィルター条件がありません。したがって、スクリプトオプションは [メンバー] 属性の親属性レベルで定義する必要があります。
スクリプト	属性値をフェッチするために使用するスクリプト。 スクリプトの戻り値のタイプは、文字列、または文字列として変換された JSON である必要があります。 スクリプトの出力は、その属性に対してプロバイダーが想定する適切な形式である必要があります。

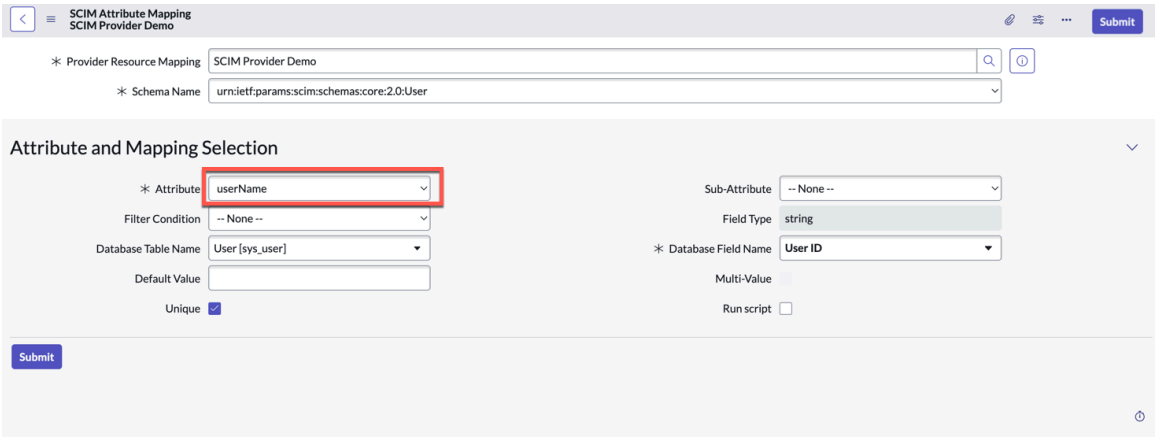
5. [送信] をクリックします。

属性マッピングの参照

属性マッピングを使用すると、ServiceNow のテーブルフィールドに対するリソースの単一のソースとして属性を使用できます。

属性

マッピングを定義する必要のある属性。*userName* など。

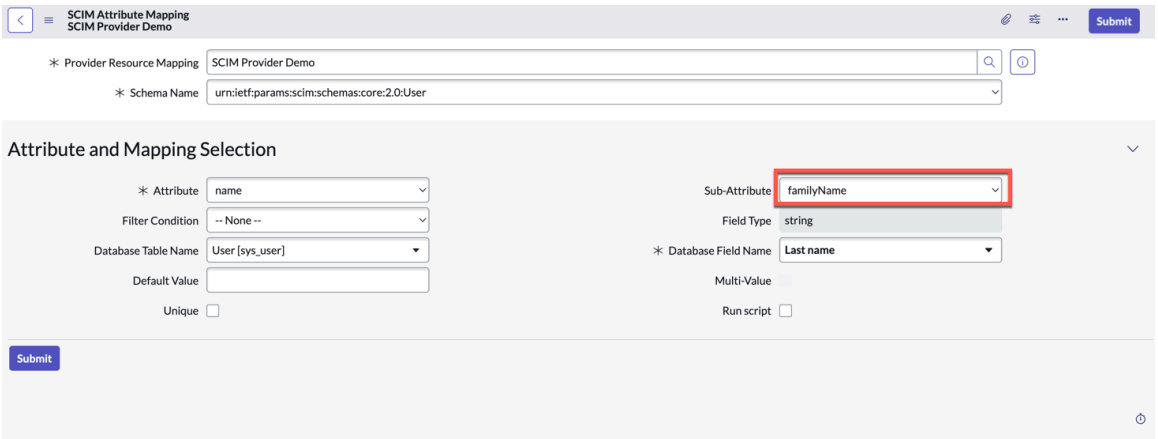


サブ属性

マッピングを定義する必要のあるサブ属性を選択します (ある場合)。

たとえば、*name.familyName* のような複合的なタイプ属性がある場合、属性は *name* で、サブ属性は *familyName* となります。

ユーザー名のような単純な属性の場合、[サブ属性] の値は [なし] になります。

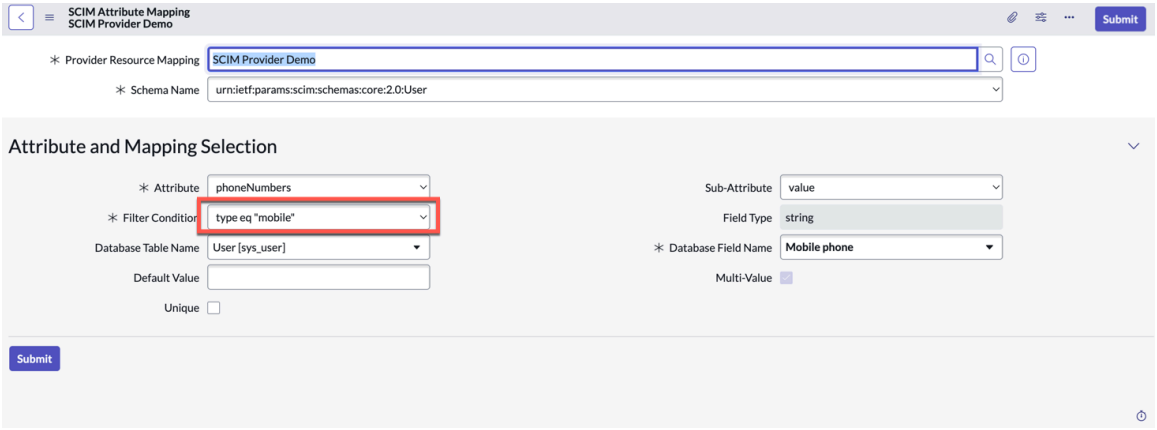


フィルター条件

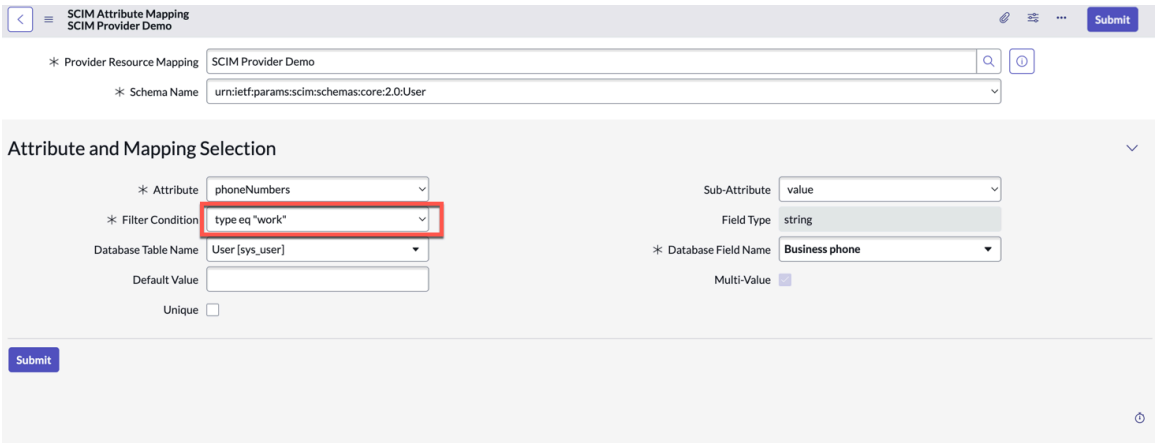
複数値の属性には、フィルター条件を使用して指定可能な追加情報を設定できます。フィルター条件の選択肢は、SCIM プロバイダーによって定義されたスキーマを使用して入力されます。

たとえば、*phoneNumbers* 属性には、職場、モバイル、自宅といった複数のタイプがあります。

一連の可能な値からフィルター条件を指定できます。たとえば、*phoneNumber* 属性にはフィルター条件を **type eq "mobile"** として設定できます。



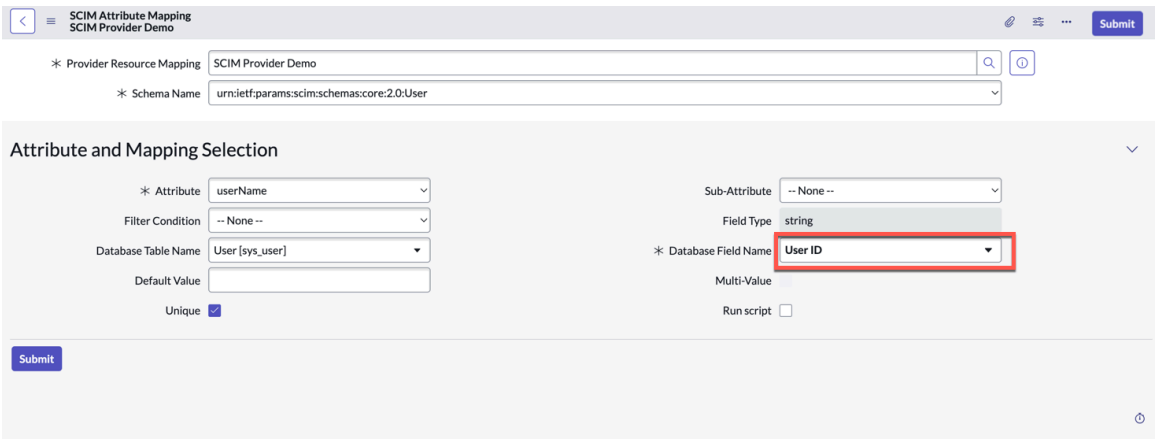
phoneNumber 属性には **type eq "work"** のフィルター条件を設定することもできます。



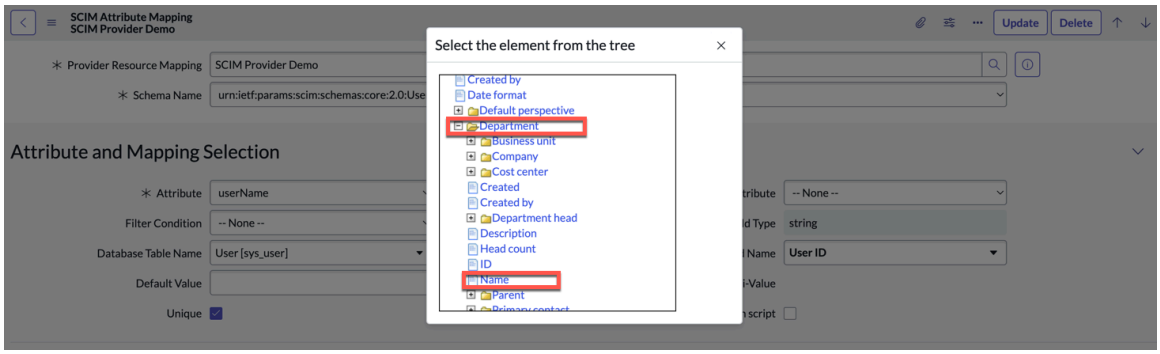
データベースフィールド名

直接の属性マッピングのオプションを選択した場合は、この属性を定義する必要があります。[データベースフィールド名] フィールドには SCIM 属性がマッピングされた ServiceNow フィールド名が表示されます。

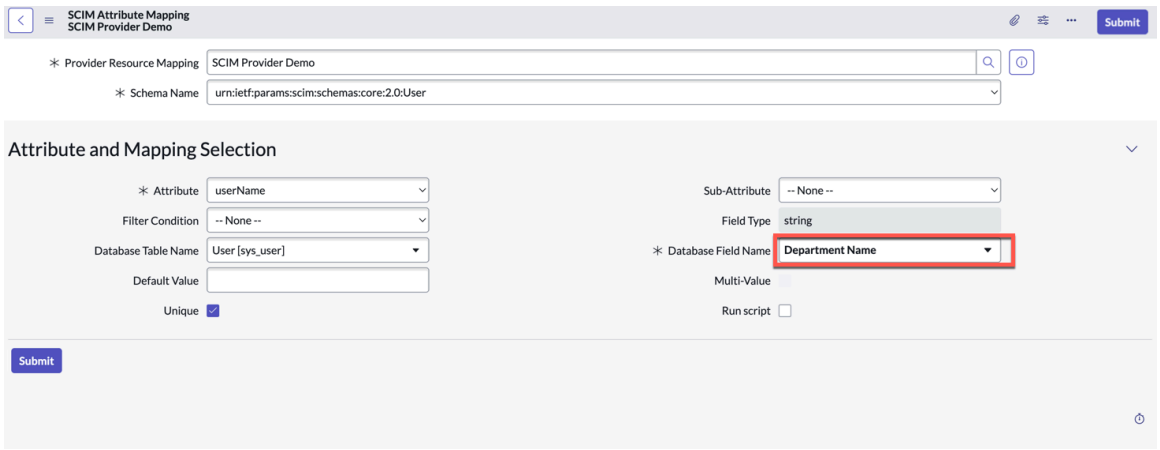
たとえば、*username*という SCIM 属性は、ユーザーには [データベーステーブル名] フィールドとして、ユーザー ID フィールドには [データベースフィールド名] フィールドとしてマッピングできます。



[データベースフィールド名]を使用してドット連結することもできます。たとえば、部門という SCIM 属性は [部門名] フィールドにマッピングできます。



ここでは、[データベーステーブル] は [ユーザー] で、[データベースフィールド名] は [部門名] です。

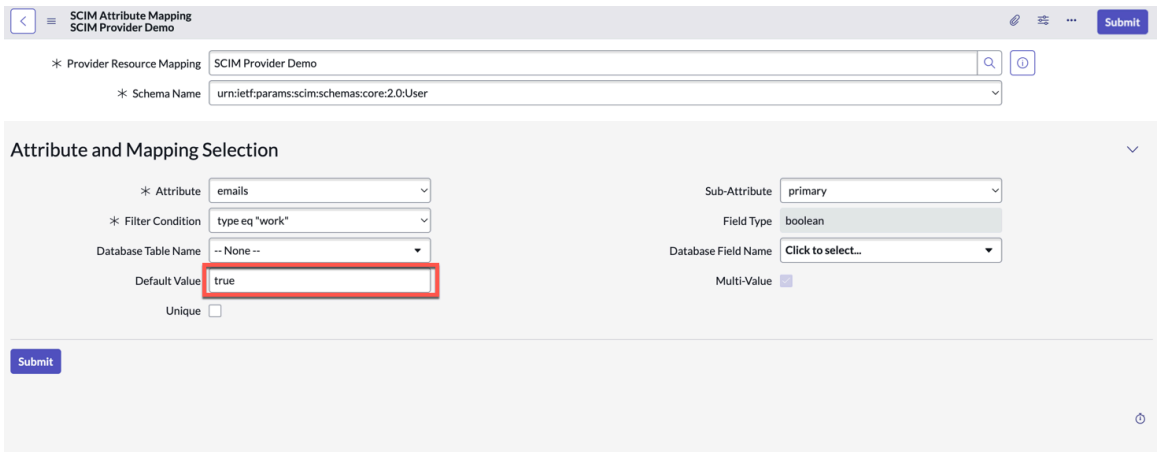


デフォルト値

そのフィールドの直接の属性マッピングが null を返すと、デフォルト値が SCIM プロバイダーに渡されます。そのデフォルト値を使用してハードコードされた値を返すこともできます。

ハードコードされた値の場合は、データベーステーブル名とフィールド名を [なし] にする必要があります。

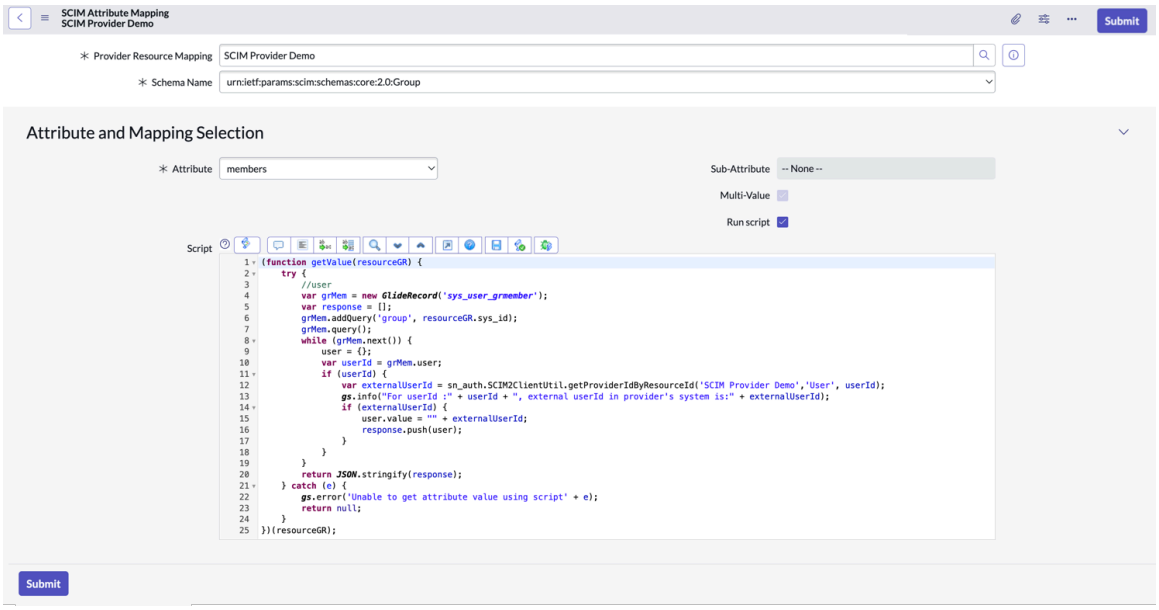
たとえば、仕事用メールのプライマリサブ属性の値は **true**としてハードコードできます。



スクリプト

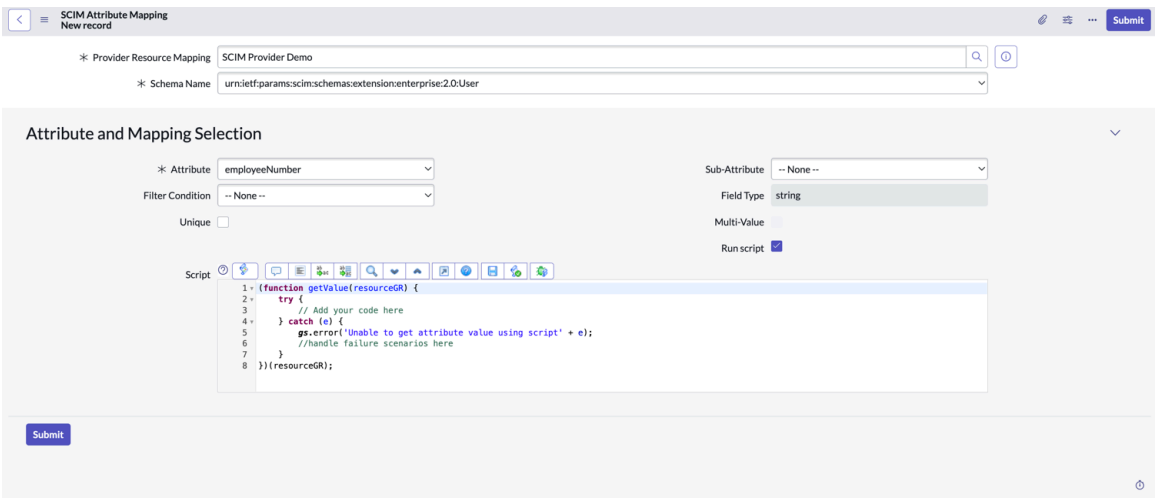
属性値をフェッチするためにはスクリプトを使用します。スクリプトの戻り値のタイプは常に、文字列、または文字列として変換された JSON である必要があります。スクリプトの出力は、その属性に対してプロバイダーが想定する適切な形式である必要があります。

複数値の属性のサンプルスクリプトは次のとおりです。



スクリプトの出力には、文字列化された JSON アレイが必要です。

単純な値の属性のサンプルスクリプトは次のとおりです。



スクリプトの出力は文字列である必要があります。

SCIM クライアントのトラブルシューティング

トラブルシューティングのアクションは、SCIM クライアントの設定や実行時の一般的な問題解決に役立ちます。

トラブルシューティング

問題	アクション
応答に含まれるメッセージは次のとおりです。 “message":"Unable to access the table core_company with id: 0c441abbc6112275000025157c651c89, Please cross check the Access control rules"	このメッセージは、API がユーザーコンテキストで呼び出され、ユーザーがテーブルにアクセスできない場合に表示されます。

トラブルシューティング (続く)

問題	アクション
	スクリプト可能 API がシステムコンテキストで呼び出されていることを確認する必要があります。
応答に含まれるメッセージは次のとおりです。 `{"message":"User Not Authenticated","detail":" Required to provide Auth information”}`	• 対応する REST メッセージによってトークンが生成され、トークンが有効であることを確認してください。 • スクリプト可能 API がシステムコンテキストで呼び出されていることを確認してください。
応答に含まれるメッセージは次のとおりです。 `Script execution failed, the reason is: Cannot cast java.lang.Integer to java.lang.String`	SCIM 属性マッピングで、この問題のあるスクリプトからフェッチするようにフィールドが定義されている場合は、戻り値のタイプが常に文字列であることを確認します。
応答に含まれるメッセージは次のとおりです。 `"status": "400", "scimType": "invalidValue", "detail": "Manager id : 02826bf03710200044e0bfc8bcbe5ds8 doesn't exist"`	ID を期待する SCIM 属性では、その ID は常にプロバイダーシステムの ID です。ペイロードで渡される ID がプロバイダーシステムで有効であることを確認してください。

トラブルシューティングの際に確認する領域

SCIM クライアント使用時のエラーのトラブルシューティングの際に確認可能な領域の一部を以下に示します。

- スクリプト可能 API の実行中に問題が見つかった場合は、「SCIM クライアントログ」セクションを参照してください。

ログのフィールド

フィールド	説明
要求 ID	呼び出されたスクリプト可能 API を表す固有の ID。
SCIM プロバイダー	API の呼び出し対象となるプロバイダー名。
リソース	API の呼び出し対象となるリソースの名前。
リソース ID	API の呼び出し対象となる ID。削除の場合、ID はプロバイダーシステムのリソース ID を表します。この ID はクライアントシステムにあります。
アクション	呼び出される API。

フィールド	説明
ステータス	ログのステータス。成功または失敗になります。
メッセージ	成功またはエラーメッセージ。エラーメッセージは、SCIM プロバイダーから出される場合と、SCIM クライアントの構成の問題が原因の場合があります。

- 発信コールを設定および表示して、要求の本文を確認します。詳細については、「送信 Web サービスのログ記録」を参照してください。
- コンテンツタイプを追加し、サンプルをテストし、対応するプロバイダーの REST メッセージ内の対応する [HTTP メソッド] フォームに移動して、ログのレベルを更新します。
- 要求の本文が切り捨てられている場合は、システムプロパティ `glide.outbound_http.content.max_limit` を使用して上限を増やします。

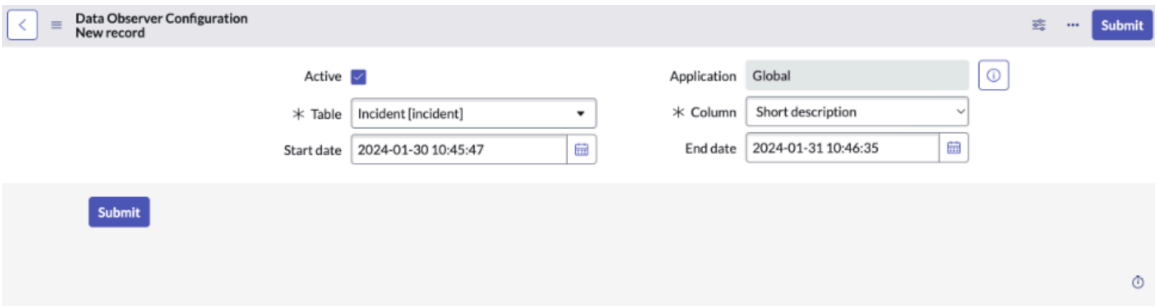
オブザーバーへのアクセス

アクセスオブザーバーを使用して、インスタンス上のユーザーとプロセスアクセスデータを把握します。

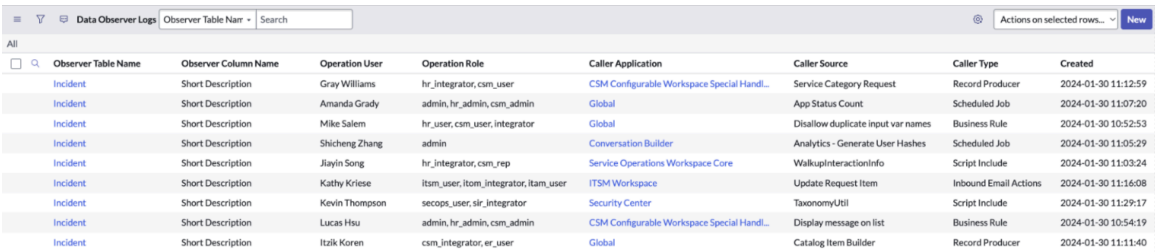
アクセスオブザーバーを使用すると、セキュリティアドミニストレーターは次のことができます。

- ユーザー、ロール、スコープ対象アプリ、スクリプトなど、データにアクセスするエンティティを把握できます。
- この予備知識を使用して、データへの不要なアクセスを制限しながら、アクセスが必要なユーザーが業務を通常どおり遂行できるように、セキュリティの最適な適用を把握できます。
- セキュリティを変更する前に、データがどのようにアクセスされているかを明確に把握することで、自動化の破綻を回避できます。
- お使いのインスタンスでの暗号化の適用方法に関する情報を提供する必要性に対処できます。

アクセスオブザーバー設定レコードを作成して、アクセスオブザーバーを設定します。これらのレコード内で、観測対象の特定のテーブルと列、および列を観測する時間を定義します。



オブザーバーログへのアクセスレコードテーブルで観察事項の結果を検索します。このテーブルでは、指定した列がアクセスされるたびに、レコードの詳細を確認できます。



Observer Table Name	Observer Column Name	Operation User	Operation Role	Caller Application	Caller Source	Caller Type	Created
Incident	Short Description	Gray Williams	hr_integrator, csm_user	CSM Configurable Workspace Special Handl...	Service Category Request	Record Producer	2024-01-30 11:12:59
Incident	Short Description	Amanda Grady	admin, hr_admin, csm_admin	Global	App Status Count	Scheduled Job	2024-01-30 11:07:20
Incident	Short Description	Mike Salem	hr_user, csm_user, integrator	Global	Disallow duplicate input var names	Business Rule	2024-01-30 10:52:53
Incident	Short Description	Shicheng Zhang	admin	Conversation Builder	Analytics - Generate User Hashes	Scheduled Job	2024-01-30 11:05:29
Incident	Short Description	Jiayin Song	hr_integrator, csm_rep	Service Operations Workspace Core	WalkupInteractionInfo	Script Include	2024-01-30 11:03:24
Incident	Short Description	Kathy Kriese	itsm_user, itom_integrator, itam_user	ITSM Workspace	Update Request Item	Inbound Email Actions	2024-01-30 11:16:08
Incident	Short Description	Kevin Thompson	secops_user, sir_integrator	Security Center	TaxonomyUtil	Script Include	2024-01-30 11:29:17
Incident	Short Description	Lucas Hsu	admin, hr_admin, csm_admin	CSM Configurable Workspace Special Handl...	Display message on list	Business Rule	2024-01-30 10:54:19
Incident	Short Description	Itzik Koren	csm_integrator, er_user	Global	Catalog Item Builder	Record Producer	2024-01-30 11:11:40

アクセス観測の構成

アクセス観測レコードを作成して、指定した期間のデータ列へのアクセスを確認します。

始める前に
必要なロール：security_admin

手順

1. 移動先 すべて > アクセスオブザーバー > アクセスオブザーバー構成.
2. [新規] を選択して、レコードを作成します。
3. フォームのフィールドに入力します。

[オブザーバー構成へのアクセス] フィールド

フィールド	説明
アクティブ	レコードが有効かどうか。観測期間はレコードが自動的にアクティブとマークされ、それ以外の場合は非アクティブになります。
アプリケーション	レコードのスコープ対象のアプリケーションです。このフィールドは読み取り専用です。
テーブル	観測する列を含むテーブル
列	選択したテーブル内の観測対象の列。
ジョブをすぐに開始	選択すると、レコードの作成と同時に観測が開始されます。
終了日時	観測が終了する時刻。
開始日時	観測を開始する時刻。このフィールドは、[ジョブをすぐに開始] が選択されていない場合のみ表示されます。

4. [送信] をクリックしてレコードを保存します。

次のタスク
定義した観測期間が開始されると、アクセスオブザーバーログ [sys_data_ob_log] テーブルのレコードで、列がアクセスされた各インスタンスを詳細に確認できます。

アクセスオブザーバーログの確認

アクセスオブザーバーログレコードの情報を使用して、データへのアクセス方法に関するインサイトを取得します。

1 つ以上のアクセス観測レコードを設定すると、インスタンスは、選択した列へのアクセスに関する詳細を含むオブザーバーログレコードの作成を開始します。これらのレコードは、インスタンス上の次の場所にあります。 すべて > アクセスオブザーバー > アクセスオブザーバーログ.

オブザーバーログへのアクセスの結果

次の表を参考にログに表示される情報を把握してください。

[オブザーバーログへのアクセス] フィールド

フィールド	説明
オブザーバーテーブル名	このレコードを生成したアクセス観測レコードで選択したテーブル。
オブザーバー列名	このレコードを生成したアクセス観測レコードで選択した列。
操作ユーザー	列にアクセスしたユーザー
操作ロール	列にアクセスしたユーザーのロール
呼び出し側アプリケーション	データへのアクセス元のスコープ対象のアプリケーション。
呼び出し側タイプ	列にアクセスした要素のタイプ (レコードプロデューサー、スケジュール済みジョブ、ビジネスルールなど)。

[オブザーバーログへのアクセス] フィールド (続く)

フィールド	説明
呼び出し側ソースドキュメント ID	
呼び出し側ソース	列にアクセスした要素。[呼び出し側タイプ] フィールドとともも、具体的に何が列にアクセスしたのかを確認するために使用されます。 たとえば、呼び出し元タイプが Business Rule の場合、呼び出し元ソースは列にアクセスしたビジネスルールの名前になります。
プライマリハッシュ	
繰り返し数	
Java スタック	
Javascript スタック	
セッション ID	列がアクセスされたセッションの ID。

プラットフォームセキュリティ製品およびソリューションに関するその他のリソース

プラットフォームセキュリティのベストプラクティス、トラブルシューティング、またはその他の実装ガイドラインを探している場合は、機能またはリソースタイプを選択して、他の関連 Web サイトで ServiceNow リソースを検出します。

注: この表の多くのリソースでは、 ServiceNow University、Now Create、 ServiceNow Community などのサイトにログインする必要があります。期待するリソースがロードされない場合は、ログインしてリソースへのアクセスを再試行してください。

リソースリンク

プラットフォームセキュリティの機能または製品	リソースタイプ	リソース
セキュリティセンター	はじめに	ServiceNow Security Center
セキュリティセンター	ベストプラクティス	• ベストプラクティス - Security Center • Security Center 強化のベストプラクティス
セキュリティセンター	FAQ	ServiceNow セキュリティ強化 - Security Center
コード署名	はじめに	• 多層防御を強化するためのコード署名 • コード署名と信頼のサークル (CoT): はじめに
データプライバシー	ヒントと例	[Washington リリース]データプライバシーによる不注意による機密データの漏洩リスクの軽減
フィールド暗号化	ヒントと例	ServiceNow のフィールドレベル暗号化
フィールド暗号化エンタープライズ	はじめに	フィールドレベル暗号化とプラットフォームレベル暗号化の違い
フィールド暗号化	トラブルシューティング	• 電話番号データタイプのフィールドにフィールドレベルの暗号化が適用されないのはなぜですか • 労務 (ER) ケースのフィールドレベルを暗号化しています • 「フィールドレベル暗号化」を使用した添付ファイルの暗号化 • 暗号化フィールド構成を削除できません • ケーステーブルは暗号化フィールド構成で使用できません • スケジュール済みジョブで暗号化フィールドをクエリすると一貫した結果が返されない

自動翻訳

リソースリンク (続く)

プラットフォームセキュリティの機能または製品	リソースタイプ	リソース
フィールド暗号化エンタープライズ	トラブルシューティング	• フィールドレベル暗号化エンタープライズ:初期セットアップ • フィールドレベル暗号化エンタープライズ:制限事項と考慮事項 • フィールドレベル暗号化エンタープライズ:管理
エッジ暗号化	トラブルシューティング	• ServiceNow のエッジ暗号化の概要 • エッジ暗号化:初期設定 • エッジ暗号化のルール • エッジ暗号化:制限事項 • エッジ暗号化:管理
エッジ暗号化	ヒントと例	• エッジ暗号化リソース • エッジ暗号化プロキシのログ • エッジ暗号化証明書の有効期限通知
エッジ暗号化	トラブルシューティング	• エッジ暗号化のパフォーマンスグラフ • エッジ暗号化によるフィールド長の拡張
データベース暗号化	ヒントと例	ServiceNow でのデータベース暗号化:CCS を使用した TSE/DBE
アクセス管理	ヒントと例	ServiceNow の ID とアクセス管理
Zero Trust アクセス	トラブルシューティング	ServiceNow のアクセスアナライザー、Zero Trust アクセス、適応認証に関するヘルプ
サービスプロバイダーのドメインセパレーション	トラブルシューティング	サービスプロバイダー向け ServiceNow ドメインセパレーション
データフィルタリング	はじめに	「Data Filtration」プラグインについて
セキュリティロール	ヒントと例	セキュリティロールはどのように機能しますか?
接続と資格情報	はじめに	接続と認証情報の概要:統合について学ぶ ServiceNow AI Platform
ServiceNow アクセス制御	ヒントと例	アクセス制御 - 簡単な方法
セキュリティセンター	ベストプラクティス	ベストプラクティス - Security Center

リソースリンク (続く)

プラットフォームセキュリティの機能または製品	リソースタイプ	リソース
プラットフォーム セキュリティ	ベストプラクティス	• ServiceNow セキュリティベストプラクティスガイド • ServiceNow AI プラットフォームの保護