



# 認証情報の 侵害リスクを緩和する ベストプラクティス

## ホワイトペーパーの目的と免責事項

このホワイトペーパーは、Snowflakeの機能を活用して強力な認証を適用し、認証情報の盗難リスクを緩和するベストプラクティスについてガイダンスをお客様に提供することを目的としています。このホワイトペーパーは、パスワードのみの認証からの脱却を目指す最新のSnowflakeの戦略について紹介する[このブログ](#)と併せて使用するよう構成されています。

このガイドのサンプルクエリは、Snowflakeのお客様をサポートする目的で挙げている例であり、実環境に実装することを目的にはしていません。

Seth Youssef Snowflake グローバルセキュリティおよびガバナンス フィールドCTO  
Anoosh Saboori Snowflake プロダクトセキュリティ担当責任者

# 目次

- 4 はじめに
- 4 Snowflake接続セッションのライフサイクル
- 5 ネットワークポリシーと認証ポリシーの全般的なガイドライン
- 6 認証ポリシーとネットワークポリシーを適用する  
ベストプラクティス
  - 6 フェーズ 1：検出
  - 9 フェーズ 2：移行計画
  - 10 フェーズ 3：保護
  - 17 フェーズ 4：継続的なモニタリング

## はじめに

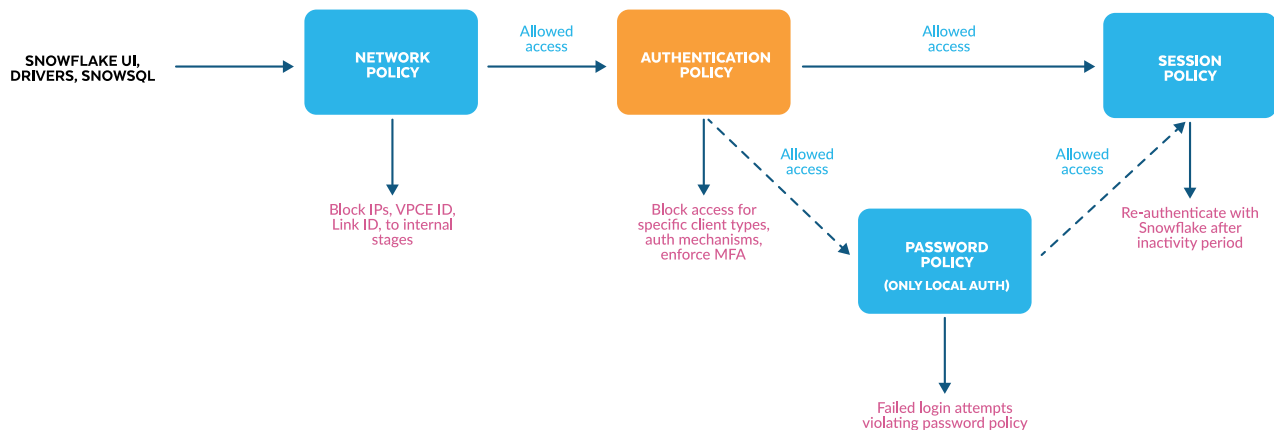
ブログ「[Snowflake管理者による必須MFAの適用](#)」で説明しているように、Snowflakeは主に以下の3つの柱にフォーカスすることでアカウントのセキュリティの維持を容易にします。

- **プロンプト**：セキュリティのベストプラクティス（多要素認証（MFA）の設定など）を使用していないユーザーに取り入れるように促す
- **適用**：管理者がデフォルトでセキュリティを適用できるようにする（すべての人間ユーザーにMFAの使用を義務付けるなど）
- **監視**：セキュリティポリシーの遵守状況を可視化する（どのユーザーがMFAを設定していないか監査するなど）

以降では、主に[Snowflake Trust Center](#)を活用した監視のベストプラクティスと、認証ポリシーとネットワークポリシーを活用した適用手順について取り上げます。

## SNOWFLAKE接続セッションのライフサイクル

Snowflakeへの接続は、以下の図のようにドライバー、コネクタ、またはUIからスタートします。



ユーザーまたはサービスがSnowflakeに接続すると、以下のようになります。

- **ネットワークポリシー**が設定されていれば、評価される。ユーザーレベルのネットワークポリシーがアカウントレベルのネットワークポリシーよりも優先されることに注意する
- **認証ポリシー**が設定されていれば、評価される。ユーザーレベルの認証ポリシーがアカウントレベルの認証ポリシーよりも優先されることに注意する
  - ユーザーまたはサービスがSnowflakeネイティブのパスワード認証を使用する場合は、**パスワードポリシー**が設定されていれば評価される
- ユーザーまたはサービスが上記の制御によって認証され、許可されると、アクティブでない期間後のユーザーの再認証方法を制御する**セッションポリシー**が評価される。ユーザーレベルのセッションポリシーがアカウントレベルのセッションポリシーよりも優先される

## ネットワークポリシーと認証ポリシーの全般的なガイドライン

お客様は、あらゆるケースにおいて以下のガイドラインを考慮に入れる必要があります（詳細は [フェーズ 3：保護セクション](#) を参照）。

- Snowflakeでは、以下を設定し、適用することを強くお勧めします。
  - PERSON（人）、SERVICE、LEGACY\_SERVICEを区別する、ユーザーのTYPE属性
    - PERSON：多要素認証（MFA）が適用されるインタラクティブな操作を行う人間用。本書の執筆時点では、TYPEを指定しない場合、デフォルトはNULLです。NULLは多要素認証（MFA）適用の観点からPERSONとして扱われます。
    - SERVICE：プログラマティックアクセス用。多要素認証（MFA）の適用からは免除されますが、これらのユーザーはパスワード認証をサポートしなくなります。また、OAuthまたはキーペアのいずれかのサポートのみになります。
    - LEGACY\_SERVICE：パスワードのみの認証をサポートする唯一のユーザータイプ（MFAの適用からは免除されます）。このユーザータイプは認証方式移行の過渡期に一時的に利用するための設定であり、お客様はネットワークポリシーによってユーザーを保護し、[漏洩したパスワードの保護機能](#)によって監視する必要があります。  
注：LEGACY\_SERVICEは、認証方式移行の過渡期に一時的に利用するための設定として使用する必要があります。2025年11月に廃止予定です。
  - お客様はSCIMを活用して、[顧客の属性によってユーザータイプ](#)を自動的にプロビジョニングし、設定できる
  - [ネットワークポリシー](#)によって、可能な限り、ユーザーとサービスが認証済みの信頼できるソースから来ていることを保証する
  - [認証ポリシー](#)によって、OAuthやSAMLの様な一時的な認証情報を使用する強力な認証手段を適用する
  - [パスワードポリシー](#)によって、その組織のパスワードポリシーを適用する
  - [セッションポリシー](#)によって、アイドルセッション時間を制限する
- サービスユーザーには、可能な限りOAuthなどの一時的な認証情報を使用することをお勧めします。また、[キーペア](#)などの長期間使用可能な認証情報を使用する場合、そうしたキーを定期的にローテーションさせて、可能な限りネットワークポリシーと組み合わせて使用することをお勧めします。
- インタラクティブな操作を行う人間のユーザーの場合、ユーザーのSnowflakeアカウントを、その組織全体のIDプロバイダー、独自の[SAML](#)フェデレーション認証、独自の多要素認証（MFA）と統合する必要があります。
- Snowflakeは、緊急時のユースケースかつネイティブパスワードを使用するユーザーには、Snowflakeネイティブの[多要素認証（MFA）](#)を使用することを強くお勧めします。
- 組織内のセキュリティポリシーに従って認証情報を定期的にローテーションさせる必要があります。
- 常に[Trust Center](#)を活用して、リスクの高いユーザーを監視し、そのSnowflakeアカウントをその組織のセキュリティオペレーションセンターと統合する必要があります。

## 認証ポリシーとネットワークポリシーを適用するベストプラクティス

Snowflakeは、以下の4つの移行ステップに従って認証基盤を強化することをお勧めします。

1. リスクの高いユーザーを検出する
2. 中断を最小限に抑えるように移行を計画する
3. Snowflake内のユーザーを保護する
4. リスクの高いユーザーがいらないか継続的に監視する

### 移行フェーズ

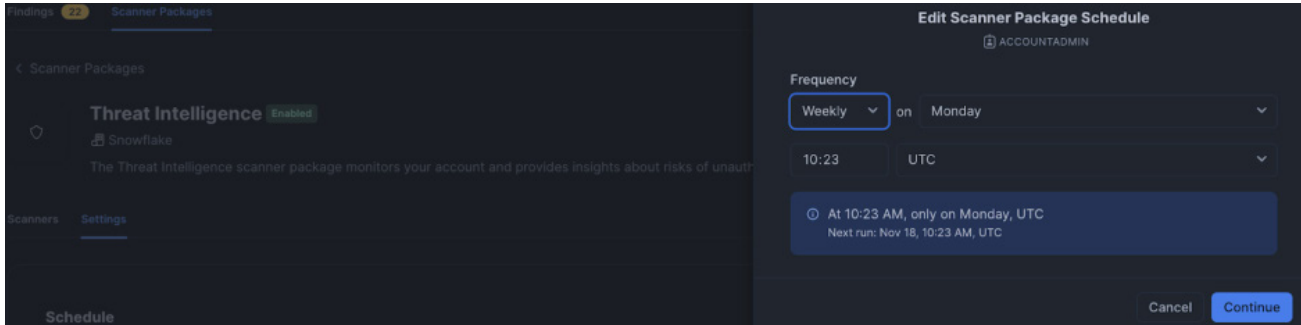


#### フェーズ 1: 検出

Snowflakeは主に以下の2つの機能を提供しています。

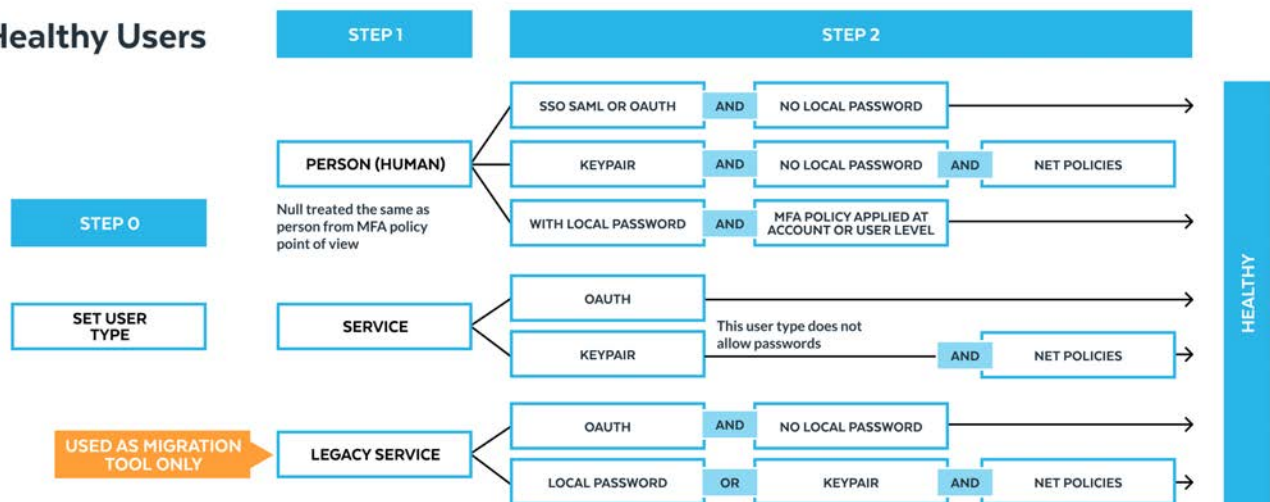
- **Threat Intelligence スキャナー**：このスキャナーは、サンプルクエリを示した次の図のようなロジックに基づいてリスクの高いユーザーを特定し、リスクの高いユーザーとリスクが高い理由を列挙します。
- **漏洩したパスワードの保護**：この機能は、ダークウェブで発見されたユーザーパスワードを検証し、自動的に無効にします。漏洩したパスワードに組み込みの保護機能を提供し、データ流出の可能性を制限します。侵害されたユーザーは、アカウント管理者に連絡して、パスワードをリセットできます。

お客様はThreat Intelligenceスキャナーを有効にし、頻繁にスキャンングをカスタマイズしてください。通常、このスキャナーを週に一度実行して、最新のリスクユーザーに関するレポートを生成してください。すべてのTrustスキャナーから得た調査結果がSNOWFLAKE.TRUST\_CENTERスキーマに保存されます。お客様はSnowflakeネイティブのアラート通知を活用して、セキュリティ管理者に自動的に通知したり、リスクの高いユーザーが検出された場合にアクションを実行することができます。



### 健全なユーザーの評価ロジック

#### Healthy Users



## リスクの高いユーザーのリスト

リスクの高いユーザーとリスクが高い理由をリストするサンプルクエリ

```
SELECT DISTINCT
  F.VALUE:ENTITY_ID::VARCHAR AS ENTITY_ID,
  F.VALUE:ENTITY_NAME::VARCHAR AS ENTITY_NAME,
  F.VALUE:ENTITY_DETAIL:USER_TYPE::VARCHAR AS USER_TYPE,
  F.VALUE:ENTITY_DETAIL:HAS_PASSWORD::VARCHAR AS HAS_PASSWORD,
  F.VALUE:ENTITY_DETAIL:HAS_RSA_PUBLIC_KEY::VARCHAR AS HAS_RSA_PUBLIC_KEY,
  F.VALUE:ENTITY_DETAIL:MFA_ENABLED::VARCHAR AS MFA_ENABLED,
  F.VALUE:ENTITY_DETAIL:ACCOUNT_AUTH_POLICY_NAME::VARCHAR AS ACCOUNT_LEVEL_AUTH_POLICY,
  F.VALUE:ENTITY_DETAIL:ACCOUNT_AUTH_POLICY_REQUIRES_MFA::VARCHAR AS ACCOUNT_LEVEL_ENFORCEMFA_POLICY,
  F.VALUE:ENTITY_DETAIL:USER_AUTH_POLICY_NAME::VARCHAR AS USER_LEVEL_AUTH_POLICY,
  F.VALUE:ENTITY_DETAIL:USER_AUTH_POLICY_REQUIRES_MFA::VARCHAR AS USER_LEVEL_ENFORCEMFA_POLICY,

  F.VALUE:ENTITY_DETAIL:ACCOUNT_NETWORK_POLICY_NAME::VARCHAR AS ACCOUNT_LEVEL_NET_POLICY_NAME,
  F.VALUE:ENTITY_DETAIL:ACCOUNT_NETWORK_POLICY_ALLOWLIST::VARCHAR AS ACCOUNT_LEVEL_NET_POLICY_ALLOWLIST,
  F.VALUE:ENTITY_DETAIL:USER_NETWORK_POLICY_NAME::VARCHAR AS USER_LEVEL_NET_POLICY_NAME,
  F.VALUE:ENTITY_DETAIL:USER_NETWORK_POLICY_ALLOWLIST::VARCHAR AS USER_LEVEL_NET_POLICY_ALLOWLIST,

FROM
  SNOWFLAKE.TRUST_CENTER.FINDINGS,
  LATERAL FLATTEN(INPUT => AT_RISK_ENTITIES) AS F
WHERE
  EVENT_ID = 'xxxxxx';
```

## USER DISTRIBUTION ACROSS TYPEおよびAUTHN METHOD USED

上記のクエリに加えて、ユーザータイプと、使用された認証手段の分布を列挙すると有益です。これを、SAMLやOAuthなど、より強力な認証手段にユーザーを移行する戦略の立案に役立てることができます。たとえば、Snowflakeにパスワードがあるユーザーがリスクと見なされている場合、そのユーザーがSAML認証のみを使用しているのであれば、できるだけ早くそのパスワードをSnowflakeから削除することをお勧めします。

```
WITH USERS AS (
  SELECT DISTINCT
    USER_ID
  , NAME
  , LOGIN_NAME
  , TYPE
  , EMAIL
FROM
  SNOWFLAKE.ACCOUNT_USAGE.USERS
WHERE
  DELETED_ON IS NULL)
SELECT
  U.USER_ID
  , A.EVENT_TIMESTAMP
  , A.USER_NAME
  , U.TYPE
  , A.REPORTED_CLIENT_TYPE
  , A.FIRST_AUTHENTICATION_FACTOR
  , A.SECOND_AUTHENTICATION_FACTOR
FROM SNOWFLAKE.ACCOUNT_USAGE.LOGIN_HISTORY AS A
JOIN USERS U
  ON A.USER_NAME = U.NAME
;
```

## フェーズ2:移行計画

**フェーズ1:検出**に基づいて移行計画が開始されます。リスクの高いユーザーを特定したら、**ネットワークポリシーと認証ポリシーの全般的なガイドライン**の計画を立て始める必要があります。可能な限りパスワードやキーペアなどの静的な認証情報から離れて、インタラクティブなユーザー（PERSON）にはSAMLまたはOAuth、プログラマティックなユーザー（SERVICE）にはOAuthなど、シングルサインオン（フェデレーション認証）を活用する必要があります。

レガシーアプリケーションをサポートするために静的な認証情報（キーペアやパスワード）を使用する必要がある場合は、計画時に以下のことを考慮する必要があります。

- 可能な限りネットワークポリシーを活用する
- 組織のポリシーに従って静的な認証情報をローテーションさせる

## 移行時の考慮事項

移行を計画する際に考慮すべきことを紹介します。

- はじめに、ユーザー**タイプ**を設定する必要があり、これは**SCIMで自動化できる**
- 2番目に、アプリケーションがどのような認証手段をサポートしているかを確認する必要があります。**Snowflakeは、OAuth、SAML、キーペア、多要素認証（MFA）などの多様な認証手段**をサポートしていますが、Snowflakeに接続するアプリケーションも強力な認証手段をサポートする必要があります。以下のような2つのユースケースが考えられます。
  - アプリケーションがすでにSAMLやOAuthをサポートしている場合、できるだけ早くその認証手段に移行することが望ましい
  - アプリケーションが古く、サポートしているのがパスワードのみで、SAMLやOAuthなどの強力な認証手段をサポートしていない場合、古いアプリケーションをアップデートすることが望ましい。アプリケーションをアップデートするまでの間は、ネットワークポリシー、パスワードのローテーション、漏洩したパスワードの保護機能を活用する
- 次に、ローカルユーザー（Snowflakeで手動で作成され、SAMLもOAuthも有効でないユーザー）のために以下のことを検討する必要があります
  - SSOをサポートしているアプリケーションについては、ローカルユーザーをSSOが可能なユーザーに切り替え、ユーザーを切り替える際のダウンタイムを考慮に入れる
  - ローカルユーザーをSSOが可能なユーザーに切り替えるには、そうしたユーザーがIdPに登録されていることを確認し、Snowflakeで手動で、またはできればSCIMで自動的にプロビジョニングする必要がある
  - 使用されていないローカルユーザーを無効にする
- Snowflakeは、ユーザーレベルとアカウントレベル両方の認証ポリシー、ネットワークポリシー、パスワードポリシーをサポートしている。まずユーザーレベルのポリシーについて検討して、徐々に移行していく必要がある（ユーザーレベルのポリシーはアカウントレベルのポリシーよりも優先される）
  - サービスユーザー（TYPE = SERVICEまたはLEGACY\_SERVICE）を使用するアプリケーションについては、ユーザーレベルのネットワークポリシーが望ましい
  - 人間のユーザー（TYPE = PERSONまたはNULL）については、ユーザーレベルのネットワークポリシーから始め、その後にアカウントレベルでネットワークポリシーを適用することで、ユーザーレベル固有のポリシーが存在しないユーザー層すべてを保護できる
  - 多要素認証（MFA）ポリシーと同じコンセプトを、まずユーザーレベルのポリシーから開始する

### フェーズ3:保護

以下のステップに従ってSnowflakeアカウントの侵害リスクを緩和してください。

1. ユーザータイプを設定する
2. 不要であればローカルパスワードを削除する
3. サービスユーザー向けの認証ポリシーを作成する
4. 人間のユーザーに多要素認証（MFA）を適用する認証ポリシーを作成する
5. パスワードポリシーを作成する
6. セッションポリシーを作成する
7. サービスユーザー向けのネットワークポリシーを作成する
8. アカウントレベルのネットワークポリシーを作成する
9. サービスユーザーを保護する
10. アカウントレベルでパスワードポリシーを適用する
11. アカウントレベルでセッションポリシーを適用する
12. サービスユーザーをテストする
13. 多要素認証（MFA）の適用によってアカウントを保護する
14. アカウントレベルのネットワークポリシーを適用する
15. 使用されていないユーザーを無効にする
16. Snowflake Trust Centerまたはセキュリティモニタリングソリューションを活用して常に監視する

### ユーザータイプを設定する

前述のとおり、保護フェーズではまず、**SCIMで自動的に**、または手動でユーザータイプを設定します。

```
ALTER USER SVC_USER1 SET TYPE = SERVICE
ALTER USER USER1@HUMAN.COM SET TYPE = PERSON
-- LEGACY APPLICATION ONLY
ALTER USER SVC_USER2 SET TYPE = LEGACY_SERVICE
```

また、アカウントの作成時に管理ユーザータイプを設定できるようになりました。

```
CREATE ACCOUNT <NAME> [ ADMIN_USER_TYPE = PERSON | SERVICE | LEGACY_SERVICE | NULL ]
```

ヒント：通常、多くのお客様のサービスユーザー名には一定のパターン（Local\_Svc\_user1など）があり、この命名パターンを使用してSERVICEタイプを大規模に設定できます。

## 不要であればローカルパスワードを削除する

User Distribution Across TYPEおよびAuthN Method Usedのクエリと、Trust Centerで得た調査結果に基づく [リスクの高いユーザーのリスト](#)を活用して、Snowflake内でSAML、OAuth、またはキーペアのみを使用しているユーザーのパスワードの削除を開始します。

## サービスユーザー向けの認証ポリシーを作成する

Snowflakeは、プログラマティックサービスユーザーにはOAuthを使用することをお勧めします。以下のように認証ポリシーによってOAuthを適用できます。

```
CREATE AUTHENTICATION POLICY PROGRAMMATIC_ACCESS_USER_AUTH
  CLIENT_TYPES = ('DRIVERS', 'SNOWSQL')
  AUTHENTICATION_METHODS = ('OAUTH')
  SECURITY_INTEGRATIONS = ('<OAUTH SECURITY INTEGRATIONS>');

ALTER USER <SERVICE_USER> SET AUTHENTICATION POLICY PROGRAMMATIC_ACCESS_USER_AUTH;
```

キーペアを使用してSERVICEタイプユーザーによる認証に対応することもできますが、ネットワークポリシーと組み合わせて、キーを定期的にローテーションさせる必要があります。

注：キーペアもOAuthもサポートしていないレガシーシステムがあり、認証にパスワードを使用する必要がある場合は、「PASSWORD」を認証手段とした追加の認証ポリシーを作成し、それを特定のプログラマティックユーザーに適用します。それを [ネットワークポリシーと認証ポリシーの全般的なガイドライン](#)と組み合わせて使用します。

## 人間のユーザーに多要素認証 (MFA) を適用する認証ポリシーを作成する

Snowflakeは、お客様独自のSAML IdPと、そのIdPがサポートする多要素認証 (MFA) ソリューションを使用することをお勧めします。以下の認証ポリシーの例は、以下の達成をサポートするものです。

- Snowflakeネイティブの多要素認証 (MFA) を、Snowflakeネイティブのパスワードを使用する人間のユーザーに適用する
- お客様のSAML IdPを使用して、シングルサインオンユーザーに多要素認証 (MFA) を適用する

```
CREATE AUTHENTICATION POLICY HUMAN_ACCESS_ACCOUNT_ENFORCE_MFA
  AUTHENTICATION_METHODS = ('SAML', 'PASSWORD')
  SECURITY_INTEGRATIONS = ('<SAML SECURITY INTEGRATIONS>');
  MFA_AUTHENTICATION_METHODS = ('PASSWORD'); -- ENFORCE SF MFA FOR NATIVE PASSWORDS ONLY
  MFA_ENROLLMENT = 'REQUIRED'
```

緊急時について考慮し、お客様のIdPがオフラインになってもアカウント管理者がSnowflakeアカウントにログインできるようにする必要があります。

```
CREATE AUTHENTICATION POLICY ACCOUNTADMIN_BREAKGLASS_MFA
  AUTHENTICATION_METHODS = ('PASSWORD')
  MFA_AUTHENTICATION_METHODS = ('PASSWORD'); -- ENFORCE SF MFA FOR NATIVE PASSWORDS ONLY
  MFA_ENROLLMENT = 'REQUIRED'
  COMMENT = '<string_literal>'
```

## 二重のMFA

注：より厳格なポリシーにするために、追加の多要素認証（MFA）適用ポリシーを作成して、ユーザーレベルで直接適用できます。たとえば、お客様のIdPがMFAも二重のMFAもサポートしていない場合は、IdPレベルのMFAの適用状況に関わらず、Snowflake MFAをユーザーに適用します（一部のお客様はこのオプションを使用して、アカウント管理者など権限の高いユーザーに二重のMFAを適用できます）。

```
CREATE AUTHENTICATION POLICY ACCOUNTADMIN_BREAKGLASS_MFA
AUTHENTICATION_METHODS = ('PASSWORD')
MFA_AUTHENTICATION_METHODS = ('PASSWORD'); -- ENFORCE SF MFA FOR NATIVE PASSWORDS ONLY
MFA_ENROLLMENT = 'REQUIRED'
```

## パスワードポリシーを作成する

レガシーシステムがある場合や、緊急時にSnowflakeネイティブのパスワードを使用する必要がある場合は、Snowflakeの[パスワードポリシー](#)を活用し、それがデフォルトのポリシーと異なる場合は組織のパスワードポリシーと一致させてください。

```
CREATE PASSWORD POLICY password_policy_account
PASSWORD_MIN_LENGTH = 32
--PASSWORD_MAX_LENGTH = <integer>
PASSWORD_MIN_UPPER_CASE_CHARS = 6
PASSWORD_MIN_LOWER_CASE_CHARS = 6
PASSWORD_MIN_NUMERIC_CHARS = 4
PASSWORD_MIN_SPECIAL_CHARS = 8
PASSWORD_MIN_AGE_DAYS = 10
PASSWORD_MAX_AGE_DAYS = 30
PASSWORD_MAX_RETRIES = 3
PASSWORD_LOCKOUT_TIME_MINS = 30
PASSWORD_HISTORY = 24
COMMENT = '<string_literal>'
```

## セッションポリシーを作成する

Snowflakeは、セッションポリシーを作成して、一定のアクティブでない期間後に再認証を適用することをお勧めします。これはあくまで例で、個々のユーザーレベルでポリシーをカスタマイズできます。

```
CREATE SESSION POLICY session_policy_account
SESSION_IDLE_TIMEOUT_MINS = 240 -- Snowflake Clients and programmatic clients
SESSION_UI_IDLE_TIMEOUT_MINS = 20 -- For the Snowflake web interface
COMMENT = '<string_literal>'
```

## サービスユーザー向けのネットワークポリシーを作成する

通常、サービスユーザーやプログラマティックアクセスユーザーは、承認済みのIPアドレス（またはプライベート接続の場合はVPCE IDやLinkIDなど）からアクセスする必要があります。

Snowflakeは、サービスユーザーレベルのネットワークポリシーを作成して、承認済みのソースや信頼できるソースからのプログラマティックアクセスユーザーへのアクセスを制限することをお勧めします。ネットワークルールを内部ステージに適用することも検討する必要があります。

注：内部ステージのネットワークルールは、AWSのSnowflakeのみでサポートされています。Azureについては、パブリックアクセスのブロックを検討できます。GCPでサービスを制御する場合は、Snowflakeサポートにお問い合わせください。

注：クラウドの動的な性質により、クラウドプロバイダーによっては、Snowflakeのネットワークポリシーに列挙する必要がある一連のIPアドレスを提供できません。その場合、ネットワークポリシーと認証ポリシーの全般的なガイドラインに従ってください。または、使用するツールがプライベート接続をサポートしている場合は、プライベート接続を検討してください。

お客様がプライベート接続を使用しているかどうかに応じて、プライベートネットワークで接続する場合とパブリックネットワークから接続する場合があります。適切なIP（パブリックまたはプライベート）やCSPタグ（VPCE IDやLinkIDなど）を含めたネットワークルールを複数追加することで、パブリック接続とプライベート接続の両方を同時に許可することに注意してください。

```
-- ACCESS FROM PUBLIC IP ADDRESSES
CREATE NETWORK RULE PROGRAMMATIC_ACCESS_USER_NET_RULE_PUBLIC
  TYPE = IPV4
  VALUE_LIST = ( 'PUBLIC IP1' , 'XX.XX.XX.XX/24' , ... ] )
  MODE = INGRESS
;

-- ACCESS FROM PRIVATE NETWORK
CREATE NETWORK RULE PROGRAMMATIC_ACCESS_USER_NET_RULE_PL
  TYPE = AWSVPCEID
  VALUE_LIST = ( 'VPCE-123ABC3420C1931' , 'VPCE-123ABC3420C1932' )
  MODE = INGRESS
;

-- RESTRICT ACCESS TO INTERNAL STAGE USING VPCE ID
CREATE NETWORK RULE PROGRAMMATIC_ACCESS_USER_NET_RULE_INTERNAL_STAGE
  TYPE = AWSVPCEID
  VALUE_LIST = ( 'VPCE-123ABC3420C1933' )
  MODE = INTERNAL_STAGE
;

CREATE NETWORK POLICY PROGRAMMATIC_ACCESS_USER_NET_POLICY
  ALLOWED_NETWORK_RULE_LIST =
(
  'PROGRAMMATIC_ACCESS_USER_NET_RULE_PUBLIC' ,
  'PROGRAMMATIC_ACCESS_USER_NET_RULE_PL' ,
  'PROGRAMMATIC_ACCESS_USER_NET_RULE_INTERNAL_STAGE'
)
  --BLOCKED_NETWORK_RULE_LIST = ( 'OPTIONAL BLOCKED LIST OF IPS' )
;'
```

## アカウントレベルのネットワークポリシーを作成する

アカウントレベルのポリシーは、ネットワークポリシーが直接適用されていないユーザーの、デフォルトのセキュリティネットワークポリシーとして機能します。ベストプラクティスは、このポリシーをできるだけ限定的かつ短くし、ユーザーレベルのポリシーを使用して特定のユーザーニーズに対応することです。

注：Snowflakeは、組織のあらゆるニーズに対応するために膨大なアカウントレベルのネットワークポリシーを作成することはお勧めしません。代わりに、ユーザーレベルのポリシーによってきめ細かく制御できるようにします。

ユーザーレベルのネットワークポリシーと同様に、お客様が**プライベート接続**を使用しているかどうかに応じて、プライベートネットワークで接続する場合とパブリックネットワークから接続する場合があります。適切なIP（パブリックまたはプライベート）やCSPタグ（VPCE IDやLinkIDなど）を含めたネットワークルールを複数追加することで、パブリック接続とプライベート接続の両方を同時に許可できることに注意してください。

```
-- ACCESS FROM PUBLIC IP ADDRESSES
CREATE NETWORK RULE HUMAN_ACCESS_ACCOUNT_NET_RULE_PUBLIC
  TYPE = IPV4
  VALUE_LIST = ( 'PUBLIC IP1' , 'XX.XX.XX.XX/24' , ... )
  MODE = INGRESS
;

-- ACCESS FROM PRIVATE NETWORK
CREATE NETWORK RULE HUMAN_ACCESS_ACCOUNT_NET_RULE_PL
  TYPE = AWSVPCEID
  VALUE_LIST = ( 'VPCE-123ABC3420C1934' , 'VPCE-123ABC3420C1936' )
  MODE = INGRESS
;

-- RESTRICT ACCESS TO INTERNAL STAGE USING VPCE ID
CREATE NETWORK RULE HUMAN_ACCESS_ACCOUNT_NET_RULE_INTERNAL_STAGE
  TYPE = AWSVPCEID
  VALUE_LIST = ( 'VPCE-123ABC3420C1937' )
  MODE = INTERNAL_STAGE
;

CREATE NETWORK POLICY ACCOUNT_LEVEL_NET_POLICY
  ALLOWED_NETWORK_RULE_LIST =
(
  'HUMAN_ACCESS_ACCOUNT_NET_RULE_PUBLIC' ,
  'HUMAN_ACCESS_ACCOUNT_NET_RULE_PL' ,
  'HUMAN_ACCESS_ACCOUNT_NET_RULE_INTERNAL_STAGE'
)
  --BLOCKED_NETWORK_RULE_LIST = ( 'OPTIONAL BLOCKED LIST OF IPS' )
;
```

## サービスユーザーを保護する

TYPE=SERVICEのユーザーは、インタラクティブでないユースケースのセキュリティ状況を改善するため、アカウントレベルの多要素認証（MFA）適用ポリシーが免除され、制限が適用されます。特にSERVICEタイプのユーザーはパスワードやSAML SSOを使用してログインできません。以下の注意事項を参照してください。

```
-- FOR EVERY SERVICE PROGRAMMATIC ACCESS USER
ALTER USER SERVICE_USER_1 SET
  TYPE = SERVICE
  NETWORK_POLICY = PROGRAMMATIC_ACCESS_USER_NET_POLICY
  AUTHENTICATION_POLICY = PROGRAMMATIC_ACCESS_USER_AUTH;
```

**注意：**SERVICEタイプのユーザーは認証手段としてパスワードを使用できないため、より強力な認証形式をサポートしていないレガシーシステムを使用している場合、LEGACY\_SERVICEというユーザータイプを使用することをお勧めします。LEGACY\_SERVICEユーザーは、多要素認証（MFA）の対象外ですが、引き続きパスワード認証を使用できます。

この2つの新しいユーザータイプ（SERVICEとLEGACY\_SERVICE）の使用をお勧めするのは、Trust Centerがこれらのユーザーを多要素認証（MFA）ポリシーモニタリングから除外するためです（認証ポリシーによる除外では、MFAポリシーモニタリングからは除外されません）。Snowflakeは、ツールをアップグレードしてより強力な認証手段に移行すること、および移行が完了するまではLEGACY\_SERVICEタイプを一時的な解決策として使用することをお勧めします。パスワードの定期的なローテーションに加えて、login\_historyとquery\_historyを使用してユーザーの活動を常に監視する必要があります。

**注：**LEGACY\_SERVICEは2025年11月に廃止予定であることに注意してください。

## アカウントレベルでパスワードポリシーとセッションポリシーを適用する

Snowflakeセキュリティ管理者は、アカウントレベルでベースラインパスワードとセッションポリシーを適用し、その後、必要に応じてこうしたポリシーをユーザーレベルのポリシーによって上書きする必要があります。

セキュリティ管理者は、サービスユーザーを数名テストし、正常に動作することを確認、接続が信頼できるソースからのものであり、適切な認証手段が使用されていることを確認する必要があります。管理者はTrust Centerに加えてLOGIN\_HISTORYを使用して、ネットワークポリシーによってサービスユーザーが保護されていることを確認する必要があります。

```
ALTER ACCOUNT SET
  SESSION_POLICY = SESSION_POLICY_ACCOUNT;
  PASSWORD_POLICY = PASSWORD_POLICY_ACCOUNT;
```

## サービスユーザーをテストする

この段階で、アカウントにはパスワードポリシーとセッションポリシーが適用されています。サービスプログラマティックユーザーは、多要素認証（MFA）が免除され、また信頼できるソースからの接続と推奨されている認証手段（OAuth、キーペアなど）の使用を確実にするための、独自の特定の認証ポリシーとネットワークポリシーが適用されています。

セキュリティ管理者はサービスユーザーを数名テストして、スムーズに運用されていることを確認し、信頼できるソースから接続されており、適切な認証手段を使用していることを確認する必要があります。管理者はTrust Centerに加えてLOGIN\_HISTORYを使用して、ネットワークポリシーによってサービスユーザーが保護されていることを確認する必要があります。

```
SELECT *
FROM TABLE(INFORMATION_SCHEMA.LOGIN_HISTORY(TIME_RANGE_START =>
DATEADD('HOURS',-1,CURRENT_TIMESTAMP()),CURRENT_TIMESTAMP()))
ORDER BY EVENT_TIMESTAMP;
```

## 多要素認証 (MFA) の適用によってアカウントを保護する

SERVICE TYPEでないユーザーすべてに多要素認証（MFA）を適用するために、MFA適用ポリシーをアカウントレベルで適用します。

これらのポリシーによって、人間のインタラクティブユーザーすべてが自身のIdPまたはネイティブのSnowflake MFAのいずれかからMFAを確実に有効化できます。

```
ALTER ACCOUNT SET
AUTHENTICATION POLICY = HUMAN_ACCESS_ACCOUNT_ENFORCE_MFA;
```

アカウント管理者など権限の高いユーザーに**二重のMFA**を適用する場合は、ユーザーレベルで多要素認証（MFA）を適用します。ただし、IdPがダウンした場合の緊急時の手順と、セキュリティ要件とのバランスを取る必要があります。

```
ALTER USER SUPPER_PROTECTED_ACCOUNTADMIN_1
AUTHENTICATION POLICY = ACCOUNTADMIN_DOUBLE_MFA;
```

緊急の場合：

```
ALTER USER BREAKGLASS_ACCOUNTADMIN_1
AUTHENTICATION POLICY = ACCOUNTADMIN_BREAKGLASS_MFA;
```

## アカウントレベルのネットワークポリシーを適用する

最後に、アカウントレベルでネットワークポリシーを適用して、明示的にネットワークポリシーが適用されていない、その他のユーザーすべてを保護します。

```
ALTER ACCOUNT SET
  NETWORK_POLICY = ACCOUNT_LEVEL_NET_POLICY;
```

## 使用されていないユーザーを無効にする

Trust Center CISスキャナー（1.8）は、直近90日間にアクティブでないユーザーを監視します。以下の図のように、「ワークシートを開く」をクリックして、アクティブでないユーザーをリストするクエリを表示できます。

```
SELECT
  F.VALUE:ENTITY_ID::VARCHAR AS ENTITY_ID,
  F.VALUE:ENTITY_NAME::VARCHAR AS ENTITY_NAME,
  F.VALUE:ENTITY_OBJECT_TYPE::VARCHAR AS ENTITY_OBJECT_TYPE,
  F.VALUE:ENTITY_DETAIL AS ENTITY_DETAIL
FROM
  SNOWFLAKE.TRUST_CENTER.FINDINGS,
  LATERAL FLATTEN(INPUT => AT_RISK_ENTITIES) AS F
```

対象ユーザーを無効にすることを強くお勧めします。

Ensure that users who did not log in for 90 days are disabled →

Event ID: 49804

Summary Remediation

Details

Last Scan Time 11/20/24, 8:22:02 AM

Severity **Medium**

Scanner [CIS\\_BENCHMARKS\\_CIS1.8](#)

Scanner Package CIS Benchmarks

Description Access grants tend to accumulate over time unless explicitly set to expire. Regularly revoking unused access grants and disabling inactive user accounts is a good countermeasure to this dynamic. If credentials of an inactive user account are leaked or stolen, it may

Show More

Audit Result

79 entities were found to have violated the benchmark during the last auditing.

Preview of sample objects:

## フェーズ 4: 継続的なモニタリング

Trust Center Threat Intelligenceスキャナーパッケージを活用して、多要素認証（MFA）とネットワークポリシーの適用状況を監視します。Snowflakeは、[漏洩したパスワードの保護機能](#)を使用して、盗難された認証情報がないかダークウェブを監視し、ダークウェブで見つかったパスワードと一致するパスワードハッシュを持つユーザーを無効にします。お客様は、Trust Centerに加えて、Snowflakeネイティブの[アラート機能とカスタムクエリ](#)を使用して、以下に対応する追加のカスタムアラートを作成できます。

- タイプが具体的に設定されていないユーザー
- パスワードやキーペアなど、静的な認証情報を使用するアプリケーション
- LEGACY\_SERVICEが追加された新規ユーザー
- レガシーアプリケーションを定期的にアップグレードして、より強力な認証方法（AuthN）を使用する

# SNOWFLAKEについて

Snowflakeは、シンプルかつ効率的で信頼性の高いエンタープライズAIを実現します。SnowflakeのAIデータクラウドは、世界最大規模の数百の企業を含む世界中の数多くのお客様に利用されており、データ共有、AI/機械学習アプリケーションの構築、ビジネスの強化に貢献しています。これからは、エンタープライズAIの時代です。

詳しくは、[snowflake.com/ja](https://snowflake.com/ja)（ニューヨーク証券取引所：SNOW）をご覧ください。



© 2024 Snowflake Inc. All rights reserved. Snowflake、Snowflakeのロゴ、および本書に記載されているその他すべてのSnowflakeの製品、機能、サービス名は、米国およびその他の国におけるSnowflake Inc.の登録商標または商標です。本書で言及または使用されているその他すべてのブランド名またはロゴは、識別目的でのみ使用されており、各所有者の商標である可能性があります。Snowflakeが、必ずしもかかる商標所有者と関係を持ち、または出資や支援を受けているわけではありません。