Red Hat Enterprise Linux 9

Identity Management の計画

IdM 環境のインフラストラクチャーとサービスの統合計画

Last Updated: 2024-06-27

Red Hat Enterprise Linux 9 Identity Management の計画

IdM 環境のインフラストラクチャーとサービスの統合計画

法律上の通知

概要

Identity Management (IdM) は、アイデンティティーストア、認証、認可ポリシーを一元管理する方法を提供します。 環境への IdM の統合を成功させるには、IdM のコンポーネントについて学び、インストールを計画してください。たとえば、フェイルオーバーとロードバランシング、Active Directory (AD) への統合、DNS ゾーンと認証局 (CA) の構造、バックアップとリカバリーのシナリオのためのレプリケーショントポロジーを計画します。

目次

RED HAT ドキュメントへのフィードバック (英語のみ)

Red Hat ドキュメントに関するご意見やご感想をお寄せください。また、改善点があればお知らせください。

Jira からのフィードバック送信 (アカウントが必要)

1. Jira の Web サイトにログインします。
2. 上部のナビゲーションバーで **Create** をクリックします。
3. **Summary** フィールドにわかりやすいタイトルを入力します。
4. **Description** フィールドに、ドキュメントの改善に関するご意見を記入してください。ドキュメントの該当部分へのリンクも追加してください。
5. ダイアログの下部にある **Create** をクリックします。

第1章 RHEL における IDM とアクセス制御の概要

Identity Management (IdM) を使用して、アイデンティティー管理の一元化、セキュリティー制御の適用、ベストプラクティスとセキュリティーポリシーへの準拠を行う方法を説明します。Linux 環境と Windows 環境の両方における IdM 実装の一般的なお客様のシナリオと解決策を紹介します。

1.1. IDM の概要

Identity Management (IdM) は、Linux ベースのドメインでアイデンティティーストア、認証、ポリシー、認可ポリシーを一元管理する方法を提供します。

Red Hat Enterprise Linux における IdM の目的

IdM は、異なるサービスを個別に管理するオーバーヘッドと、異なるマシンで異なるツールを使用するオーバーヘッドを大幅に削減します。

IdM は、以下に対応する数少ない集中型 ID、ポリシー、および認証ソフトウェアです。

- Linux オペレーティングシステム環境の高度な機能
- Linux マシンの大規模なグループの一元化
- Active Directory とのネイティブな統合

IdM は、Linux ベースおよび Linux 制御のドメインを作成します。

- IdM は、既存のネイティブ Linux ツールとプロトコルを基盤とします。独自のプロセスと設定がありますが、その基盤となる技術は Linux システムで十分に確立されおり、Linux 管理者から信頼されています。
- IdM サーバーおよびクライアントは Red Hat Enterprise Linux マシンです。IdM クライアントは、標準プロトコルに対応してさえいれば別の Linux および UNIX のディストリビューションにすることもできます。Windows クライアントは IdM ドメインのメンバーにはなれませんが、Active Directory (AD) が管理する Windows システムにログインしているユーザーは、Linux クライアントに接続したり、IdM が管理するサーバーにアクセスしたりできます。これは、AD ドメインと IdM ドメインとの間に、フォレスト間の信頼関係を確立することで実現します。

複数の Linux サーバーにおける ID およびポリシーの管理

IdM を使用しない場合 - 各サーバーが個別に管理されます。パスワードはすべてローカルマシンに保存されます。IT 管理者は、すべてのマシンでユーザーを管理し、認証ポリシーおよび認可ポリシーを別々に設定し、ローカルパスワードを維持します。ただし、多くの場合は、AD を用いた直接統合など、その他の集中型ソリューションを使用することになります。システムは、複数のソリューションを使用して AD に直接統合できます。

- レガシーの Linux ツール (使用は推奨されません)
- Samba winbind に基づくソリューション (特定のユースケースでのみ推奨)
- サードパーティー製ソフトウェアに基づくソリューション (通常は、他のベンダーのライセンスが必要)
- SSSD に基づくソリューション (ネイティブ Linux と、ほとんどのユースケースに推奨)

IdM を使用する場合 - IT 管理者は以下が可能になります。

- ID を一か所で管理 - IdM サーバー

- 複数のマシンで同時にポリシーを均一に適用
- ホストベースのアクセス制御、委譲などのルールを使用してユーザーに異なるアクセスレベルを設定
- 権限昇格ルールの一元管理
- ホームディレクトリーのマウント方法の定義

エンタープライズ SSO

IdM Enterprise の場合、シングルサインオン (SSO) は Kerberos プロトコルを使用して実装されます。このプロトコルは、インフラストラクチャーレベルで一般的であり、SSH、LDAP、NFS、CUPS、DNS などのサービスで SSO を有効にします。別の Web スタック (Apache、EAP、Django など) を使用した Web サービスでも、SSO に Kerberos を使用できます。ただし、実際には、Web アプリケーションには SSO を基にした OpenID Connect または SAML を使用する方が便利です。2 つの層をブリッジするには、Kerberos 認証を OpenID Connect チケットまたは SAML アサーションに変換できる Identity Provider (IdP) ソリューションをデプロイすることが推奨されます。このような IdP の一例として、Keycloak オープンソースプロジェクトをベースとした Red Hat SSO テクノロジーがあります。

IdM を使用しない場合 - ユーザーはシステムにログインし、サービスやアプリケーションにアクセスする度にパスワードを求められます。これらのパスワードは異なる場合もあるため、アプリケーションごとに使用する認証情報を覚えている必要があります。

IdM を使用する場合 - システムにログインすると、認証情報を繰り返し聞かれることなく、複数のサービスやアプリケーションにアクセスできます。これにより、以下が可能になります。

- ユーザービリティーの向上
- パスワードを書き留めたり安全でない場所に保存したりすることによるセキュリティーリスクの低減
- ユーザーの生産性向上

Linux と Windows の混合環境の管理

IdM を使用しない場合 - Windows システムは AD フォレストで管理されますが、開発、実稼働環境などのチームは Linux システムを多数使用します。Linux システムは、AD 環境から除外されます。

IdM を使用する場合 - IT 管理者は以下が可能になります。

- ネイティブの Linux ツールを使用して Linux システムを管理する
- Active Directory により一元管理されている環境に Linux システムを統合して、一元管理されたユーザーストアを保護する
- 規模に応じて、または必要に応じて、新しい Linux システムを簡単にデプロイする
- 他のチームに依存することなく遅延を回避しながら、ビジネスニーズに迅速に対応し、Linux インフラストラクチャーの管理に関連する決定を下す

IdM と標準 LDAP ディレクトリーの比較

Red Hat Directory Server などの標準 LDAP ディレクトリーは汎用ディレクトリーで、幅広いユースケースに適用するようにカスタマイズできます。

- スキーマー - ユーザー、マシン、ネットワークエンティティー、物理的設備、建物といった非常に幅広いエントリー用にカスタマイズ可能な柔軟性のあるスキーマー

- 典型的な使用例 - インターネット上でサービスを提供するビジネスアプリケーションなど、他のアプリケーションのデータを保存するバックエンドのディレクトリー

IdM には、企業内 ID と、その ID に関連する認証ポリシーおよび認可ポリシーを管理するという特定の目的があります。

- スキーマー - ユーザーやマシンの ID のエントリーといった特定の目的に関連するエントリーセットを定義する特定のスキーマ
- 典型的な使用例 - 企業やプロジェクトの境界内におけるアイデンティティーを管理する ID および認証サーバー

Red Hat Directory Server と IdM では、基礎となるディレクトリーサーバーの技術は同じです。ただし、IdM は企業内の ID 管理用に最適化されています。これにより全般的な拡張性は制限されますが、シンプルな設定、リソース管理の自動化の改善、企業の ID 管理における効率性の向上などの利点がもたらされます。

関連情報

- Red Hat Enterprise Linux Blog のブログ投稿 Identity Management or Red Hat Directory Server – Which One Should I Use?
- 標準プロトコル に関するナレッジベースの記事

1.2. 一般的な IDM のお客様のシナリオとその解決策

Linux 環境と Windows 環境の両方における一般的なアイデンティティー管理とアクセス制御のユースケースとその解決策の例を紹介します。

シナリオ 1

状況

あなたは会社の Windows 管理者です。
Windows システムとは別に、管理する Linux システムもいくつかあります。

環境のどの部分の制御も Linux 管理者に委任できないため、Active Directory (AD) ですべてのセキュリティー制御を処理する必要があります。

解決方法

Linux ホストを AD に直接統合します。
sudo ルールを LDAP サーバーで一元的に定義する場合は、AD ドメインコントローラー (DC) にスキーマ拡張を実装する必要があります。この拡張を実装する権限がない場合は、Identity Management (IdM) のインストールを検討してください。以下の「シナリオ 3」を参照してください。IdM にはすでにスキーマ拡張が含まれているため、IdM で直接 **sudo** ルールを管理 できます。

将来さらに Linux のスキルが必要になると予想される場合のさらなるアドバイス

Linux コミュニティーに接続して、他のユーザーがユーザー、ホスト、サービスなどの ID をどのように管理しているかを確認してください。
ベストプラクティスを調査します。

Linux に慣れてください。

- 可能な限り、RHEL Web コンソール を使用してください。

- 可能な限り、コマンドラインで簡単なコマンドを使用してください。
- Red Hat システム管理コースに参加してください。

シナリオ 2

状況

あなたは会社の Linux 管理者です。
Linux ユーザーには、会社のリソースへのさまざまなレベルのアクセスが必要です。

Linux マシンの厳密で一元的なアクセス制御が必要です。

解決方法

IdM をインストール し、ユーザーをそこに移行します。

あなたの会社が将来的に拡大することを期待している場合のさらなるアドバイス

IdM をインストールしたら、ホストベースのアクセス制御 と sudo ルール を設定します。これらは、制限されたアクセスと最小限の特権のセキュリティーのベストプラクティスを維持するために必要です。
セキュリティー目標を達成するために、プロトコルを使用してインフラストラクチャー層とアプリケーション層の両方を保護する一貫したアイデンティティーおよびアクセス管理 (IAM) 戦略を策定します。

シナリオ 3

状況

あなたは会社の Linux 管理者であり、Linux システムを会社の Windows サーバーと統合する必要があります。Linux システムへのアクセス制御の唯一のメンテナーであり続けたいと考えています。ユーザーが異なれば、Linux システムへのアクセスレベルも異なりますが、それらはすべて AD に存在します。

解決方法

AD 制御は十分に堅牢ではないため、Linux 側で Linux システムへのアクセス制御を設定する必要があります。IdM をインストール し、IdM-AD 信頼を確立 します。

環境のセキュリティーを強化するためのさらなるアドバイス

IdM をインストールしたら、ホストベースのアクセス制御 と sudo ルール を設定します。これらは、制限されたアクセスと最小限の特権のセキュリティーのベストプラクティスを維持するために必要です。
セキュリティー目標を達成するために、プロトコルを使用してインフラストラクチャー層とアプリケーション層の両方を保護する一貫したアイデンティティーおよびアクセス管理 (IAM) 戦略を策定します。

シナリオ 4

状況

セキュリティー管理者は、すべての Red Hat 製品を含むすべての環境で ID とアクセスを管理する必要があります。すべての ID を 1 か所で管理し、すべてのプラットフォーム、クラウド、製品にわたってアクセス制御を維持する必要があります。

解決方法

IdM、Red Hat Single Sign-On、Red Hat Satellite、Red Hat Ansible Automation Platform などの Red Hat 製品を統合します。

シナリオ 5

状況

国防総省 (DoD) またはインテリジェンスコミュニティー (IC) 環境のセキュリティーおよびシステム管理者は、スマートカードまたは RSA 認証を使用する必要があります。PIV 証明書または RSA トークンを使用する必要があります。

解決方法

1. IdM で証明書マッピングを設定 します。
2. IdM-AD 信頼が存在する場合は、GSSAPI 委任が有効になっていることを確認してください。
3. IdM で RSA トークンの radius 設定の使用を設定します。
4. スマートカード認証 用に IdM サーバーと IdM クライアントを設定します。

関連情報

- Ansible を使用して IdM タスクを自動化 し、クライアントの設定時間と複雑さを軽減し、ミスを減らします。

1.3. IDM のサーバーおよびクライアントの概要

Identity Management (IdM) ドメインには、以下のタイプのシステムが含まれます。

IdM クライアント

IdM クライアントは、サーバーに登録され、このサーバーで IdM サービスを使用するように設定された Red Hat Enterprise Linux システムです。
クライアントは、IdM サーバーと対話して、そのサーバーが提供するサービスにアクセスします。たとえば、クライアントは、Kerberos プロトコルを使用して認証を実行し、企業のシングルサインオン (SSO) のチケットを取得し、LDAP を使用して ID 情報およびポリシー情報を取得し、DNS を使用してサーバーとサービスの場所と、その接続方法を検出します。

IdM サーバー

IdM サーバーは、IdM ドメイン内の ID、認証、および認可の要求に応答する Red Hat Enterprise Linux システムです。ほとんどのデプロイメントでは、IdM サーバーとともに統合認証局 (CA) がインストールされています。
IdM サーバーは、ID 情報およびポリシー情報の中央リポジトリーです。IdM サーバーは、ドメインメンバーが使用する任意のサービスをホストすることもできます。

- 認証局 (CA)
- KRA (Key Recovery Authority)
- DNS
- Active Directory (AD) 信頼コントローラー
- Active Directory (AD) 信頼エージェント

IdM サーバーは、組み込み IdM クライアントでもあります。クライアントが自身に登録されるため、サーバーは、他のクライアントと同じ機能を提供します。

冗長性と可用性だけでなく、多数のクライアントにサービスを提供するため、IdM では 1 つのドメインに複数の IdM サーバーをデプロイできます。最大 60 台のサーバーをデプロイできます。これは、IdM ドメインで現在サポートされている、レプリカとも呼ばれる IdM サーバーの最大数です。IdM サーバーは、クライアントにさまざまなサービスを提供します。すべてのサーバーが、可能なサービスをすべて提供する必要があるわけではありません。Kerberos や LDAP などの一部のサーバーコンポーネントは、常にすべてのサーバーで利用できます。その他のサービス (CA、DNS、Trust Controller、Vault など) は必要に応じて使用します。つまり、デプロイメントでは、通常、さまざまなサーバーがさまざまなロールを果たしています。

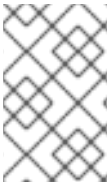
IdM トポロジーに統合 CA が含まれている場合は、1 台のサーバーで 証明書失効リスト (CRL) パブリッシャーサーバー のロール、1 台のサーバーで CA 更新サーバー のロールがあります。

デフォルトでは、最初にインストールした CA サーバーはこの 2 つのロールに対応しますが、これらのロールを別のサーバーに割り当てることができます。

警告

CA 更新サーバー は、CA サブシステムの 証明書および鍵 を追跡するドメインで唯一のシステムであるため、IdM デプロイメントにとっては極めて重要です。IdM デプロイメントに影響する障害からの復旧方法の詳細は、Identity Management を使用した障害復旧の実行 を参照してください。

管理者は、既存のサーバーの **レプリカ** を作成して追加のサーバーを作成し、冗長化および負荷分散を図ります。レプリカの作成時、IdM は既存サーバーの設定を複製します。レプリカは、ユーザー、システム、証明書、設定されたポリシーなど、そのコア設定を初期サーバーと共有します。



注記

CA 更新 と **CRL パブリッシャー** のロール以外は、レプリカと、レプリカを作成したサーバーは機能的に同じです。そのため、ここでは **サーバー** と **レプリカ** という用語を、文脈に応じて同じ意味で使用します。

1.4. IDM クライアントのインストールをサポートする RHEL のバージョン

IdM サーバーが Red Hat Enterprise Linux 9 の最新マイナーバージョンで実行されている Identity Management デプロイメントでは、以下の最新マイナーバージョンで実行されているクライアントがサポートされます。

- RHEL 7
- RHEL 8
- RHEL 9

注記

他のクライアントシステム (Ubuntu など) は IdM 9 サーバーと連携できますが、Red Hat では、これらのクライアントのサポートを提供していません。

1.5. RHEL における IDM およびアクセス制御: 中央対ローカル

Red Hat Enterprise Linux では、システムのドメイン全体に集中型のツールを使用するか、1 台のシステムにローカルのツールを使用して、ID およびアクセス制御ポリシーを管理できます。

複数の Red Hat Enterprise Linux サーバーでのアイデンティティーとポリシーの管理

IT 管理者は、IdM で以下が可能になります。

- ID とグループ化メカニズムを一か所 (IdM サーバー) で管理
- パスワード、PKI 証明書、OTP トークン、SSH 鍵などのさまざまな種類の認証情報を一元管理
- 複数のマシンで同時にポリシーを均一に適用
- 外部の Active Directory ユーザー用に、POSIX およびその他の属性を管理
- ホストベースのアクセス制御、委譲などのルールを使用してユーザーに異なるアクセスレベルを設定
- 特権昇格規則 (sudo) と必須アクセス制御 (SELinux ユーザーマッピング) の一元管理
- 中央の PKI インフラストラクチャーおよび秘密ストアの維持
- ホームディレクトリーのマウント方法の定義

IdM を使用しない場合:

- 各サーバーが個別に管理されます。
- パスワードがすべてローカルマシンに保存されます。
- IT 管理者が、すべてのマシンのユーザーを管理し、認証と認可のポリシーを個別に設定し、ローカルパスワードを管理します。

1.6. IDM の用語

Active Directory フォレスト

Active Directory (AD) フォレストは、共通のグローバルカタログ、ディレクトリースキーマ、論理構造、およびディレクトリー設定を共有する 1 つ以上のドメインツリーのセットです。フォレストは、ユーザー、コンピューター、グループ、およびその他のオブジェクトにアクセスできるセキュリティー境界を表します。詳細は、Forests の Microsoft ドキュメントを参照してください。

Active Directory グローバルカタログ

グローバルカタログは Active Directory (AD) の機能であり、オブジェクトがドメインコントローラーのドメインのメンバーかどうかに関わらず、ドメインコントローラーがフォレスト内のオブジェクトに関する情報を提供できるようにします。グローバルカタログ機能が有効になっているドメインコントローラーは、グローバルカタログサーバーと呼ばれます。グローバルカタログは、マルチドメイン Active Directory ドメインサービス (AD DS) にあるすべてのドメインのすべてのオブジェクトの検索可能なカタログを提供します。

Active Directory セキュリティー識別子

セキュリティー識別子 (SID) は、ユーザー、グループ、ホストなど、Active Directory のオブジェクトに割り当てられた一意の ID 番号です。これは、Linux の UID および GID と同等の機能です。

Ansible プレイ

Ansible のプレイは、Ansible Playbook のビルディングブロックです。プレイの目的は、ホストのグループを、Ansible タスクで表す明確に定義されたロールにマッピングすることです。

Ansible Playbook

Ansible Playbook は、1 つ以上の Ansible プレイを含むファイルです。詳細は、Playbook に関する公式の Ansible ドキュメント を参照してください。

Ansible タスク

Ansible タスクは、Ansible のアクションの単位です。Ansible play には、複数のタスクを含めることができます。各タスクの目的は、非常に特殊な引数を使用してモジュールを実行することです。Ansible タスクは、特定の Ansible ロールまたはモジュールにより定義された状態を実現する一連の手順です。また、そのロールまたはモジュールの変数により微調整されます。詳細は、公式の Ansible タスクのドキュメント を参照してください。

Apache Web Server

Apache HTTP Server (通称 Apache) は、Apache License 2.0 の条件に基づいてリリースされた、無料かつオープンソースのクロスプラットフォーム Web サーバーアプリケーションです。Apache は、World Wide Web の初期の成長において重要なロールを果たし、現在は、主要な HTTP サーバーとなっています。そのプロセス名は **httpd** で、**HTTP デーモン** の略になります。Red Hat Identity Management (IdM) は、Apache Web Server を使用して IdM Web UI を表示し、Directory Server や認証局などのコンポーネント間の通信を調整します。

証明書

証明書とは、個人、サーバー、会社、または他のエンティティーを特定し、その ID を公開鍵に関連付けるために使用される電子ドキュメントです。ドライバーのライセンスやパスポートなど、証明書は、ユーザー ID の一般的に認識される証明を提供します。公開鍵暗号では、証明書を使用してなりすましの問題に対処します。

IdM の認証局 (CA)

デジタル証明書を発行するエンティティーです。Red Hat Identity Management では、プライマリー CA は IdM CA **ipa** です。**ipa** CA 証明書は、次のいずれかの種類になります。

- 自己署名。この場合、**ipa** CA はルート CA です。
- 外部署名。この場合、**ipa** CA は外部 CA に従属します。

IdM では、複数の **サブ CA** も作成できます。サブ CA は、証明書が以下のいずれかの種類である IdM CA です。

- **ipa** CA により署名されます。
- それ自体と **ipa** CA との間にある中間 CA で署名されます。サブ CA の証明書は自己署名できません。

CA サービスの計画 も参照してください。

フォレスト間の信頼

信頼は、2 つの Kerberos レルム間のアクセス関係を確立し、あるドメインのユーザーとサービスが別のドメインのリソースにアクセスできるようにします。
Active Directory (AD) フォレストルートドメインと IdM ドメインとの間のフォレスト間の信頼関係により、AD フォレストドメインのユーザーは、IdM ドメインの Linux マシンおよびサービスと相互作用できます。AD の観点から観ると、Identity Management は、1 つの AD ドメインを持つ個別の AD フォレストを表します。詳細は、信頼の仕組み を参照してください。

Directory Server

Directory Server は、ユーザー ID とアプリケーション情報を一元管理します。アプリケーション設定、ユーザープロファイル、グループデータ、ポリシー、アクセス制御情報を保存するためのオペレーティングシステムに依存しない、ネットワークベースのレジストリーを提供します。ネットワーク上の各リソースは、Directory Server によりオブジェクトと見なされます。特定リソースに関する情報は、そのリソースまたはオブジェクトに関連付けられた属性の集合として保存されます。Red Hat Directory Server は、LDAP 規格に準拠しています。

DNS PTR レコード

DNS ポインター (PTR) レコードは、ホストの IP アドレスをドメインまたはホスト名に解決します。PTR レコードは DNS A と AAAA レコードの逆で、ホスト名を IP アドレスに解決します。DNS PTR レコードは、逆引き DNS ルックアップを有効にします。PTR レコードは DNS サーバーに保存されます。

DNS SRV レコード

DNS サービス (SRV) レコードは、ドメインで利用可能なサービスのホスト名、ポート番号、トランスポートプロトコル、優先度、および重みを定義します。SRV レコードを使用して、IdM サーバーおよびレプリカを特定できます。

ドメインコントローラー (DC)

ドメインコントローラー (DC) は、ドメイン内のセキュリティー認証要求に応答し、そのドメイン内のリソースへのアクセスを制御するホストです。IdM サーバーは、IdM ドメインの DC として機能します。DC はユーザーを認証し、ユーザーアカウント情報を保存し、ドメインのセキュリティーポリシーを強制します。ユーザーがドメインにログインすると、DC はユーザーの認証情報を認証および検証し、アクセスを許可または拒否します。

完全修飾ドメイン名

完全修飾ドメイン名 (FQDN) は、DNS (Domain Name System) の階層内のホストの正確な場所を指定するドメイン名です。親ドメイン **example.** com にホスト名 **myhost** を持つデバイスには FQDN **myhost.example.com** があります。FQDN は、他のドメインの **myhost** と呼ばれる他のホストとデバイスを一意に区別します。

DNS 自動検出を使用してホスト **machine1** に IdM クライアントをインストールし、DNS レコードが正しく設定されている場合は、**machine1** の FQDN のみが必要になります。詳細は IdM のホスト名および DNS 要件 を参照してください。

GSSAPI

Generic Security Service Application Program Interface (GSSAPI または GSS-API) を使用すると、開発者はアプリケーションがピアアプリケーションに送信されるデータを保護する方法を抽象化できます。セキュリティーサービスベンダーは、セキュリティーソフトウェアを使用して、一般的なプロシージャ呼び出しの GSSAPI 実装をライブラリーとして提供できます。これらのライブラリーは、アプリケーションを作成し、ベンダーに依存しない GSSAPI のみを使用できるアプリケーション作成者向けに、GSSAPI 互換のインターフェイスを提供します。この柔軟性により、開発者は、セキュリティー実装を、特定のプラットフォーム、セキュリティーメカニズム、タイプの保護、またはトランスポートプロトコルに合わせて調整する必要がなくなります。

Kerberos は主要な GSSAPI メカニズムの実装であり、Red Hat Enterprise Linux および Microsoft Windows Active Directory Kerberos の実装を API 互換にすることができます。

非表示のレプリカ

非表示レプリカは、稼働中および利用可能なすべてのサービスを持つ IdM レプリカですが、サーバーロールは無効であり、クライアントは DNS に SRV レコードがないため、レプリカを検出できません。

非表示のレプリカは、主に IdM サービスのシャットダウンが必要なバックアップ、一括インポートおよびエクスポート、アクションなどのサービス用に設計されています。非表示のレプリカを使用するクライアントはないため、管理者はクライアントに影響を与えることなく、このホスト上のサービスを一時的にシャットダウンできます。詳細は 非表示のレプリカモード を参照してください。

HTTP サーバー

Web サーバー を参照してください。

ID マッピング

SSSD は、AD ユーザーの SID を使用して、**ID マッピング** と呼ばれるプロセスにおいてアルゴリズムで POSIX ID を生成できます。ID マッピングは、AD の SID と Linux の ID との間にマップを作成します。

- SSSD が新しい AD ドメインを検出すると、利用可能な ID の範囲を新しいドメインに割り当てます。したがって、各 AD ドメインは、すべての SSSD クライアントマシンで同じ ID 範囲を持ちます。
- AD ユーザーが SSSD クライアントマシンに初めてログインすると、SSSD は、ユーザーの SID およびそのドメインの ID 範囲を基にした UID など、SSSD キャッシュにユーザーのエントリーを作成します。
- AD ユーザーの ID は、同じ SID から一貫した方法で生成されるため、Red Hat Enterprise Linux システムにログインする場合は、そのユーザーに同じ UID と GID が使用されます。

ID 範囲

ID 範囲は、IdM トポロジーまたは特定のレプリカに割り当てられた ID 番号の範囲です。ID 範囲を使用して、新規ユーザー、ホスト、およびグループの UID および GID の有効な範囲を指定できます。ID 範囲は、ID 番号の競合を避けるために使用されます。IdM の ID 範囲には、以下の 2 つのタイプがあります。

- **IdM ID 範囲**
  この ID 範囲を使用して、IdM トポロジー全体でユーザーおよびグループの UID および GID を定義します。最初の IdM サーバーをインストールすると、IdM ID 範囲が作成されます。IdM ID の範囲は、作成後に変更することはできません。ただし、(元の ID 範囲が枯渇に近づいた場合などに) 追加の IdM ID 範囲を作成できます。
- **分散型数値割り当て (DNA) の ID 範囲**
  この ID 範囲を使用して、レプリカが新規ユーザーの作成時に使用する UID および GID を定義します。IdM レプリカに新しいユーザーまたはホストエントリーを追加すると、そのレプリカに DNA ID 範囲が割り当てられます。管理者は DNA ID 範囲を変更できますが、新しい定義は既存の IdM ID 範囲内に収まるようにする必要があります。

  IdM の範囲と DNA 範囲は一致しますが、相互接続されていないことに注意してください。1 つの範囲を変更する場合は、別の範囲を一致させるように変更してください。

詳細は、ID 範囲 を参照してください。

ID ビュー

ID ビューを使用すると、POSIX ユーザーまたはグループ属性に新しい値を指定でき、新しい値が適用されるクライアントホストを 1 つまたは複数定義できます。たとえば、ID ビューを使用して以下を行うことができます。

- 環境ごとに異なる属性値を定義します。
- 以前生成された属性の値を別の値に置き換えます。

IdM-AD 信頼設定では、**Default Trust View** は、AD ユーザーおよびグループに適用される ID ビューです。**Default Trust View** を使用すると、AD ユーザーおよびグループのカスタム POSIX 属性を定義できます。これにより、AD で定義された値をオーバーライドできます。

詳細は ID ビューを使用した IdM クライアントのユーザー属性値を上書きする を参照してください。

IdM CA サーバー

IdM 認証局サービス (CA) がインストールされ、実行している IdM サーバー。
別名 - **CA サーバー**

IdM デプロイメント

IdM インストール全体を対象とする用語。以下の質問に回答することで、IdM デプロイメントについて説明できます。

- IdM デプロイメントは、テスト用デプロイメントまたは実稼働デプロイメントですか?
  - IdM サーバーは何台ありますか?
- IdM デプロイメントに 統合 CA は含まれていますか?
  - 含まれている場合、統合 CA は自己署名、または外部署名ですか?
  - 含まれている場合、どのサーバーで CA ロール を利用できますか?KRA ロールは、どのサーバーで利用できますか?
- IdM デプロイメントに 統合 DNS は含まれていますか?
  - 含まれている場合、どのサーバーが DNS ロールを利用できますか?
- IdM デプロイメントは AD フォレスト と信頼関係にありますか?
  - その場合、どのサーバーで AD 信頼コントローラーまたは AD 信頼エージェント ロールを使用できますか?

IdM サーバーおよびレプリカ

IdM デプロイメントの最初のサーバーをインストールするには、**ipa-server-install** コマンドを使用する必要があります。
管理者は、**ipa-replica-install** コマンドを使用して、最初にインストールしたサーバーに加えて **レプリカ** をインストールできます。デフォルトでは、レプリカをインストールすると、それが作成された IdM サーバーとの レプリカ合意 が作成され、残りの IdM への更新の送受信が実現します。

最初にインストールしたサーバーとレプリカの間に機能的な違いはありません。どちらも完全に機能する読み取り/書き込み IdM サーバー です。

非推奨名: **マスターサーバー**

IdM CA 更新サーバー

IdM トポロジーに統合認証局 (CA) が含まれている場合は、1 台のサーバーに CA 更新サーバー 固有のロールがあります。このサーバーは、IdM システム証明書を管理して更新します。
デフォルトでは、最初にインストールした CA サーバーがこのロールに対応しますが、どの CA サーバーでも CA 更新サーバーに設定できます。統合 CA のないデプロイメントには、CA 更新サーバーはありません。

非推奨名: **マスター CA**

IdM CRL パブリッシャーサーバー

IdM トポロジーに統合認証局 (CA) が含まれている場合は、1 台のサーバーには、証明書失効リスト (CRL) パブリッシャーサーバー 固有のロールがあります。このサーバーは CRL を管理します。デフォルトでは、**CA 更新サーバー** のロールに対応するサーバーは、このロールにも対応しますが、CA サーバーを CRL パブリッシャーサーバーとして設定することもできます。統合 CA のないデプロイメントには CRL パブリッシャーサーバーはありません。

IdM トポロジー

IdM ソリューションの構造、特に個々のデータセンターとクラスターとの間、およびその内部でレプリカ合意がどのように設定されるかを指す用語。

Kerberos 認証インジケーター

認証インジケーターは Kerberos チケットに割り当てられ、チケットの取得に使用される初期認証方法を表します。

- 2 要素認証 (パスワード + ワンタイムパスワード) の **otp**
- **radius** - Remote Authentication Dial-In User Service (RADIUS) 認証 (通常 802.1x 認証の場合)
- Kerberos (PKINIT)、スマートカード、または証明書認証用の公開鍵暗号化の **pkinit**
- **強化** - ブルートフォース攻撃に対して強化されたパスワードスワードのために

詳細は、Kerberos 認証インジケーター を参照してください。

Kerberos キータブ

パスワードはユーザーのデフォルトの認証方法ですが、キータブはホストおよびサービスのデフォルト認証方法です。Kerberos キータブは、Kerberos プリンシパルとその関連暗号鍵のリストが含まれるファイルで、サービスは独自の Kerberos キーを取得し、ユーザーのアイデンティティーを検証できます。
たとえば、すべての IdM クライアントには、Kerberos レルムのクライアントマシンを表す **host** プリンシパルに関する情報を格納する **/etc/krb5.keytab** ファイルがあります。

Kerberos プリンシパル

一意の Kerberos プリンシパルは、Kerberos レルムの各ユーザー、サービス、およびホストを特定します。

エンティティー	命名規則	例
ユーザー	**identifier@REALM**	**admin@EXAMPLE.COM**
サービス	**service/fully-qualified-hostname@REALM**	**http/server.example.com@EXAMPLE.COM**
ホスト	**host/fully-qualified-hostname@REALM**	**host/client.example.com@EXAMPLE.COM**

Kerberos プロトコル

Kerberos は、秘密鍵の暗号化を使用してクライアントおよびサーバーアプリケーションに強力な認証を提供するネットワーク認証プロトコルです。IdM および Active Directory は、ユーザー、ホスト、およびサービスの認証に Kerberos を使用します。

Kerberos レルム

Kerberos レルムには、Kerberos Key Distribution Center (KDC) が管理するすべてのプリンシパルが含まれます。IdM デプロイメントでは、Kerberos レルムには、IdM ユーザー、ホスト、およびサービスがすべて含まれます。

Kerberos チケットポリシー

Kerberos Key Distribution Center (KDC) は、接続ポリシーによりチケットアクセス制御を強制し、チケットライフサイクルポリシーで Kerberos チケットの期間が管理されます。たとえば、デフォルトのグローバルチケットの有効期間は 1 日で、デフォルトのグローバル最大更新期間は 1 週間です。
詳細は、IdM Kerberos チケットポリシータイプ を参照してください。

キー配布センター (KDC)

Kerberos Key Distribution Center (KDC) は、Kerberos 認証情報情報を管理する中央で信頼できる認証局として機能するサービスです。KDC は Kerberos チケットを発行し、IdM ネットワーク内のエンティティーから送信されるデータの信頼性を確保します。
詳細は、IdM KDC のロール を参照してください。

LDAP

LDAP (Lightweight Directory Access Protocol) は、ネットワーク経由で分散ディレクトリー情報サービスにアクセスし、維持するためのオープンで、ベンダーに依存しないアプリケーションプロトコルです。この仕様の一部は、ディレクトリー情報ツリー (DIT) です。これは、ディレクトリーサービスエントリーの DN (識別名) で設定される階層ツリー形式のデータを表します。LDAP は、ネットワーク内のディレクトリーサービスに関する ISO X.500 標準で規定されている DAP (Directory Access Protocol) の "lightweight" バージョンです。

軽量サブ CA

IdM では、軽量サブ CA は認証局 (CA) で、証明書が IdM ルート CA またはその下位の CA のいずれかによって署名されます。軽量のサブ CA は、VPN 接続または HTTP 接続のセキュリティーを保護するなど、特定目的でのみ証明書を発行します。
詳細は、証明書のサブセットだけを信頼するアプリケーションの制限 を参照してください。

パスワードポリシー

パスワードポリシーは、特定の IdM ユーザーグループのパスワードが満たさなければならない条件です。条件には、以下のパラメーターを含めることができます。

- パスワードの長さ
- 使用される文字クラスの数
- パスワードの最大有効期間。

詳細は パスワードポリシーとは を参照してください。

POSIX 属性

POSIX 属性は、オペレーティングシステム間の互換性を維持するためのユーザー属性です。
Red Hat Identity Management 環境では、ユーザーの POSIX 属性には以下が含まれます。

- **cn** (ユーザーの名前)
- **UID** (アカウント名 (ログイン))
- **uidNumber** (ユーザー番号 (UID))
- **gidNumber** (プライマリーグループ番号 (GID))

- **homeDirectory** (ユーザーのホームディレクトリー)

Red Hat Identity Management 環境では、グループの POSIX 属性には以下が含まれます。

- **cn** (グループ名)
- **gidNumber** (グループ番号 (GID))

これらの属性は、ユーザーおよびグループを個別のエンティティーとして識別します。

レプリカ合意

レプリカ合意は、同じ IdM デプロイメントの 2 つの IdM サーバー間の合意です。レプリカ合意は、データと設定が 2 台のサーバー間で継続的に複製されることを保証します。
IdM は、2 種類のレプリカ合意を使用します。ID 情報を複製する **ドメインレプリカ** の合意と、証明書情報を複製する **証明書のレプリカ** の合意です。

詳細は、以下を参照してください。

- レプリカ合意
- 適切なレプリカ数の決定
- トポロジー内でレプリカの接続
- レプリカトポロジーの例

スマートカード

スマートカードは、リソースへのアクセスを制御するために使用されるリムーバブルデバイスまたはカードです。集積回路 (IC) チップを搭載したプラスチック製のクレジットカードサイズのカード、Yubikey などの小型 USB デバイス、またはその他の同様のデバイスになります。スマートカードは、ユーザーがスマートカードをホストコンピューターに接続でき、そのホストコンピューターのソフトウェアは、スマートカードに保存されている鍵マテリアルと相互作用してユーザーを認証できます。

SSSD

SSSD (System Security Services Daemon) は、RHEL ホストでユーザー認証およびユーザー認可を管理するシステムサービスです。SSSD は、必要に応じて、オフライン認証時に、リモートプロバイダーから取得したユーザー ID および認証情報のキャッシュを保持します。詳細は SSSD とその利点について を参照してください。

SSSD バックエンド

SSSD バックエンド (通常はデータプロバイダーとも呼ばれます) は、SSSD キャッシュを管理し、作成する SSSD 子プロセスです。このプロセスは LDAP サーバーと通信し、異なるルックアップクエリーを実行し、結果をキャッシュに保存します。また、LDAP または Kerberos に対してオンライン認証を実行し、ログインするユーザーにアクセスポリシーおよびパスワードポリシーを適用します。

TGT (Ticket-granting ticket)

Kerberos Key Distribution Center (KDC) に認証した後、ユーザーはチケット保証チケット (TGT) を受け取ります。このチケットは、Web サイトや電子メールなどの他のサービスにアクセスチケットを要求するのに使用できる認証情報の一時的なセットです。
TGT を使用してさらにアクセスを要求すると、ユーザーは複数のサービスにアクセスするために一度だけ認証する必要があるため、ユーザーはシングルサインオンのエクスペリエンスが得られます。TGT は更新可能で、Kerberos チケットポリシーはチケット更新の制限とアクセス制御を決定します。

詳細は Kerberos チケットポリシーの管理 を参照してください。

Web server

Web サーバーは、コンピューターのソフトウェアで、ページ、イメージ、アプリケーションなどの Web コンテンツの要求を受け入れる基本となるハードウェアです。Web ブラウザーなどのユーザーエージェントは、HTTP を使用して特定のリソース、Web コンテンツの配布に使用されるネットワークプロトコル、またはそのセキュアバリアントの HTTPS を要求します。Web サーバーは、そのリソースの内容またはエラーメッセージで応答します。Web サーバーは、ユーザーエージェントから送信されたリソースを受け入れ、保存することもできます。Red Hat Identity Management (IdM) は、Apache Web Server を使用して IdM Web UI を表示し、Directory Server や認証局 (CA) などのコンポーネント間の通信を調整します。Apache Web Server を参照してください。

その他の用語集

この用語に Identity Management 用語が見つからない場合は、Directory Server and Certificate System の用語を参照してください。

- Directory Server 11 の用語
- Certificate System 9 の用語

第2章 IDM でのフェイルオーバー、負荷分散、高可用性

Identity Management (IdM) には、IdM クライアント向けのフェイルオーバーメカニズムと、IdM サーバー向けの負荷分散および高可用性機能があります。

2.1. クライアント側のフェイルオーバー機能

- デフォルトでは、IdM クライアントの **SSSD** サービスは、DNS からのサービス (SRV) リソースレコードを使用して、接続先に最も適した IdM サーバーを自動的に決定するように設定されています。この動作は、**/etc/sssd/sssd.conf** ファイルの **ipa_server** パラメーターの **_srv_** オプションで制御します。

```
[root@client ~]# cat /etc/sssd/sssd.conf

[domain/example.com]
id_provider = ipa
ipa_server = _srv_, server.example.com
...
```

  IdM サーバーがオフラインになると、IdM クライアントの SSSD サービスが、自動的に検出した別の IdM サーバーに接続します。

- パフォーマンス上の理由から DNS ルックアップをバイパスする場合は、 **ipa_server** パラメーターから **_srv_** エントリーを削除し、クライアントが接続すべき IdM サーバーを優先順に指定します。

```
[root@client ~]# cat /etc/sssd/sssd.conf

[domain/example.com]
id_provider = ipa
ipa_server = server1.example.com, server2.example.com
...
```

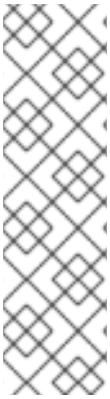
2.2. サーバー側の負荷分散およびサービスの可用性

複数の IdM レプリカをインストールして、IdM で負荷分散および高可用性を実行できます。

- 地理的に分散したネットワークがある場合には、データセンターごとに複数の IdM レプリカを設定することで、IdM クライアントと、最寄りのアクセス可能なサーバーとの間のパスを短くできます。

- Red Hat は、最大 60 台のレプリカを使用する環境をサポートします。

- IdM レプリケーションメカニズムでは、アクティブ/アクティブのサービスの可用性 (全 IdM レプリカのサービスを同時利用可) を提供します。



注記

Red Hat は、IdM およびその他の負荷分散または高可用性 (HA) ソフトウェアを組み合わせることを推奨します。

サードパーティーの高可用性ソリューションの多くは、アクティブ/パッシブのシナリオを想定しており、IdM へのサービスが不要に中断されてしまう可能性があります。他のソリューションでは、クラスター化されたサービスごとに仮想 IP または単一のホスト名を使用します。このような方法はすべて、通常、IdM ソリューションが提供するタイプのサービスの可用性では適切に機能しません。また、Kerberos との統合性が非常に低く、デプロイメントのセキュリティーと安定性が全体的に低下します。

第3章 レプリカトポロジーの計画

ユースケースに適したレプリカトポロジーを決定するためのガイダンスをご確認ください。

3.1. 高パフォーマンスおよび障害復旧のソリューションとなる複数のレプリカサーバー

既存の IdM サーバーのレプリカを作成することで、Identity Management (IdM) サービスの継続的な機能と高可用性を実現できます。

適切な数の IdM レプリカを作成すると、負荷分散を使用してクライアントの要求を複数のサーバーに分散し、IdM サービスのパフォーマンスを最適化できます。IdM を使用すると、企業の組織構造を反映するように、地理的に分散したデータセンターに追加のサーバーを配置できます。これにより、IdM クライアントと、アクセスできる一番近いサーバーとの間の経路が短くなります。さらに、複数のサーバーを使用することで、負荷を分散し、より多くのクライアントに拡張できます。

IdM サーバーのレプリカ作成は、サーバーの損失を軽減または防止するための一般的なバックアップメカニズムでもあります。たとえば、1 台のサーバーに障害が発生しても、残りのサーバーがドメインへのサービスの提供を継続します。障害が発生していないサーバーの 1 台から新しいレプリカを作成し、失われたサーバーを回復することもできます。

3.2. IDM のサーバーおよびクライアントの概要

Identity Management (IdM) ドメインには、以下のタイプのシステムが含まれます。

IdM クライアント

IdM クライアントは、サーバーに登録され、このサーバーで IdM サービスを使用するように設定された Red Hat Enterprise Linux システムです。
クライアントは、IdM サーバーと対話して、そのサーバーが提供するサービスにアクセスします。たとえば、クライアントは、Kerberos プロトコルを使用して認証を実行し、企業のシングルサインオン (SSO) のチケットを取得し、LDAP を使用して ID 情報およびポリシー情報を取得し、DNS を使用してサーバーとサービスの場所と、その接続方法を検出します。

IdM サーバー

IdM サーバーは、IdM ドメイン内の ID、認証、および認可の要求に応答する Red Hat Enterprise Linux システムです。ほとんどのデプロイメントでは、IdM サーバーとともに統合認証局 (CA) がインストールされています。
IdM サーバーは、ID 情報およびポリシー情報の中央リポジトリーです。IdM サーバーは、ドメインメンバーが使用する任意のサービスをホストすることもできます。

- 認証局 (CA)
- KRA (Key Recovery Authority)
- DNS
- Active Directory (AD) 信頼コントローラー
- Active Directory (AD) 信頼エージェント

IdM サーバーは、組み込み IdM クライアントでもあります。クライアントが自身に登録されるため、サーバーは、他のクライアントと同じ機能を提供します。

冗長性と可用性だけでなく、多数のクライアントにサービスを提供するため、IdM では 1 つのドメインに複数の IdM サーバーをデプロイできます。最大 60 台のサーバーをデプロイできます。これは、IdM ドメインで現在サポートされている、レプリカとも呼ばれる IdM サーバーの最大数です。IdM サーバーは、クライアントにさまざまなサービスを提供します。すべてのサーバーが、可能なサービスをすべて提供する必要があるわけではありません。Kerberos や LDAP などの一部のサーバーコンポーネントは、常にすべてのサーバーで利用できます。その他のサービス (CA、DNS、Trust Controller、Vault など) は必要に応じて使用します。つまり、デプロイメントでは、通常、さまざまなサーバーがさまざまなロールを果たしています。

IdM トポロジーに統合 CA が含まれている場合は、1 台のサーバーで 証明書失効リスト (CRL) パブリッシャーサーバー のロール、1 台のサーバーで CA 更新サーバー のロールがあります。

デフォルトでは、最初にインストールした CA サーバーはこの 2 つのロールに対応しますが、これらのロールを別のサーバーに割り当てることができます。

警告

CA 更新サーバー は、CA サブシステムの 証明書および鍵 を追跡するドメインで唯一のシステムであるため、IdM デプロイメントにとっては極めて重要です。IdM デプロイメントに影響する障害からの復旧方法の詳細は、Identity Management を使用した障害復旧の実行 を参照してください。

管理者は、既存のサーバーの **レプリカ** を作成して追加のサーバーを作成し、冗長化および負荷分散を図ります。レプリカの作成時、IdM は既存サーバーの設定を複製します。レプリカは、ユーザー、システム、証明書、設定されたポリシーなど、そのコア設定を初期サーバーと共有します。

注記

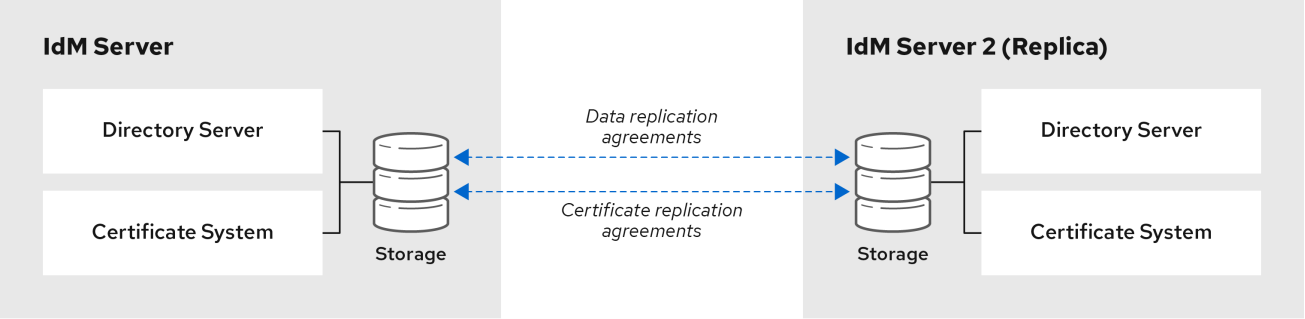
CA 更新 と **CRL パブリッシャー** のロール以外は、レプリカと、レプリカを作成したサーバーは機能的に同じです。そのため、ここでは **サーバー** と **レプリカ** という用語を、文脈に応じて同じ意味で使用します。

3.3. IDM レプリカ間のレプリカ合意

管理者が、既存のサーバーに基づいてレプリカを作成すると、Identity Management (IdM) は、初期サーバーとレプリカとの間に **レプリカ合意** を作成します。レプリカ合意は、データと設定が 2 台のサーバー間で継続的に複製されることを保証します。

IdM は、**複数の読み取り/書き込みレプリカ複製** を使用します。この設定では、レプリカ合意に参加しているすべてのレプリカが更新の受信と提供を行うので、サプライヤーとコンシューマーとみなされます。レプリカ合意は常に双方向です。

図3.1 サーバーとレプリカ合意



64_RHEL_0120

IdM は、2 種類のレプリカ合意を使用します。

ドメインのレプリカ合意

この合意は、識別情報を複製します。

証明書のレプリカ合意

この合意は、証明書情報を複製します。

両方の複製チャンネルは独立しています。2 台のサーバー間で、いずれかまたは両方の種類のレプリカ合意を設定できます。たとえば、サーバー A とサーバー B にドメインレプリカ合意のみが設定されている場合は、証明書情報ではなく ID 情報だけが複製されます。

3.4. トポロジー内の IDM レプリカの適切な数を決定するためのガイドライン

組織の要件に合わせて IdM トポロジーを計画し、最適なパフォーマンスとサービスの可用性を確保してください。

各データセンターに少なくとも 2 つのレプリカをセットアップする

各データセンターに少なくとも 2 つのレプリカをデプロイして、1 台のサーバーに障害が発生した場合にレプリカが引き継いで要求を処理できるようにします。

クライアントにサービスを提供するために十分な数のサーバーをセットアップする

1 台の Identity Management (IdM) サーバーで 2000 - 3000 台のクライアントにサービスを提供できます。ここでは、クライアントがサーバーに対して 1 日に複数回クエリーする (毎分ではありません) ことを想定しています。より頻繁なクエリーが予想される場合は、より多くのサーバーを計画してください。

十分な数の認証局 (CA) レプリカを設定します。

CA ロールがインストールされているレプリカのみが、証明書データを複製できます。IdM CA を使用する場合は、環境に、証明書のレプリカ合意がある CA レプリカが 2 つ以上あることを確認します。

1 つの IdM ドメインに最大 60 台のレプリカを設定

Red Hat は、最大 60 のレプリカを持つ環境に対応します。

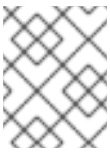
3.5. トポロジーで IDM レプリカを接続するためのガイドライン

1 台のレプリカを少なくとも 2 つのレプリカに接続

追加のレプリカ合意を設定すると、初期レプリカと最初にインストールしたサーバーとの間だけでなく、他のレプリカ間でも情報が複製されます。

レプリカを、その他のレプリカ (最大 4 つ) に接続 (必須要件ではありません)

サーバーごとに多数のレプリカ合意を設定しても、大きな利点はありません。受信側のレプリカは、一度に1つの他のレプリカによってのみ更新できます。その間、その他のレプリカ合意はアイドル状態になります。通常、レプリカごとに 4 つ以上のレプリカ合意があると、リソースが無駄になります。



注記

この推奨事項は、証明書のレプリカ合意とドメインのレプリカ合意の両方に適用されます。

レプリカごとに 4 つのレプリカ合意という制限は、次の 2 つの場合には、例外として適用されません。

- 特定のレプリカがオンラインでない場合や応答していない場合にフェイルオーバーパスが必要な場合
- 大規模デプロイメントで、特定のノード間に追加の直接リンクが必要な場合

レプリカ合意を多数設定すると、全体のパフォーマンスに悪影響が及ぶ可能性があります。トポロジー内の複数のレプリカ合意が更新を送信すると、特定のレプリカの changelog データベースファイル上で、受信する更新と送信する更新の間の競合が増大することがあります。

レプリカごとにさらに多くのレプリカ合意を使用する場合は、レプリケーションの問題やレイテンシーが発生しないようにしてください。距離が長く、中間ノードの数が多いと、レイテンシーの問題が発生する場合があることに注意してください。

データセンター内のレプリカを互いに接続

これにより、データセンター内のドメインレプリケーションが確実になります。

各データセンターを少なくとも 2 つの他のデータセンターに接続

これにより、データセンター間のドメインレプリケーションが確実になります。

少なくとも一対のレプリカ合意を使用してデータセンターを接続

データセンター A および B に、A1 への B1 までのレプリカ合意がある場合は、A2 から B2 へのレプリカ合意があれば、いずれかのサーバーがダウンしても、2 つのデータセンター間でレプリケーションを続行できます。

3.6. レプリカトポロジーの例

次のいずれかの例を使用して、信頼性の高いレプリカトポロジーを作成できます。

図3.2 4 つのデータセンターで構成されるレプリカトポロジー。各データセンターに、レプリカ合意で接続された 4 台のサーバーがある

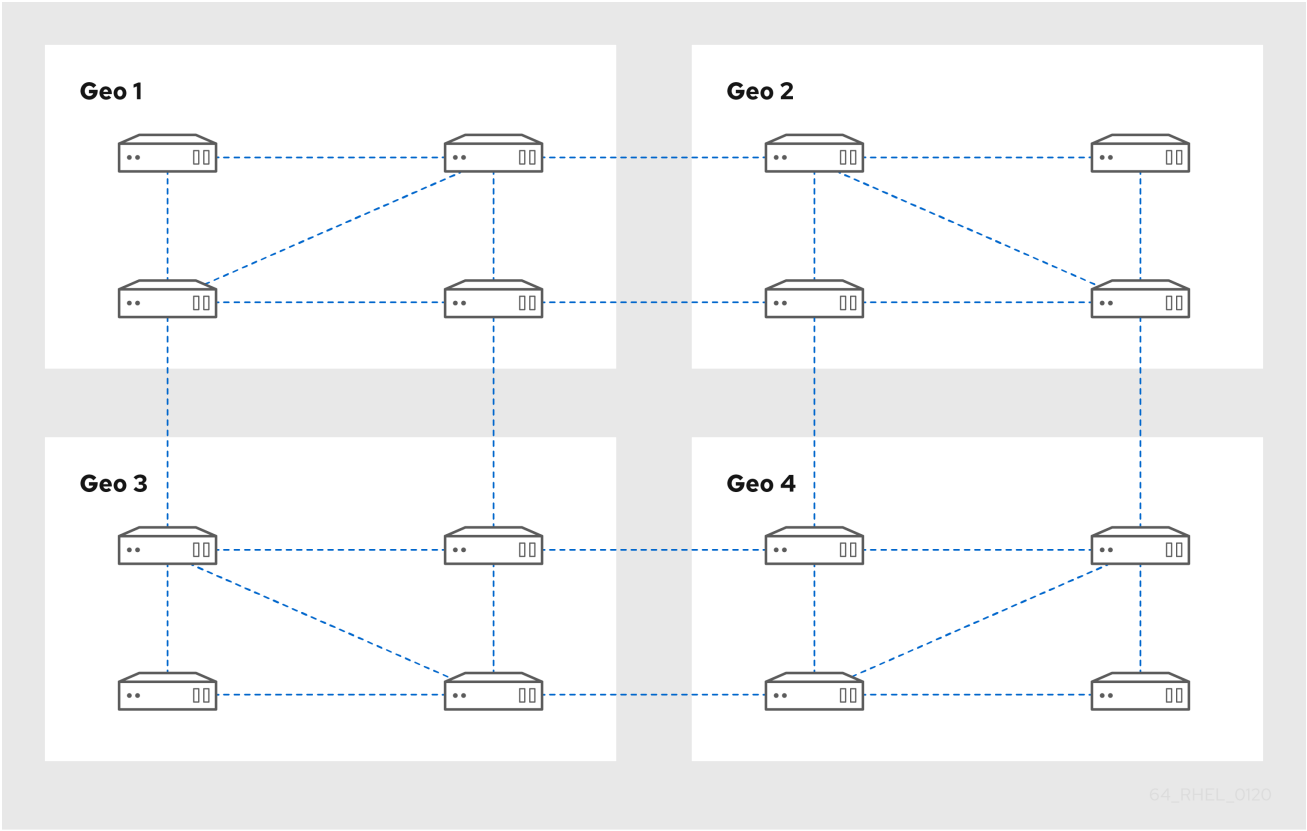
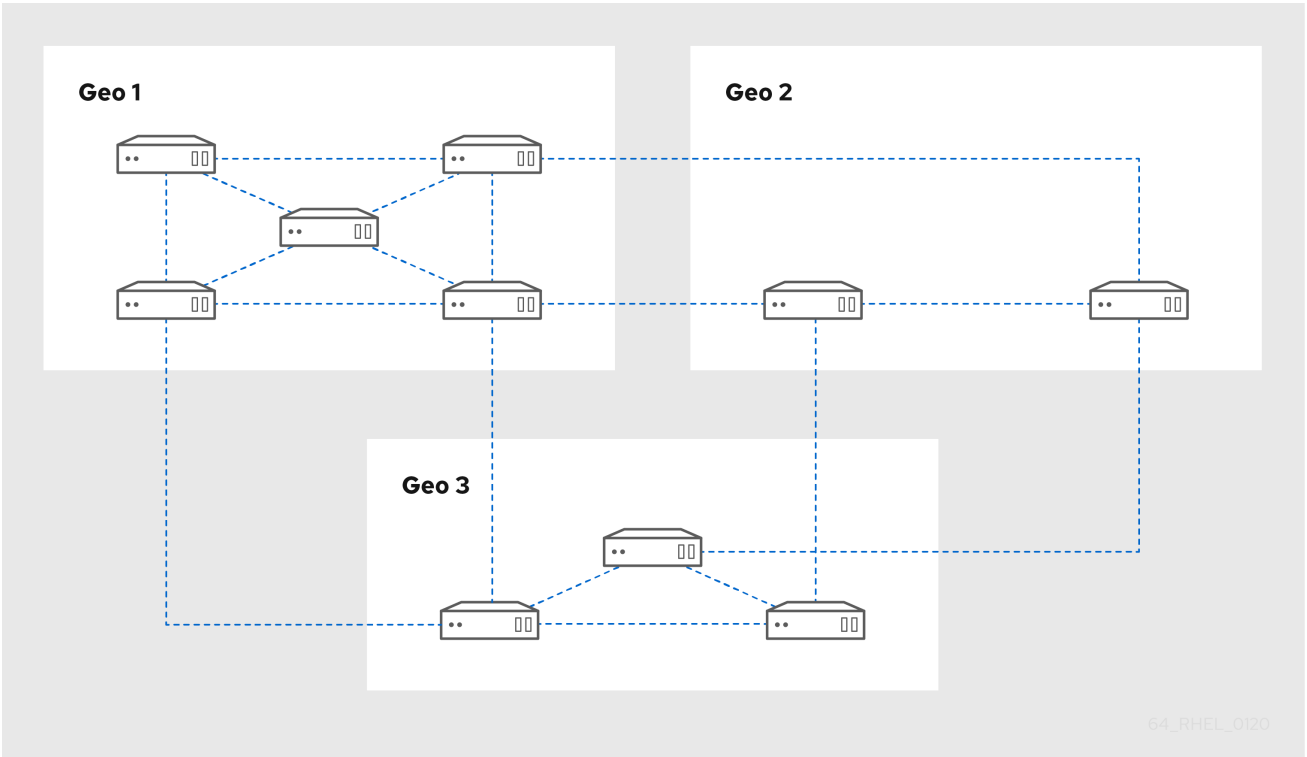


図3.3 3 つのデータセンターで構成されるレプリカトポロジー。各データセンターに異なる数のサーバーがあり、それらがすべてレプリカ合意を通じて相互接続されている



3.7. 非表示のレプリカモード

非表示のレプリカは、稼働中および利用できるすべてのサービスを持つ IdM サーバーです。ただし、非表示のレプリカには DNS に SRV レコードがなく、LDAP サーバーロールが有効になっていません。そのため、クライアントはサービス検出を使用して非表示のレプリカを検出することができません。

デフォルトでは、レプリカをセットアップすると、インストールプログラムによって DNS にサービス (SRV) リソースレコードが自動的に作成されます。このレコードにより、クライアントはレプリカとそのサービスを自動検出できます。レプリカを非表示のレプリカとしてインストールする場合は、**ipa-replica-install** コマンドに **--hidden-replica** パラメーターを渡します。

非表示のレプリカは主に、クライアントの停止を引き起こす可能性のある専用サービス用に設計されています。たとえば、IdM の完全バックアップでは、サーバー上のすべての IdM サービスをシャットダウンする必要があります。非表示のレプリカを使用するクライアントはないため、管理者はクライアントに影響を与えることなく、このホスト上のサービスを一時的にシャットダウンできます。

その他のユースケースには、大量インポートや詳細なクエリーなど、IdM API または LDAP サーバーの高負荷操作が含まれます。

非表示のレプリカをバックアップする前に、クラスターで使用されるすべての必要なサーバーロール (特に統合 CA が使用されている場合は Certificate Authority ロール) をインストールする必要があります。したがって、新しいホストで非表示のレプリカからバックアップを復元すると、常に通常のレプリカが作成されます。

関連情報

- Identity Management レプリカのインストール
- IdM のバックアップおよび復元
- 非表示レプリカの降格または昇格

第4章 DNS サービスとホスト名の計画

Identity Management (IdM) は、IdM サーバーにさまざまな DNS 設定を提供します。以下のセクションでは、各設定を説明し、ユースケースに最適なものを判断するためのアドバイスを提供します。

4.1. IDM サーバーで利用可能な DNS サービス

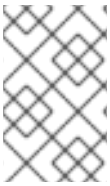
Identity Management サーバー (IdM) は、統合 DNS の使用に関わらずインストールできます。

表4.1 統合 DNS がある IdM と統合 DNS のない IdM の比較

	統合 DNS あるサーバー	統合 DNS のないサーバー
概要:	IdM は、IdM ドメインに独自の DNS サービスを実行します。	IdM は、外部 DNS サーバーが提供する DNS サービスを使用します。
制限:	IdM が提供する統合 DNS サーバーは、IdM のデプロイメントとメンテナンスに関連する機能のみに対応します。汎用 DNS サーバーの高度な機能の一部はサポートされていません。具体的な制限は次のとおりです。 ● IdM DNS ネームサーバーは、そのゾーンに対して権限を持つ必要があります。 ● サポートされているレコードタイプは、A、AAAA、A6、AFSDB、CERT、CNAME、DLV、DNAME、DS、KX、LOC、MX、NAPTR、NS、PTR、SRV、SSHFP、TLSA、TXT、および URI です。 ● スプリット DNS (スプリットビュー、スプリットホライズン、スプリットブレイン DNS とも呼ばれます) はサポートされていません。 ● マルチコア環境で DNS ネームサーバーが再起動する場合は、既知の問題があります。たとえば、ログのローテーションによりネームサーバーが再起動すると、ネームサーバーがクラッシュする可能性があります。マルチコア設定を使用する必要がある場合は、障害発生後に **systemd** がネームサーバーを再起動できるようにします。	DNS は、ネイティブの IdM ツールとは統合されません。たとえば、IdM は、トポロジーの変更後に DNS レコードを自動的に更新しません。
最適な条件:	IdM デプロイメントにおける基本的な使用方法。 IdM サーバーで DNS を管理する際に、DNS はネイティブの IdM ツールと密接に統合されるため、DNS レコードの管理タスクの一部を自動化できます。	IdM DNS のスコープを超える高度な DNS 機能が必要な環境。 外部 DNS サーバーの使用を維持する必要のある、適切に確立された DNS インフラストラクチャーがある環境。

Identity Management サーバーがプライマリー DNS サーバーとして使用されている場合でも、その他の外部 DNS サーバーはセカンダリーサーバーとしても使用できます。たとえば、Active Directory (AD) と統合されている DNS サーバーなどの別の DNS サーバーを、環境がすでに使用している場合は、IdM

のプライマリードメインのみを、IdM と統合している DNS に委譲できます。DNS ゾーンの IdM DNS への移行は必要ありません。



注記

SAN (Subject Alternative Name) 拡張機能の IP アドレスを使用して IdM クライアントの証明書を発行する必要がある場合は、IdM 統合 DNS サービスを使用する必要があります。

4.2. DNS ドメイン名および KERBEROS レルム名を計画するためのガイドライン

最初の Identity Management (IdM) サーバーをインストールする場合は、インストールに、IdM ドメイン名および Kerberos レルム名の入力が求められます。これらのガイドラインは、名前を正しく設定するのに役立ちます。

警告

サーバーをインストールしてから、IdM のプライマリードメイン名および Kerberos レルム名を変更することはできません。この名前を変更し (例: **lab.example.com** から **production.example.com** へ)、テスト環境で実稼働環境に移行することは意図していません。

サービスレコード用の個別の DNS ドメイン

IdM に使用されている **プライマリー DNS ドメイン** が他のシステムと共有されていないことを確認してください。これにより、DNS レベルでの競合が回避されます。

適切な DNS ドメイン名委譲

DNS ドメインのパブリック DNS ツリーで有効な委任があることを確認します。プライベートネットワーク上でも委譲されていないドメイン名は使用しないでください。

マルチラベルの DNS ドメイン

シングルラベルのドメイン名 (**.company** など) は使用しないでください。IdM ドメインは、トップレベルドメインと、1つ以上のサブドメイン (**example.com** や **company.example.com** など) で設定する必要があります。

一意の Kerberos レルム名

レルム名が、Active Directory (AD) が使用する名前など、その他の既存の Kerberos レルム名と競合していないことを確認します。

Kerberos レルム名 (プライマリー DNS 名の大文字バージョン)

レルム名を、プライマリー DNS ドメイン名 (**example.com**) の大文字 (**EXAMPLE.COM**) に設定することを検討してください。

警告

Kerberos レルム名をプライマリー DNS 名の大文字に設定しない場合は、AD 信頼を使用することができません。

DNS ドメイン名および Kerberos レルム名の計画に関する注意点

- IdM デプロイメントでは、常に Kerberos レルムが 1 つだけ使用されます。
- 複数の DNS ドメイン (**example.com**、**example.net**、**example.org**) にある IdM クライアントを、 1 つの Kerberos レルム (**EXAMPLE.COM**) に統合できます。
- IdM クライアントは、プライマリー DNS ドメインに置く必要がありません。たとえば、IdM ドメインが **idm.example.com** の場合、クライアントは **clients.example.com** ドメインに指定できますが、DNS ドメインと Kerberos レルムとの間でマッピングを設定する必要があります。

注記

マッピングを作成する標準的な方法は、**_kerberos** TXT DNS レコードを使用することです。IdM 統合 DNS は、このレコードを自動的に追加します。

DNS 転送の計画

- IdM デプロイメント全体に 1 つのフォワーダーのみを使用する場合は、**グローバルフォワーダー** を設定します。
- 地理的に離れた地域にある複数のサイトに会社が分散している場合は、グローバルフォワーダーが実用的ではない可能性があります。**サーバーごとのフォワーダー** を設定します。
- 会社に、パブリックインターネットでは解決できない内部 DNS ネットワークがある場合は、IdM ドメインのホストがこの他の内部 DNS ネットワークからホストを解決できるように、**転送ゾーン** と **ゾーンフォワーダー** を設定します。

第5章 CA サービスの計画

Red Hat Enterprise Linux の Identity Management (IdM) は、さまざまな認証局 (CA) 設定を提供します。以下のセクションでは、さまざまなシナリオを紹介し、ユースケースに最適な設定を選択するのに役に立つアドバイスを提供します。

CA 発行先 DN

認証局 (CA) 発行先識別名 (DN) は CA の名前です。Identity Management CA インフラストラクチャーではグローバルに一意である必要があり、インストール後に変更することはできません。IdM CA を外部に署名する必要がある場合は、外部 CA の管理者に、IdM CA 発行先識別名の形式を問い合わせる必要がでてくる場合もあります。

5.1. IDM サーバーで利用可能な CA サービス

Identity Management (IdM) サーバーは、統合 IdM 認証局 (CA) を使用、または使用せずにインストールできます。

表5.1 統合 CA を使用した IdM と、CA を使用しない IdM の比較

	統合 CA あり	CA なし
概要:	IdM は、独自の公開鍵インフラストラクチャー (PKI) サービスを **CA 署名の証明書** と共に使用して、IdM ドメインで証明書を作成して署名します。 ● ルート CA が統合 CA の場合、IdM は自己署名の CA 証明書を使用します。 ● ルート CA が外部 CA の場合、統合 IdM CA は外部 CA の下位局になります。IdM が使用する CA 証明書は外部 CA により署名されますが、IdM ドメインのすべての証明書は、統合証明書システムインスタンスにより発行されます。 ● 統合 CA は、ユーザー、ホスト、またはサービスの証明書を発行することもできます。 外部 CA は、企業 CA またはサードパーティーの CA です。	IdM は独自の CA を設定しませんが、外部 CA の署名付きホスト証明書を使用します。 CA を使用せずにサーバーをインストールするには、サードパーティーの認証局から以下の証明書を要求する必要があります。 ● LDAP サーバー証明書 ● Apache サーバー証明書 ● PKINIT 証明書 ● LDAP および Apache のサーバー証明書を発行した CA の完全な CA 証明書チェーン

	統合 CA あり	CA なし
制限:	統合 CA が外部 CA の下位局になる場合、IdM ドメインで発行された証明書は、以下を含むさまざまな証明書属性用の外部 CA により設定される制限の影響を受ける可能性があります。 ● 有効期間 ● IDM CA またはその下位局が発行する証明書に表示されるサブジェクト名に関する制約 ● IDM CA 自身が中間 CA 証明書を発行するかどうか、中間証明書チェーンがどのぐらい深くなるかに関する制約	IdM 以外で証明書を管理すると、以下のような多くの追加アクティビティーが発生します。 ● 証明書の作成、アップロード、および更新は手動のプロセスです。 ● **certmonger** サービスは、IPA 証明書 (LDAP サーバー、Apache サーバー、および PKINIT 証明書) を追跡せず、証明書が期限切れになる際に通知がされません。管理者は、外部に発行される証明書に関する通知を手動で設定したり、**certmonger** が証明書を追跡する必要がある場合に証明書の追跡要求を設定したりする必要があります。
最適な条件:	証明書インフラストラクチャーを作成および使用できるようにする環境。	インフラストラクチャーの制限により、サーバーと統合されている証明書サービスをインストールすることができない場合は、非常に稀なケースとなります。

注記

自己署名の CA から外部署名の CA への切り替え (またはその逆)、もしくは IdM CA 証明書を発行する外部 CA の変更は、インストール後も可能になります。CA を使用せずにインストールしてから、統合 CA を設定することもできます。詳細は、Installing an IdM server: With integrated DNS, without a CA を参照してください。

関連情報

- IdM が内部で使用する証明書について

5.2. CA サービスの配布ガイドライン

以下の手順は、認証局 (CA) サービス配布のガイドラインを提供します。

手順

1. トポロジー内の複数のサーバーに CA サービスをインストールします。
   CA を使用せずに設定されたレプリカは、トポロジー内のすべての証明書操作要求を CA サーバーに転送します。

警告

CA を使用するすべてのサーバーが失われると、すべての CA 設定が失われ、復元できません。この場合は、新しい CA を設定し、新しい証明書を発行してインストールする必要があります。

2. デプロイメントで CA 要求を処理するのに十分な数の CA サーバーを維持します。

適切な数の CA サーバーに関する詳細な推奨事項については、次の表を参照してください。

表5.2 適切な CA サーバー数を設定するためのガイドライン

デプロイメントの説明	CA サーバーの推奨数
発行された証明書の数が非常に多いデプロイメント	3 台から 4 台の CA サーバー
複数のリージョン間での帯域幅または可用性問題があるデプロイメント	リージョンごとに、デプロイメント用に合計 3 台以上のサーバーを持つ 1 台の CA サーバー
その他すべてのデプロイメント	2 台の CA サーバー

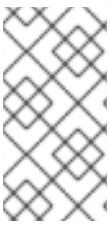
重要

同時証明書要求の数が多くない場合は、通常、トポロジー内の 4 つの CA サーバーで十分です。4 つを超える CA サーバー間でレプリケーションプロセスを実行すると、プロセッサーの使用量が増加し、パフォーマンスの低下につながる可能性があります。

5.3. IDM のランダムなシリアル番号

RHEL 9.1 以降、Identity Management (IdM) には **dogtagpki 11.2.0** が含まれており、これにより Random Serial Numbers バージョン 3 (RSNv3) を使用できるようになります。**ansible-freeipa ipaserver** ロールには、RHEL 9.3 アップデートの **ipaserver_random_serial_numbers** 変数が含まれています。

RSNv3 を有効にすると、IdM は範囲管理なしで PKI の証明書とリクエストに対して完全にランダムなシリアル番号を生成します。RSNv3 は、IdM を再インストールした場合の競合も阻止します。RSNv3 はシリアル番号に 128 ビットのランダムな値を使用するため、各証明書のシリアル番号のサイズは最大 40 桁の 10 進数値になります。これにより、数値は事実上ランダムになります。



注記

以前、Dogtag アップストリームプロジェクトでは、複数のクローン間での一意性を確保するために、範囲ベースのシリアル番号を使用していました。ただし、この経験に基づいて、Dogtag チームは、範囲ベースのシリアル番号は、有効期間の短い証明書を使用するクラウド環境にはうまく適合しないと判断しました。

RSNv3 は、新しい IdM CA インストールでのみサポートされます。デフォルトでは、**ipa-server-install** コマンドを使用してプライマリー IdM サーバーをインストールするときに、最初の IdM CA をインス

トールします。ただし、最初に CA なしで IdM 環境をインストールした場合は、後で **ipa-ca-install** コマンドを使用して CA サービスを追加できます。RSNv3 を有効にするには、**--random-serial-numbers** オプションを指定して、**ipa-server-install** または **ipa-ca-install** コマンドを使用します。

有効にした場合、CA や Key Recovery Authority (KRA) を含む、デプロイメント内のすべての公開鍵インフラストラクチャー (PKI) サービスで RSNv3 を使用する必要があります。KRA のインストール時にチェックが実行され、基盤となる CA で RSNv3 が有効になっている場合は自動的に有効になります。

関連情報

- Random Serial Numbers v3 (RSNv3)

第6章 AD を使用した統合の計画

以下のセクションでは、Red Hat Enterprise Linux と Active Directory (AD) を統合するためのオプションを紹介します。

6.1. LINUX システムの ACTIVE DIRECTORY への直接統合

直接統合では、Linux システムは、Active Directory (AD) に直接接続されています。次の種類の統合が可能です。

System Security Services Daemon (SSSD) との統合

SSSD は、Linux システムをさまざまな ID および認証ストア (AD、Identity Management (IdM)、もしくは汎用の LDAP サーバーまたは Kerberos サーバー) に接続できます。
SSSD の統合に関する重要な要件

- AD と統合すると、SSSD は、デフォルトで 1 つの AD フォレスト内でのみ機能します。マルチフォレストを設定する場合は、ドメインのエミュレーションを手動で設定します。
- **idmap_ad** プラグインがリモートフォレストユーザーを正常に処理するには、リモートの AD フォレストがローカルフォレストを信頼する必要があります。

SSSD は、直接統合と間接統合の両方に対応します。また、莫大な移行コストをかけずに、ある統合アプローチから別のアプローチへ切り替えることもできます。

Samba Winbind との統合

Samba スイートの Winbind コンポーネントは、Linux システムで Windows クライアントをエミュレートし、AD サーバーと通信します。
Samba Winbind の統合に関する重要な要件

- マルチフォレストの AD 設定における Winbind との直接統合は、双方向の信頼が必要になります。
- リモートの AD ドメインユーザーに関する完全な情報を **idmap_ad** プラグインで使用できるようにするには、Linux システムのローカルドメインから、ユーザーが所属するリモートの AD フォレスト内ドメインへの双方向パスが存在する必要があります。

推奨事項

- SSSD は、AD 統合のほとんどのユースケースに対応し、クライアントシステムとさまざまな ID および認証プロバイダー (AD、IdM、Kerberos、および LDAP) との間の汎用ゲートウェイとして堅牢なソリューションを提供します。
- Samba FS をデプロイする予定の AD ドメインメンバーサーバーへのデプロイには、Winbind が推奨されます。

6.2. アイデンティティー管理を使用した LINUX システムの ACTIVE DIRECTORY への間接統合

間接統合により、Linux システムが最初に集中型サーバーに接続し、次に集中型サーバーが Active Directory (AD) に接続します。間接統合により、管理者は Linux システムとポリシーを一元管理でき、AD のユーザーは透過的に Linux システムとサービスにアクセスできます。

AD を使用したフォレスト間の信頼に基づく統合

Identity Management (IdM) サーバーは、Linux システムを制御する集中型サーバーとして機能します。AD を使用したレルム間の Kerberos 信頼が確立され、AD のユーザーが Linux システムおよびリソースにログインしてアクセスできるようになります。IdM は、それ自体を別のフォレストとして AD に提示し、AD で対応しているフォレストレベルの信頼を利用します。
信頼を使用すると、以下が可能になります。

- AD ユーザーは、IdM リソースにアクセスできます。
- IdM サーバーおよびクライアントは、AD のユーザーおよびグループの ID を解決できます。
- AD ユーザーおよびグループは、ホストベースのアクセス制御など、IdM が定義する条件下で IdM にアクセスします。
- AD ユーザーおよびグループは、引き続き AD 側で管理されます。

同期に基づく統合

このアプローチは WinSync ツールに基づいています。WinSync レプリカ合意は、AD から IdM へユーザーアカウントを同期します。

警告

WinSync は、Red Hat Enterprise Linux 8 で積極的に開発されなくなりました。間接統合に推奨されるソリューションはフォレスト間の信頼です。

同期に基づく統合の制限は次のとおりです。

- グループは、IdM から AD に同期されません。
- AD と IdM にユーザーが重複しています。
- WinSync は、1 つの AD ドメインのみをサポートします。
- IdM 内の 1 つのインスタンスへのデータ同期には、AD のドメインコントローラーを 1 つだけ使用できます。
- ユーザーパスワードを同期する必要があります。そのためには、PassSync コンポーネントを AD ドメイン内のすべてのドメインコントローラーにインストールする必要があります。
- すべての AD ユーザーは、同期を設定してから手動でパスワードを変更しないと、PassSync を同期できません。

6.3. 直接統合と間接統合を決定するためのガイドライン

これらのガイドラインは、どのタイプの統合が自分のユースケースに適しているかを判断するのに役立ちます。

Active Directory に接続するシステムの数

30 ~ 50 台未満のシステムを接続 (必須要件ではない)

30 ~ 50 台未満のシステムを接続する場合は、直接統合を検討してください。間接統合により、不要なオーバーヘッドが発生する可能性があります。

30 - 50 台を超えるシステムを接続 (必須制限ではない)

30 ~ 50 台を超えるシステムを接続する場合は、Identity Management を使用した間接統合を検討してください。このアプローチでは、Linux システムの一元管理の恩恵を受けることができます。

管理する Linux システムの数は少ないが、今後急増する見込み

このシナリオでは、間接的な統合を検討し、後で環境を移行しなくても済むようにします。

新しいシステムをデプロイする頻度とその種類

ベアメタルシステムの不規則なデプロイメント

新しいシステムをデプロイすることがほとんどなく、通常はベアメタルシステムをデプロイする場合は、直接統合を検討してください。そのような場合、直接統合は、通常、最も単純で簡単です。

仮想システムの頻繁なデプロイメント

新しいシステムを頻繁にデプロイし、それが通常オンデマンドでプロビジョニングされた仮想システムである場合は、間接統合を検討してください。間接統合では、集中型サーバーを使用して新しいシステムを動的に管理し、Red Hat Satellite などのオーケストレーションツールと統合できます。

Active Directory が必須の認証プロバイダーである

すべてのユーザーが Active Directory に対して認証を行う必要があると、内部ポリシーに記載されていますか?

直接統合または間接統合のいずれかを選択できます。Identity Management と Active Directory との間の信頼を使用して間接統合を使用する場合、Linux システムにアクセスするユーザーは、Active Directory に対して認証を行います。Active Directory に存在するポリシーは、認証中に実行され適用されます。

第7章 IDM と AD との間のフォレスト間の信頼の計画

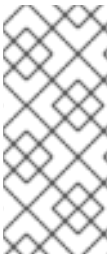
Active Directory (AD) および Identity Management (IdM) は、Kerberos、LDAP、DNS、証明書サービスなどのさまざまなコアサービスを管理する 2 つの代替環境です。**フォレスト間の信頼** 関係は、すべてのコアサービスがシームレスに相互作用できるようにすることで、その 2 つの異なる環境を透過的に統合します。次のセクションでは、フォレスト間の信頼のデプロイメントを計画して設計する方法のヒントを紹介します。

7.1. IDM と AD の間のフォレスト間と外部の信頼

IdM と AD の間のフォレスト間の信頼

純粋な Active Directory (AD) 環境では、フォレスト間の信頼は、2 つの AD フォレストルートドメインに接続します。AD と IdM との間のフォレスト間の信頼を作成すると、IdM ドメインは、それ自体を 1 つのドメインを持つ別のフォレストとして AD に提示します。その後、AD フォレストのルートドメインと IdM ドメインの間に信頼関係が確立されます。これにより、AD フォレストのユーザーは、IdM ドメインのリソースにアクセスできます。

IdM は、1 つの AD フォレスト、または関連のない複数のフォレストとの信頼関係を確立できます。



注記

cross-realm trust で、2 つの Kerberos レルムを接続できます。ただし、Kerberos レルムは認証にのみ関係し、識別操作および認可操作に関連するその他のサービスおよびプロトコルには関係しません。したがって、Kerberos のレルム間の信頼を確立しても、あるレルムのユーザーが別のレルムのリソースにアクセスできるようにするには不十分です。

AD ドメインへの外部の信頼

外部の信頼は、IdM と AD ドメインとの間の信頼関係です。フォレストの信頼では常に IdM と Active Directory フォレストのルートドメインとの間で信頼関係を確立する必要がありますが、IdM からフォレスト内の任意のドメインへの外部の信頼関係も確立できます。

7.2. 信頼コントローラーおよび信頼エージェント

Identity Management (IdM) には、Active Directory (AD) への信頼をサポートする、以下のタイプの IdM サーバーがあります。

信頼コントローラー

AD ドメインコントローラーで ID 検索が実行可能な IdM サーバーまた、Samba スイートも実行するため、AD との信頼を確立できます。AD ドメインコントローラーは、AD への信頼を確立して検証する際に信頼コントローラーに問い合わせます。AD に登録したマシンは、Kerberos 認証要求で IdM 信頼コントローラーと通信します。
信頼を設定すると、最初の信頼コントローラーが作成されます。地理的に異なる場所に複数のドメインコントローラーがある場合は、**ipa-adtrust-install** コマンドを使用して、RHEL IdM サーバーを、その場所で信頼コントローラーとして指定します。

信頼コントローラーは、信頼エージェントと比較すると、ネットワーク向けサービスを多く実行するため、侵入者が攻撃できる範囲が大きくなります。

信頼エージェント

AD ドメインコントローラーに対する RHEL IdM クライアントからの ID 検索を解決できる IdM サーバー。信頼コントローラーとは異なり、信頼エージェントは Kerberos 認証要求を処理できません。

IdM ドメインには、信頼エージェントと信頼コントローラーだけでなく、標準の IdM サーバーも追加できます。ただし、このサーバーは AD と通信しません。したがって、これらの標準サーバーと通信するクライアントは、AD ユーザーとグループを解決したり、AD ユーザーを認証および承認したりすることはできません。

注記

以下のアクションのいずれかが実行されない限り、IdM サーバーは Trust Controller または Trust Agent ロールを操作するように設定されません。

- **--setup-ad** オプションを指定した **ipa-server-install** または **ipa-replica-install** コマンドでサーバーまたはレプリカをインストールした。
- IdM サーバーで **ipa-adtrust-install** コマンドを実行して、Trust Controller ロールを設定しました。
- Trust Controller で **ipa-adtrust-install --add-agents** コマンドを実行して、別の IdM レプリカを Trust Agent に指定しました。
  デフォルトでは、IdM サーバーは、これらの操作を行わないと、信頼されたドメインからユーザーおよびグループを解決できません。

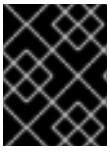
表7.1 信頼コントローラーおよび信頼エージェントが提供する機能の比較

機能	信頼エージェント	信頼コントローラー
AD ユーザーおよびグループを解決する	○	○
IdM クライアントを登録して、信頼されている AD フォレストのユーザーがアクセスできるサービスの実行	○	○
信頼アグリーメントの追加、変更、または削除	いいえ	○
トラストエージェントロールを IdM サーバーに割り当てます。	いいえ	○

信頼コントローラーと信頼エージェントのデプロイメントを計画する時に、以下のガイドラインを考慮してください。

- IdM のデプロイメントごとに、信頼コントローラーを少なくとも 2 台設定する。
- 各データセンターごとに、信頼コントローラーを少なくとも 2 台設定する。

追加の信頼コントローラーを作成する場合や、既存の信頼コントローラーが失敗した場合には、信頼エージェントまたは標準サーバーを昇格して、信頼コントローラーを新規作成してください。これには、IdM サーバーの **ipa-adtrust-install** ユーティリティーを使用してください。



重要

既存の信頼コントローラーを信頼エージェントにダウングレードすることはできません。

7.3. 一方向および双方向の信頼

一方向の信頼関係では、Identity Management (IdM) は Active Directory (AD) を信頼しますが、AD は IdM を信頼しません。AD ユーザーは IdM ドメイン内のリソースにアクセスできますが、IdM のユーザーは AD ドメインのリソースにアクセスできません。IdM サーバーは、特別なアカウントを使用して AD に接続し、ID 情報を読み取り、それを LDAP 経由で IdM クライアントに配信します。

双方向の信頼では、IdM ユーザーは AD に対して認証でき、AD ユーザーは IdM に対して認証できます。一方向の信頼の場合と同様、AD ユーザーは IdM ドメイン内のリソースに対して認証およびアクセスできます。IdM ユーザーは認証できますが、AD のほとんどのリソースにアクセスすることはできません。IdM ユーザーは、アクセス制御チェックを必要としない、AD フォレスト内の Kerbers 対応サービスにのみアクセスできます。

AD リソースへのアクセスを許可できるようにするには、IdM は Global Catalog サービスを実装する必要があります。IdM サーバーの現在のバージョンにはこのサービスがありません。そのため、IdM と AD との間の双方向の信頼は、IdM と AD との間の一方向の信頼と機能的にほぼ同等です。

7.4. AD および RHEL で一般的な暗号化タイプに対応

デフォルトでは、Identity Management は RC4、AES-128、および AES-256 の Kerberos 暗号化タイプに対応するレルム間の信頼を確立します。さらに、デフォルトでは、SSSD と Samba Winbind は RC4、AES-128、および AES-256 の Kerberos 暗号化タイプに対応します。

RC4 暗号化は、新しい暗号化タイプ AES-128 および AES-256 よりも安全ではないと見なされるため、デフォルトで非推奨となり、無効にされています。一方、Active Directory (AD) ユーザーの認証情報と AD ドメイン間の信頼は RC4 暗号化をサポートしており、すべての AES 暗号化タイプには対応していない可能性があります。

一般的な暗号化タイプがないと、RHEL ホストと AD ドメイン間の通信が機能しないか、一部の AD アカウントが認証できない可能性があります。この状況に対処するには、次のセクションで説明する設定のいずれかを実行します。

重要

IdM が FIPS モードの場合、IdM-AD 統合は機能しません。これは、AD は RC4 または AES HMAC-SHA1 暗号化の使用しかサポートしない一方で、FIPS モードの RHEL 9 は、デフォルトでは AES HMAC-SHA2 しか許可しないためです。RHEL 9 で AES HMAC-SHA1 の使用を有効にするには、**# update-crypto-policies --set FIPS:AD-SUPPORT** と入力してください。

IdM は、より制限の厳しい **FIPS:OSPP** 暗号化ポリシーはサポートしていません。このポリシーは、Common Criteria で評価されたシステムでしか使用できません。

7.4.1. AD での AES 暗号化の有効化 (推奨)

AD フォレストの Active Directory (AD) ドメイン間の信頼を確保して、強力な AES 暗号化の種類に対応するには、Microsoft の記事 AD DS: Security: Kerberos "Unsupported etype" error when accessing a resource in a trusted domain を参照してください。

7.4.2. GPO を使用した Active Directory で AES 暗号化タイプの有効化

本セクションでは、グループポリシーオブジェクト (GPO) を使用して、Active Directory (AD) で AES 暗号化タイプを有効にする方法を説明します。IdM クライアントで Samba サーバーを実行するなど、RHEL の特定の機能には、この暗号化タイプが必要です。

RHEL は、弱い DES および RC4 の暗号化タイプをサポートしなくなった点に注意してください。

前提条件

- グループポリシーを編集できるユーザーとして AD にログインしている。
- **Group Policy Management Console** がコンピューターにインストールされている。

手順

1. **Group Policy Management Console** を開きます。
2. **デフォルトドメインポリシー** を右クリックして、**編集** を選択します。**Group Policy Management Editor** を閉じます。
3. **コンピューターの設定 → ポリシー → Windows の設定 → セキュリティーの設定 → ローカルポリシー → セキュリティーオプション** に移動します。
4. **ネットワーク セキュリティー: Kerberos で許可する暗号化の種類を設定する** をダブルクリックします。
5. **AES256_HMAC_SHA1** を選択し、必要に応じて、**将来の暗号化タイプ** を選択します。
6. **OK** をクリックします。
7. **Group Policy Management Editor** を閉じます。
8. **デフォルトのドメインコントローラーポリシー** に対して手順を繰り返します。
9. Windows ドメインコントローラー (DC) がグループポリシーを自動的に適用するまで待ちます。または、GPO を DC に手動で適用するには、管理者権限を持つアカウントを使用して次のコマンドを入力します。

```
C:\> gpupdate /force /target:computer
```

7.4.3. RHEL での RC4 サポートの有効化

AD ドメインコントローラーに対する認証が行われるすべての RHEL ホストで、以下に概説する手順を実行します。

手順

1. **update-crypto-policies** コマンドを使用して、**DEFAULT** 暗号化ポリシーに加え **AD-SUPPORT-LEGACY** 暗号化サブポリシーを有効にします。

```
[root@host ~]# update-crypto-policies --set DEFAULT:AD-SUPPORT-LEGACY
Setting system policy to DEFAULT:AD-SUPPORT-LEGACY
Note: System-wide crypto policies are applied on application start-up.
It is recommended to restart the system for the change of policies
to fully take place.
```

2. ホストを再起動します。

7.4.4. 関連情報

- Using system-wide cryptographic policies を参照してください。

- 信頼コントローラーおよび信頼エージェント を参照してください。

7.5. 信頼できるドメインの KERBEROS FAST

Kerberos Flexible Authentication Secure Tunneling (FAST) は、Active Directory (AD) 環境では Kerberos アーマー機能とも呼ばれます。Kerberos FAST は、クライアントと KDC (Key Distribution Center) 間の Kerberos 通信に追加のセキュリティー層を提供します。IdM では、KDC は IdM サーバーで実行しており、FAST はデフォルトで有効になっています。IdM の 2 要素認証 (2FA) では FAST も有効にする必要があります。

AD では、AD ドメインコントローラー (DC) で、Kerberos アーマー機能はデフォルトで無効になっています。**Tools > Group Policy Management > Default Domain Controller Policy** のドメインコントローラーで有効にできます。

- **Default Domain Controller Policy** を右クリックし、**edit** を選択します。**Computer Configuration>Policies>Administrative Templates>System>KDC** に移動し、**KDC support for claims, compound authentication, and Kerberos armoring** をダブルクリックします。

クレームの KDC サポートを有効にすると、ポリシー設定で次のオプションが許可されます。

- サポート対象外
- サポート対象
- "Always provide claims"
- "Fail unarmored authentication requests"

Kerberos FAST は、IdM クライアントの Kerberos クライアントライブラリーに実装されます。IdM クライアントは、FAST を通知するすべての信頼されたドメインに FAST を使用するように設定するか、Kerberos FAST をまったく使用しないように設定できます。信頼できる AD フォレストで Kerberos アーマーを有効にすると、IdM クライアントはデフォルトで Kerberos FAST を使用します。FAST は、暗号鍵を使用してセキュアなトンネルを確立します。信頼されたドメインのドメインコントローラーへの接続を保護するために、Kerberos FAST は信頼されたドメインからクロスレルムチケット保証チケット (TGT) を取得する必要があります。これは、これらのキーが Kerberos レルム内でのみ有効であるためです。Kerberos FAST は、IdM クライアントの Kerberos hosts キーを使用し、IdM サーバーの支援でレルム間の TGT を要求します。これは、AD フォレストが IdM ドメインを信頼する場合にのみ機能します。これは、双方向の信頼が必要であることを意味します。

AD ポリシーで Kerberos FAST の使用を強制する必要がある場合は、IdM ドメインと AD フォレストとの間で双方向の信頼を確立する必要があります。IdM と AD の両方に、方向および信頼タイプに関するレコードが必要であるため、接続を確立する前にこれを計画する必要があります。

一方向の信頼をすでに確立している場合は、**ipa trust-add … --two-way=true** コマンドを実行して既存の信頼合意を削除し、双方向の信頼を作成します。これには、管理資格証明を使用する必要があります。IdM は、AD 側から既存の信頼合意を削除しようとするため、AD アクセスに管理者権限が必要です。AD 管理アカウントではなく共有秘密を使用して元の信頼を確立すると、信頼が双方向として再作成され、信頼されたドメインオブジェクトが IdM 側でのみ変更されます。Windows 管理者は、双方向の信頼を選択し、同じ共有秘密を使用して Windows UI で同じ手順を繰り返して信頼を再作成する必要があります。

双方向の信頼を使用できない場合は、すべての IdM クライアントで Kerberos FAST を無効にする必要があります。信頼できる AD フォレストのユーザーは、パスワードまたはダイレクトスマートカードで認証できます。Kerberos FAST を無効にするには、**sssd.conf** ファイルの **domain** セクションに次の設定を追加します。

```
krb5_use_fast = never
```

認証がリモートの Windows クライアントの ssh-keys、GSSAPI 認証、またはスマートカードを使用した SSH に基づく場合は、このオプションを使用する必要がありません。IdM クライアントは DC と通信する必要がないため、このようなメソッドは Kerberos FAST を使用しません。また、IdM クライアントで FAST を無効にすると、2 要素認証の IdM 機能も利用できなくなります。

7.6. AD ユーザー向けの POSIX および ID マッピング ID の範囲タイプ

Identity Management (IdM) は、ユーザーの POSIX ユーザー ID (UID) およびグループ ID (GID) に基づいてアクセス制御ルールを強制します。ただし、Active Directory (AD) ユーザーはセキュリティー識別子 (SID) で識別されます。AD 管理者は、AD ユーザーおよびグループ (**uidNumber**、**gidNumber**、**unixHomeDirectory**、**loginShell** など) の POSIX 属性を保存するように AD を設定できます。

ipa-ad-trust-posix ID 範囲で信頼を確立することで、この情報を参照するようにフォレスト間の信頼を設定できます。

```
[server ~]# ipa trust-add --type=ad ad.example.com --admin administrator --password  --range-type=ipa-ad-trust-posix
```

AD に POSIX 属性を保存しない場合、SSSD (System Security Services Daemon) は、**ID マッピング** と呼ばれるプロセスにおけるユーザーの SID に基づいて一意の UID を常にマッピングできます。**ipa-ad-trust** ID の範囲で信頼を作成することにより、この動作を明示的に選択できます。

```
[server ~]# ipa trust-add --type=ad ad.example.com --admin administrator --password  --range-type=ipa-ad-trust
```

警告

信頼の作成時に ID 範囲タイプを指定しないと、IdM はフォレストルートドメインの AD ドメインコントローラーから詳細を要求することで、適切な範囲タイプを自動的に選択しようとします。IdM が POSIX 属性を検出しない場合、信頼インストールスクリプトは **Active Directory domain** ID 範囲を選択します。

IdM がフォレストルートドメインの POSIX 属性を検出すると、信頼インストールスクリプトは、**Active Directory domain with POSIX attributes** ID 範囲を選択し、UID および GID が AD に正しく定義されていることを前提とします。POSIX 属性が AD で正しく設定されていない場合は、AD ユーザーを解決できません。

たとえば、IdM システムへのアクセスを必要とするユーザーおよびグループがフォレストルートドメインの一部ではなく、フォレストドメインの子ドメインにある場合は、インストールスクリプトで、子 AD ドメインで定義された POSIX 属性が検出されない場合があります。この場合、Red Hat は、信頼の確立時に POSIX ID 範囲タイプを明示的に選択することを推奨します。

関連情報

- Options for automatically mapping private groups for AD users

7.7. AD ユーザーのプライベートグループを自動的にマッピングするためのオプション: POSIX の信頼

Linux 環境の各ユーザーには、プライマリーユーザーグループがあります。Red Hat Enterprise Linux (RHEL) は、ユーザープライベートグループ (UPG) スキームを使用します。UPG は、作成したユーザーと同じ名前で、そのユーザーが UPG の唯一のメンバーになります。

AD ユーザーに UID を割り当てているものの、GID が追加されていない場合は、その ID 範囲の auto_private_groups 設定を調整することで、UID に基づいてユーザーのプライベートグループを自動的にマッピングするように SSSD を設定できます。

デフォルトでは、POSIX 信頼で使用される **ipa-ad-trust-posix** ID 範囲では、auto_private_groups オプションは false に設定されています。この設定により、SSSD は、AD ユーザーエントリーごとに **uidNumber** と **gidNumber** を取得します。

auto_private_groups = false

SSSD は、**uidNumber** の値をユーザーの UID に割り当て、**gidNumber** をユーザーの GID に割り当てます。その GID を持つグループが AD に存在している必要があります。存在していないと、そのユーザーを解決できません。以下の表は、AD 設定によって、AD ユーザーを解決できるかどうかを示しています。

表7.2 POSIX ID 範囲でauto_private_groups 変数が false に設定されている場合の SSSD の動作

AD のユーザー設定	id username の出力
AD ユーザーエントリーの内容 • **uidNumber** = 4000 • **gidNumber** は定義されていません。 • AD には、**gidNumber** = 4000 のグループはありません。	SSSD はユーザーを解決できません。
AD ユーザーエントリーの内容 • **uidNumber** = 4000 • **gidNumber** = 4000 • AD には、**gidNumber** = 4000 のグループはありません。	SSSD はユーザーを解決できません。
AD ユーザーエントリーの内容 • **uidNumber** = 4000 • **gidNumber** = 4000 • AD には、**gidNumber** = 4000 のグループがあります。	**# id aduser@AD-DOMAIN.COMuid=4000(aduser@ad-domain.com) gid=4000(adgroup@ad-domain.com) groups=4000(adgroup@ad-domain.com), …**

AD ユーザーにプライマリーグループが設定されていないか、その **gidNumber** が既存のグループに対

応していない場合、IdM サーバーは、そのユーザーが属するすべてのグループを検索できないため、そのユーザーを正しく解決できません。この問題を回避するには、**auto_private_groups** オプションを **true** または **hybrid** に設定して、SSSD で自動プライベートグループマッピングを有効にできます。

auto_private_groups = true

SSSD は、AD ユーザーエントリーの **uidNumber** に一致するように設定された **gidNumber** で、常にプライベートグループをマッピングします。

表7.3 POSIX ID 範囲で auto_private_groups 変数が true に設定されている場合の SSSD の動作

AD のユーザー設定	id username の出力
AD ユーザーエントリーの内容 • **uidNumber** = 4000 • **gidNumber** は定義されていません。 • AD には、GID=4000 のグループがありません。	**# id aduser@AD-DOMAIN.COMuid=4000(aduser@ad-domain.com) gid=4000(aduser@ad-domain.com) groups=4000(aduser@ad-domain.com), …**
AD ユーザーエントリーの内容 • **uidNumber** = 4000 • **gidNumber** = 5000 • AD には、**gidNumber** = 5000 のグループがありません。	**# id aduser@AD-DOMAIN.COMuid=4000(aduser@ad-domain.com) gid=4000(aduser@ad-domain.com) groups=4000(aduser@ad-domain.com), …**
AD ユーザーエントリーの内容 • **uidNumber** = 4000 • **gidNumber** = 4000 • AD には、**gidNumber** = 4000 のグループがありません。	**# id aduser@AD-DOMAIN.COMuid=4000(aduser@ad-domain.com) gid=4000(aduser@ad-domain.com) groups=4000(aduser@ad-domain.com), …**
AD ユーザーエントリーの内容 • **uidNumber** = 4000 • **gidNumber** = 5000 • AD には、**gidNumber** = 5000 のグループがあります。	**# id aduser@AD-DOMAIN.COMuid=4000(aduser@ad-domain.com) gid=4000(aduser@ad-domain.com) groups=4000(aduser@ad-domain.com), …**

auto_private_groups = hybrid

uidNumber の値が **gidNumber** に一致するものの、この **gidNumber** のグループがない場合、SSSD は、プライベートグループを、ユーザーのプライマリーユーザーグループとして、**uidNumber** に一致する **gidNumber** でマッピングします。**uidNumber** と **gidNumber** の値が異なり、この **gidNumber** のグループが存在する場合、SSSD は **gidNumber** の値を使用します。

表7.4 POSIX ID 範囲でauto_private_groups 変数が hybrid に設定されている場合の SSSD の動作

AD のユーザー設定	id username の出力
AD ユーザーエントリーの内容 • **uidNumber** = 4000 • **gidNumber** は定義されていません。 • AD には、**gidNumber** = 4000 のグループがありません。	SSSD はユーザーを解決できません。
AD ユーザーエントリーの内容 • **uidNumber** = 4000 • **gidNumber** = 5000 • AD には、**gidNumber** = 5000 のグループがありません。	SSSD はユーザーを解決できません。
AD ユーザーエントリーの内容 • **uidNumber** = 4000 • **gidNumber** = 4000 • AD には、**gidNumber** = 4000 のグループがありません。	**# id aduser@AD-DOMAIN.COMuid=4000(aduser@ad-domain.com) gid=4000(aduser@ad-domain.com) groups=4000(aduser@ad-domain.com), …**
AD ユーザーエントリーの内容 • **uidNumber** = 4000 • **gidNumber** = 5000 • AD には、**gidNumber** = 5000 のグループがあります。	**# id aduser@AD-DOMAIN.COMuid=4000(aduser@ad-domain.com) gid=5000(aduser@ad-domain.com) groups=5000(adgroup@ad-domain.com), …**

関連情報

- AD ユーザー向けの POSIX および ID マッピング ID の範囲タイプ
- CLI での POSIX ID 範囲の自動プライベートグループマッピングの有効化
- IdM WebUI での POSIX ID 範囲の自動プライベートグループマッピングの有効化

7.8. AD ユーザーのプライベートグループを自動的にマッピングするためのオプション: ID マッピングの信頼

Linux 環境の各ユーザーには、プライマリーユーザーグループがあります。Red Hat Enterprise Linux (RHEL) は、ユーザープライベートグループ (UPG) スキームを使用します。UPG は、作成したユーザーと同じ名前で、そのユーザーが UPG の唯一のメンバーになります。

AD ユーザーに UID を割り当てているものの、GID が追加されていない場合は、その ID 範囲の auto_private_groups 設定を調整することで、UID に基づいてユーザーのプライベートグループを自動的にマッピングするように SSSD を設定できます。

デフォルトでは、**auto_private_groups** オプションは、ID マッピング信頼で使用される **ipa-ad-trust** ID 範囲に対して **true** に設定されています。この設定では、SSSD が、SID (Security Identifier) に基づいて AD ユーザーの UID と GID を計算します。SSSD は、AD の POSIX 属性 (**uidNumber**、**gidNumber** など) を無視します。また、**primaryGroupID** も無視します。

auto_private_groups = true

SSSD は、AD ユーザーの SID に基づいている UID と一致するように設定された GID で、常にプライベートグループをマッピングします。

表7.5 ID マッピング ID 範囲でauto_private_groups 変数が true に設定されている場合の SSSD の動作

AD のユーザー設定	id username の出力
AD ユーザーエントリーの内容 • SID が 7000 にマップされます。 • **primaryGroupID** は 8000 にマップされます。	**# id aduser@AD-DOMAIN.COMuid=7000(aduser@ad-domain.com) gid=7000(aduser@ad-domain.com) groups=7000(aduser@ad-domain.com), 8000(adgroup@ad-domain.com), …**

auto_private_groups = false

auto_private_groups を **false** に設定すると、SSSD は、AD エントリーに設定された **primaryGroupID** を GID 番号として使用します。 **primaryGroupID** のデフォルト値は、AD の **Domain Users** グループに対応します。

表7.6 ID マッピング ID 範囲でauto_private_groups 変数が false に設定されている場合の SSSD の動作

AD のユーザー設定	id username の出力
AD ユーザーエントリーの内容 • SID が 7000 にマップされます。 • **primaryGroupID** は 8000 にマップされます。	**# id aduser@AD-DOMAIN.COMuid=7000(aduser@ad-domain.com) gid=8000(adgroup@ad-domain.com) groups=8000(adgroup@ad-domain.com), …**

関連情報

- AD ユーザー向けの POSIX および ID マッピング ID の範囲タイプ

7.9. CLI での POSIX ID 範囲の自動プライベートグループマッピングの有効化

デフォルトでは、SSSD は、AD に保存されている POSIX データに依存する POSIX 信頼を確立している場合は、Active Directory(AD) ユーザーのプライベートグループをマッピングしません。AD ユーザーにプライマリーグループが設定されていない場合、IdM はこれを解決できません。

この手順では、コマンドラインで **auto_private_groups** SSSD パラメーターに **hybrid** オプションを設定して、ID 範囲の自動プライベートグループマッピングを有効にする方法を説明します。これにより、IdM は、AD にプライマリーグループが設定されていない AD ユーザーを解決できます。

前提条件

- IdM 環境と AD 環境との間で、POSIX フォレスト間の信頼が正常に確立されました。

手順

1. すべての ID 範囲を表示し、変更する AD ID 範囲を書き留めます。

```
[root@server ~]# ipa idrange-find
----------------
2 ranges matched
----------------
  Range name: IDM.EXAMPLE.COM_id_range
  First Posix ID of the range: 882200000
  Number of IDs in the range: 200000
  Range type: local domain range

  Range name: AD.EXAMPLE.COM_id_range
  First Posix ID of the range: 1337000000
  Number of IDs in the range: 200000
  Domain SID of the trusted domain: S-1-5-21-4123312420-990666102-3578675309
  Range type: Active Directory trust range with POSIX attributes
----------------------------
Number of entries returned 2
----------------------------
```

2. **ipa idrange-mod** コマンドを使用して、AD ID 範囲の自動プライベートグループの動作を調整します。

```
[root@server ~]# ipa idrange-mod --auto-private-groups=hybrid
AD.EXAMPLE.COM_id_range
```

3. SSSD キャッシュをリセットして、新しい設定を有効にします。

```
[root@server ~]# sss_cache -E
```

関連情報

- Options for automatically mapping private groups for AD users

7.10. IDM WEBUI での POSIX ID 範囲の自動プライベートグループマッピングの有効化

デフォルトでは、SSSD は、AD に保存されている POSIX データに依存する POSIX 信頼を確立している場合は、Active Directory(AD) ユーザーのプライベートグループをマッピングしません。AD ユーザーにプライマリーグループが設定されていない場合、IdM はこれを解決できません。

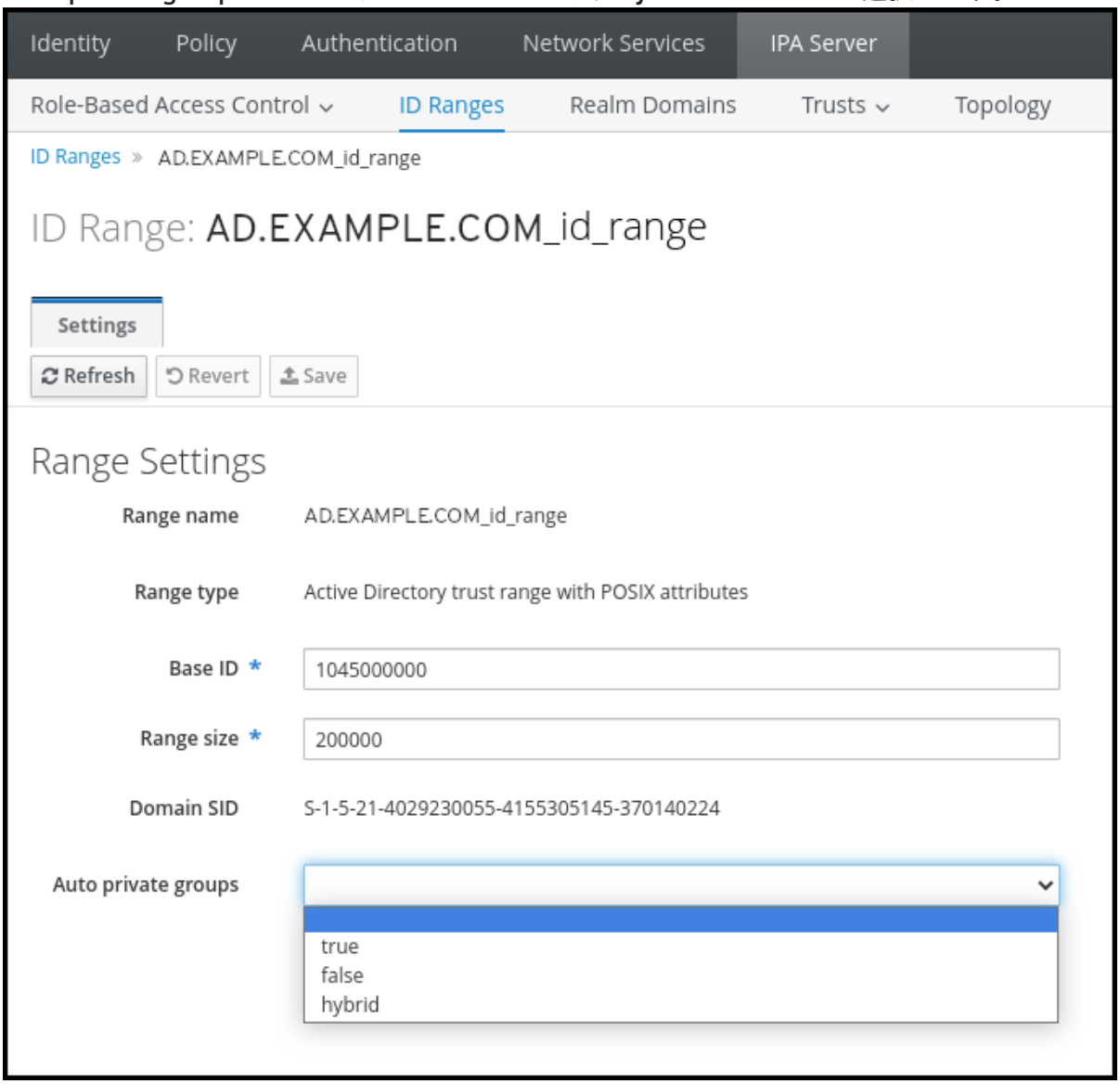
この手順では、Identity Management(IdM)WebUI の **auto_private_groups** SSSD パラメーター の **hybrid** オプションを設定して、ID 範囲の自動プライベートグループマッピングを有効にする方法を説明します。これにより、IdM は、AD にプライマリーグループが設定されていない AD ユーザーを解決できます。

前提条件

- IdM 環境と AD 環境との間で、POSIX フォレスト間の信頼が正常に確立されました。

手順

1. ユーザー名とパスワードを使用して IdM Web UI にログインします。
2. **IPA Server → ID Ranges** タブを開きます。
3. **AD.EXAMPLE.COM_id_range** など、変更する ID 範囲を選択します。
4. **Auto private groups** ドロップダウンメニューから、**hybrid** オプションを選択します。



5. **Save** ボタンをクリックして変更を保存します。

関連情報

- Options for automatically mapping private groups for AD users

7.11. 非 POSIX 外部グループと SID マッピング

Identity Management (IdM) は、グループ管理に LDAP を使用します。Active Directory (AD) エントリーは、IdM に同期またはコピーされません。つまり、AD ユーザーおよびグループには、LDAP サーバーに LDAP オブジェクトがないため、IdM LDAP のグループメンバーシップを表現するのにこのエントリーを直接使用することができません。このため、IdM の管理者は、非 POSIX 外部グループを作成する必要があります。これは、通常の IdM の LDAP オブジェクトで、IdM の中で AD ユーザーおよびグループが IdM のグループに所属していることを表現するのに使われます。

非 POSIX の外部グループのセキュリティー ID (SID) は SSSD により処理され、Active Directory のグループの SID を、IdM の POSIX グループにマップします。Active Directory では、SID はユーザー名に関連付けられています。AD のユーザー名を使用して IdM リソースにアクセスする場合、SSSD はユーザーの SID を使用して、IdM ドメイン内のユーザーの完全なグループメンバーシップ情報を構築します。

7.12. IDM-AD 信頼に DNS を設定するためのガイドライン

このガイドラインは、Identity Management (IdM) と Active Directory (AD) との間でフォレスト間の信頼を確立するために正しい DNS 設定を実現するのに役に立ちます。

一意のプライマリー DNS ドメイン

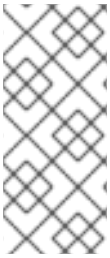
AD と IdM の両方に、独自の一意のプライマリー DNS ドメインが設定されているようにします。以下に例を示します。

- **ad.example.com** (AD の場合) および **idm.example.com** (IdM の場合)
- **example.com** (AD の場合) および **idm.example.com** (IdM の場合)

最も便利な管理ソリューションは、各 DNS ドメインが統合 DNS サーバーで管理されている環境ですが、規格に準拠した DNS サーバーも使用できます。

IdM ドメインおよび AD DNS ドメイン

IdM に参加しているシステムは、複数の DNS ドメインに分散できます。Red Hat では、Active Directory が所有するクライアントとは異なる DNS ゾーンに IdM クライアントをデプロイすることを推奨しています。プライマリー IdM DNS ドメインには、AD 信頼に対応するのに適切な SRV レコードが必要です。



注記

IdM と Active Directory との間の信頼がある一部の環境では、Active Directory DNS ドメインの一部であるホストに IdM クライアントをインストールできます。ホストは、これにより、Linux に焦点を合わせた IdM の機能の恩恵を受けることができます。これは推奨される設定ではなく、いくつかの制限があります。詳細は Active Directory DNS ドメインで IdM クライアントの設定 を参照してください。

適切な SRV レコード

プライマリー IdM DNS ドメインに、AD 信頼に対応するのに適切な SRV レコードがあることを確認します。

同じ IdM レルムにあるその他の DNS ドメインでは、AD への信頼設定時に SRV レコードを設定する必要がありません。これは、AD ドメインコントローラーが、Kerberos の鍵配布センター (KDC) の検索に SRV レコードを使用せず、信頼の名前接尾辞のルーティング情報を使用するためです。

DNS レコードが信頼内の全 DNS ドメインから解決可能である

すべてのマシンが、信頼関係内で関連するすべての DNS ドメインの DNS レコードを解決できるようにする必要があります。

- IdM DNS を設定する場合は Identity Management サーバーのインストール: 統合 DNS と外部 CA の場合 を参照してください。
- 統合 DNS を使用しない IdM を使用している場合は Identity Management サーバーのインストール: 統合 DNS がなく統合 CA がある場合 の手順を参照してください。

Kerberos レルム名は、プライマリー DNS ドメイン名を大文字にしたもの

Kerberos レルム名は、プライマリー DNS ドメイン名と同じで、すべて大文字になります。たとえば、AD のドメイン名が **ad.example.com** で、Identity Management のドメイン名が **idm.example.com** の場合、Kerberos レルム名は **AD.EXAMPLE.COM** および **IDM.EXAMPLE.COM** となります。

7.13. NETBIOS 名を設定するためのガイドライン

NetBIOS 名は通常、ドメイン名の一番左の部分です。以下に例を示します。

- ドメイン名 **linux.example.com** の NetBIOS 名は **linux** です。
- ドメイン名 **example.com** の NetBIOS 名は **example** です。

Identity Management (IdM) ドメインと Active Directory (AD) ドメインで異なる NetBIOS 名

IdM ドメインと AD ドメインが異なる NetBIOS 名を持つようにします。
AD ドメインの特定には NetBIOS 名が非常に重要になります。IdM ドメインが AD DNS のサブドメイン内にある場合、IdM ドメインおよびサービスの特定に NetBIOS 名も重要になります。

NetBIOS 名の文字制限

NetBIOS 名は最長 15 文字です。

7.14. サポート対象の WINDOWS SERVER バージョン

以下のフォレストおよびドメイン機能レベルを使用する Active Directory (AD) フォレストとの信頼関係を確立できます。

- フォレスト機能レベルの範囲 - Windows Server 2012 ~ Windows Server 2016
- ドメイン機能レベルの範囲: Windows Server 2012 - Windows Server 2016

Identity Management (IdM) は、以下のオペレーティングシステムを実行している Active Directory ドメインコントローラーとの信頼の確立に対応しています。

- Windows Server 2022 (RHEL 9.1 以降)
- Windows Server 2019
- Windows Server 2016

- Windows Server 2012 R2
- Windows Server 2012

重要

Identity Management (IdM) は、Windows Server 2008 R2 以前のバージョンを実行している Active Directory ドメインコントローラーとの間で Active Directory への信頼を確立することに対応していません。RHEL IdM との信頼関係を確立する際に、SMB 暗号化が必要です。これは、Windows Server 2012 以降でのみ対応しています。

7.15. AD サーバーの検出とアフィニティー

サーバー検出とアフィニティー設定は、IdM と AD 間のフォレスト間信頼において Identity Management (IdM) クライアントがどの Active Directory (AD) サーバーと通信するかに影響します。

地理的に同じ場所にあるサーバーを優先するようにクライアントを設定すると、クライアントが別のリモートデータセンターからサーバーにアクセスするときに発生するタイムラグなどの問題を防ぐことができます。クライアントがローカルサーバーと通信していることを確認するには、次のことを確認する必要があります。

- クライアントが、LDAP および Kerberos を介して、ローカルの IdM サーバーと通信している。
- クライアントが、Kerberos を介してローカルの AD サーバーと通信している。
- IdM サーバーの組み込みクライアントが、LDAP および Kerberos を介して、ローカルの AD サーバーと通信している。

ローカルの IdM サーバーと通信するために、IdM クライアントで LDAP と Kerberos を設定するためのオプション

統合 DNS を使用して IdM を使用する場合

デフォルトでは、クライアントは DNS レコードに基づいて自動サービスルックアップを使用します。この設定では、**DNS の場所** 機能を使用して、DNS ベースのサービス検出を設定することもできます。
自動検索を無効にするには、以下の方法で DNS 検出を無効にします。

- IdM クライアントのインストール中に、コマンドラインからフェイルオーバーのパラメーターを指定
- クライアントをインストールした後に、System Security Services Daemon (SSSD) の設定を変更

統合 DNS を使用せずに IdM を使用する場合

次のいずれかの方法で、クライアントを明示的に設定する必要があります。

- IdM クライアントのインストール中に、コマンドラインからフェイルオーバーのパラメーターを指定
- クライアントをインストールした後、SSSD の設定を変更

ローカルの AD サーバーと通信するために、IdM クライアントで Kerberos を設定するためのオプション

IdM クライアントは、どの AD サーバーと通信するかを自動的に検出できません。AD サーバーを手動で指定するには、**krb5.conf** ファイルを変更します。

- AD レルム情報を追加します。
- 以下を使用して、通信する AD サーバーを明示的に指定します。

以下に例を示します。

```
[realms]
AD.EXAMPLE.COM = {
kdc = server1.ad.example.com
kdc = server2.ad.example.com
}
```

Kerberos および LDAP を介したローカルの AD サーバーとの通信用に、IdM サーバーで組み込みクライアントを設定するためのオプション

IdM サーバーの組み込みクライアントは、AD サーバーのクライアントとしても機能します。適切な AD サイトを自動的に検出して使用できます。

組み込みクライアントが検出を実行すると、リモートの場所にある AD サーバーを最初に検出する可能性があります。リモートサーバーへの接続試行に時間がかかりすぎると、クライアントは接続を確立せずに操作を停止することがあります。クライアント上の **sssd.conf** ファイルの **dns_resolver_timeout** オプションを使用して、クライアントが DNS リゾルバーからの応答を待つ時間を長くします。詳細は man ページの **sssd.conf(5)** を参照してください。

埋め込みクライアントがローカルの AD サーバーと通信するように設定すると、SSSD は、組み込みクライアントが属する AD サイトを覚えます。そのため、SSSD は通常、ローカルドメインコントローラーに直接 LDAP ping を送信して、そのサイト情報を更新します。そのサイトが存在しなくなったか、クライアントが別のサイトに割り当てられた場合は、SSSD がフォレスト内の SRV レコードのクエリーを開始し、自動検出の全プロセスを実行します。

sssd.conf の **信頼されるドメインセクション** を使用して、デフォルトで自動的に検出される情報の一部を明示的に上書きすることもできます。

7.16. IDM と AD への間接統合中に実行する操作

次の操作とリクエストは、IdM から AD への間接的統合中に実行されます。

表を読んで、IdM トラストコントローラーから AD ドメインコントローラーへの Identity Management (IdM) から Active Directory (AD) への信頼の作成中に実行される操作と要求について学習します。

表7.7 IdM 信頼コントローラーから AD ドメインコントローラーへの操作

操作	使用プロトコル	目的
IdM 信頼コントローラーに設定された AD の DNS リゾルバーに対する DNS 解決	DNS	AD ドメインコントローラーの IP アドレスを検出する
AD DC における UDP/UDP6 ポート 389 へのリクエスト	非コネクション型 LDAP (CLDAP)	AD DC 検出を実行する

操作	使用プロトコル	目的
AD DC における TCP/TCP6 ポート 389 および 3268 へのリクエスト	LDAP	AD ユーザーおよびグループの情報をクエリーする
AD DC における TCP/TCP6 ポート 389 および 3268 へのリクエスト	DCE RPC および SMB	AD にフォレスト間の信頼を設定およびサポートする
AD DC における TCP/TCP6 ポート 135、139、および 445 へのリクエスト	DCE RPC および SMB	AD にフォレスト間の信頼を設定およびサポートする
Active Directory ドメインコントローラーの指示に従って、AD DC で動的に開かれたポートへのリクエスト (おそらく 49152 ~ 65535 (TCP/TCP6) の範囲)	DCE RPC および SMB	DCE RPC エンドポイントマッパー (ポート 135 TCP/TCP6) によるリクエストに応答する
AD DC におけるポート 88 (TCP/TCP6 および UDP/UDP6)、464 (TCP/TCP6 および UDP/UDP6)、749 (TCP/TCP6) へのリクエスト	Kerberos	Kerberos チケットの取得。Kerberos パスワードの変更。Kerberos をリモートで管理。

表を読んで、AD ドメインコントローラーから IdM 信頼コントローラーへの IdM から AD への信頼の作成中に実行される操作と要求について学習します。

表7.8 AD ドメインコントローラーから IdM 信頼コントローラーへの操作

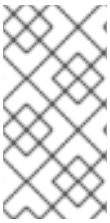
操作	使用プロトコル	目的
AD ドメインコントローラーに設定された IdM の DNS リゾルバーに対する DNS 解決	DNS	IdM 信頼コントローラーの IP アドレスを検出する
IdM 信頼コントローラーにおける UDP/UDP6 ポート 389 へのリクエスト	CLDAP	IdM 信頼コントローラー検出を実行する
IdM 信頼コントローラーにおける TCP/TCP6 ポート 135、139、445 へのリクエスト	DCE RPC および SMB	AD へのフォレスト間の信頼を確認する
IdM 信頼コントローラーの指示に従い、IdM 信頼コントローラー上で動的に開いたポートへのリクエスト (範囲はおそらく 49152 ~ 65535 (TCP/TCP6))	DCE RPC および SMB	DCE RPC エンドポイントマッパー (ポート 135 TCP/TCP6) によるリクエストに応答する

操作	使用プロトコル	目的
IdM 信頼コントローラーにおけるポート 88 (TCP/TCP6 および UDP/UDP6)、464 (TCP/TCP6 および UDP/UDP6)、および 749 (TCP/TCP6) へのリクエスト	Kerberos	Kerberos チケットの取得。Kerberos パスワードの変更。Kerberos をリモートで管理。

第8章 IDM のバックアップおよび復元

Identity Management を使用すると、データ損失イベントが発生した後に IdM システムを手動でバックアップおよび復元できます。

バックアップ中、システムが IdM セットアップに関する情報を保存するディレクトリーを作成します。このバックアップディレクトリーを使用して、元の IdM 設定を復元できます。



注記

IdM のバックアップ機能および復元機能は、データ損失を防止するように設計されています。サーバーの損失による影響を軽減し、継続的な運用を実現するために、クライアントに代替サーバーを用意してください。レプリケーショントポロジーの確立については、レプリケーションによるサーバーの損失への準備 を参照してください。

8.1. IDM バックアップの種類

ipa-backup ユーティリティーを使用すると、2 種類のバックアップを作成できます。

サーバーのフルバックアップ

- IdM に関連するすべてのサーバー設定ファイルと、LDAP データ交換形式 (LDIF) ファイルにある LDAP データがすべて **含まれます**。
- IdM サービスは **オフライン** である必要があります。
- IdM デプロイメントをゼロから再構築する場合に **適しています**。

データのみのバックアップ

- LDIF ファイルの LDAP データとレプリケーション変更ログが **含まれます**。
- IdM サービスは、**オンラインまたはオフライン** にできます。
- IdM データを以前の状態に復元する場合に **適しています**。

8.2. IDM バックアップファイルの命名規則

デフォルトでは、IdM はバックアップを **.tar** アーカイブとして **/var/lib/ipa/backup/** ディレクトリーのサブディレクトリーに保存します。

アーカイブおよびサブディレクトリーは、以下の命名規則に従います。

サーバーのフルバックアップ

ipa-full-<YEAR-MM-DD-HH-MM-SS> という名前のディレクトリーにある **ipa-full.tar** という名称のアーカイブ。時間は GMT 時間で指定されます。

```
[root@server ~]# ll /var/lib/ipa/backup/ipa-full-2021-01-29-12-11-46
total 3056
-rw-r--r--. 1 root root     158 Jan 29 12:11 header
-rw-r--r--. 1 root root 3121511 Jan 29 12:11 ipa-full.tar
```

データのみのバックアップ

ipa-data-<YEAR-MM-DD-HH-MM-SS> という名前のディレクトリーにある **ipa-data.tar** という名称のアーカイブ。時間は GMT 時間で指定されます。

```
[root@server ~]# ll /var/lib/ipa/backup/ipa-data-2021-01-29-12-14-23
total 1072
-rw-r--r--. 1 root root     158 Jan 29 12:14 header
-rw-r--r--. 1 root root 1090388 Jan 29 12:14 ipa-data.tar
```

注記

IdM サーバーをアンインストールしても、バックアップファイルは自動的に削除されません。

8.3. バックアップの作成時の考慮事項

ipa-backup コマンドの重要な動作と制限事項は次のとおりです。

- デフォルトでは、**ipa-backup** ユーティリティーはオフラインモードで実行されるため、IdM サービスがすべて停止します。このユーティリティーは、バックアップ完了後に IdM サービスを自動的に再起動します。
- サーバーのフルバックアップは、**常に** IdM サービスがオフラインの状態で実行する必要がありますが、データのみのバックアップは、サービスがオンラインの状態でも実行できます。
- デフォルトでは、**ipa-backup** ユーティリティーは、**/var/lib/ipa/backup/** ディレクトリーを含むファイルシステムにバックアップを作成します。Red Hat は、IdM が使用する実稼働ファイルシステムとは別のファイルシステムでバックアップを定期的に作成し、バックアップを固定メディア (例: テープまたは光学ストレージ) にアーカイブすることを推奨します。
- 非表示のレプリカ でのバックアップの実行を検討してください。IdM サービスは、非表示のレプリカでは、IdM クライアントに影響を及ぼさずにシャットダウンできます。
- **ipa-backup** ユーティリティーは、認証局 (CA)、ドメインネームシステム (DNS)、およびキー回復エージェント (KRA) など、IdM クラスターで使用されるすべてのサービスが、バックアップを実行中のサーバーにインストールされているかどうかを確認します。サーバーにこれらのサービスがすべてインストールされていない場合、そのホスト上で取得したバックアップではクラスターを完全に復元するには不十分なため、**ipa-backup** ユーティリティーは警告を表示して終了します。
  たとえば、IdM デプロイメントで統合認証局 (CA) を使用している場合、CA のないレプリカでバックアップを実行しても、CA データは取得されません。Red Hat は、**ipa-backup** を実行するレプリカに、クラスターで使用される IdM サービスがすべてインストールされていることを確認することを推奨します。

  ipa-backup --disable-role-check コマンドを使用すると、IdM サーバーのロールチェックを省略できます。ただし、生成されるバックアップに、IdM を完全に復元するのに必要な全データが保存されなくなります。

8.4. IDM バックアップの作成

ipa-backup コマンドを使用して、オフラインモードとオンラインモードで、完全なサーバーバックアップとデータのみのバックアップを作成します。

前提条件

- **ipa-backup** ユーティリティーを実行するには、**root** 権限が必要です。

手順

- オフラインモードでサーバーのフルバックアップを作成するには、追加オプションを指定せずに **ipa-backup** ユーティリティーを使用します。

```
[root@server ~]# ipa-backup
Preparing backup on server.example.com
Stopping IPA services
Backing up ipaca in EXAMPLE-COM to LDIF
Backing up userRoot in EXAMPLE-COM to LDIF
Backing up EXAMPLE-COM
Backing up files
Starting IPA service
Backed up to /var/lib/ipa/backup/ipa-full-2020-01-14-11-26-06
The ipa-backup command was successful
```

- オフラインでデータのみのバックアップを作成するには、**--data** オプションを指定します。

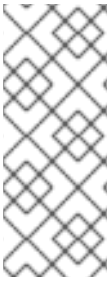
```
[root@server ~]# ipa-backup --data
```

- IdM ログファイルを含むサーバーのフルバックアップを作成するには、**--logs** オプションを使用します。

```
[root@server ~]# ipa-backup --logs
```

- IdM サービスの実行中にデータのみのバックアップを作成するには、**--data** オプションおよび **--online** オプションの両方を指定します。

```
[root@server ~]# ipa-backup --data --online
```



注記

/tmp ディレクトリーに十分なスペースがないためにバックアップが失敗する場合は、**TMPDIR** 環境変数を使用して、バックアッププロセスで作成された一時ファイルの保存先を変更します。

```
[root@server ~]# TMPDIR=/new/location ipa-backup
```

検証手順

- バックアップディレクトリーにバックアップを含むアーカイブが含まれていることを確認します。

```
[root@server ~]# ls /var/lib/ipa/backup/ipa-full-2020-01-14-11-26-06
header  ipa-full.tar
```

関連情報

- ipa-backup コマンドの終了に失敗する

8.5. GPG2 で暗号化した IDM バックアップの作成

GPG (GNU Privacy Guard) 暗号化を使用して、暗号化バックアップを作成できます。以下の手順では、IdM バックアップを作成し、GPG2 キーを使用して暗号化します。

前提条件

- GPG2 キーを作成している。GPG2 キーの作成 を参照してください。

手順

- **--gpg** オプションを指定して、GPG で暗号化したバックアップを作成します。

```
[root@server ~]# ipa-backup --gpg
Preparing backup on server.example.com
Stopping IPA services
Backing up ipaca in EXAMPLE-COM to LDIF
Backing up userRoot in EXAMPLE-COM to LDIF
Backing up EXAMPLE-COM
Backing up files
Starting IPA service
Encrypting /var/lib/ipa/backup/ipa-full-2020-01-13-14-38-00/ipa-full.tar
Backed up to /var/lib/ipa/backup/ipa-full-2020-01-13-14-38-00
The ipa-backup command was successful
```

検証手順

- バックアップディレクトリーに、ファイル拡張子が **.gpg** の暗号化されたアーカイブが含まれていることを確認します。

```
[root@server ~]# ls /var/lib/ipa/backup/ipa-full-2020-01-13-14-38-00
header  ipa-full.tar.gpg
```

関連情報

- IdM バックアップの作成

8.6. GPG2 キーの作成

以下の手順では、暗号化ユーティリティーで使用する GPG2 キーを生成する方法を説明します。

前提条件

- **root** 権限がある。

手順

1. **pinentry** ユーティリティーをインストールして設定します。

```
[root@server ~]# dnf install pinentry
[root@server ~]# mkdir ~/.gnupg -m 700
[root@server ~]# echo "pinentry-program /usr/bin/pinentry-curses" >> ~/.gnupg/gpg-agent.conf
```

2. 希望する内容で、GPG キーペアの生成に使用する **key-input** ファイルを作成します。以下に例を示します。

```
[root@server ~]# cat >key-input <<EOF
%echo Generating a standard key
Key-Type: RSA
Key-Length: 2048
Name-Real: GPG User
Name-Comment: first key
Name-Email: root@example.com
Expire-Date: 0
%commit
%echo Finished creating standard key
EOF
```

3. **(オプション)** デフォルトでは、GPG2 はキーリングを **~/.gnupg** ファイルに保存します。カスタムのキーリングの場所を使用するには、**GNUPGHOME** 環境変数を、root のみがアクセスできるディレクトリーに設定します。

```
[root@server ~]# export GNUPGHOME=/root/backup

[root@server ~]# mkdir -p $GNUPGHOME -m 700
```

4. **key-input** ファイルの内容に基づいて、新しい GPG2 キーを生成します。

```
[root@server ~]# gpg2 --batch --gen-key key-input
```

5. GPG2 キーを保護するパスフレーズを入力します。このパスフレーズは、秘密鍵へのアクセスと復号に使用します。

```
┌──────────────────────────────────────────────────────┐
│ Please enter the passphrase to                       │
│ protect your new key                                 │
│                                                      │
│ Passphrase: <passphrase>                             │
│                                                      │
│  <OK>                    <Cancel>                    │
└──────────────────────────────────────────────────────┘
```

6. パスフレーズを再度入力して、正しいパスフレーズを確認します。

```
┌──────────────────────────────────────────────────────┐
│ Please re-enter this passphrase                      │
│                                                      │
│ Passphrase: <passphrase>                             │
│                                                      │
│  <OK>                    <Cancel>                    │
└──────────────────────────────────────────────────────┘
```

7. 新しい GPG2 キーが正常に作成されたことを確認します。

```
gpg: keybox '/root/backup/pubring.kbx' created
gpg: Generating a standard key
gpg: /root/backup/trustdb.gpg: trustdb created
```

```
gpg: key BF28FFA302EF4557 marked as ultimately trusted
gpg: directory '/root/backup/openpgp-revocs.d' created
gpg: revocation certificate stored as '/root/backup/openpgp-
revocs.d/8F6FCF10C80359D5A05AED67BF28FFA302EF4557.rev'
gpg: Finished creating standard key
```

検証手順

- サーバーの GPG キーのリストを表示します。

```
[root@server ~]# gpg2 --list-secret-keys
gpg: checking the trustdb
gpg: marginals needed: 3  completes needed: 1  trust model: pgp
gpg: depth: 0  valid:   1  signed:   0  trust: 0-, 0q, 0n, 0m, 0f, 1u
/root/backup/pubring.kbx
------------------------
sec   rsa2048 2020-01-13 [SCEA]
      8F6FCF10C80359D5A05AED67BF28FFA302EF4557
uid           [ultimate] GPG User (first key) <root@example.com>
```

関連情報

- GNU Privacy Guard

8.7. IDM バックアップから復元するタイミング

IdM バックアップから復元すると、いくつかの障害シナリオに対応できます。

- **LDAP コンテンツに望ましくない変更が加えられた** - エントリーは変更または削除され、デプロイメント全体でそれらの変更が行われ、これらの変更を元に戻すようにします。データのみのバックアップを復元すると、IdM 設定自体に影響を与えずに LDAP エントリーが以前の状態に戻ります。

- **インフラストラクチャーの損失の合計、またはすべての CA インスタンスの損失** - 障害によりすべての認証局レプリカが損傷した場合、デプロイメントは追加のサーバーをデプロイすることで、それ自体を再構築する機能を失うようになりました。この場合は、CA レプリカのバックアップを復元し、そこから新しいレプリカを構築します。

- **分離されたサーバーのアップグレードに失敗** - オペレーティングシステムは機能し続けますが、IdM データが破損するため、IdM システムを既知の正常な状態に復元したい理由になります。Red Hat では、問題を診断してトラブルシューティングするために、テクニカルサポートの利用を推奨しています。以上の作業にすべて失敗した場合は、サーバーのフルバックアップから復元します。

重要

ハードウェアまたはアップグレードの失敗で推奨されるソリューションは、失われたサーバーをレプリカから再構築することです。詳細は、レプリケーションを使用した 1 台のサーバーの復旧 を参照してください。

8.8. IDM バックアップから復元する際の注意点

ipa-backup ユーティリティーでバックアップを作成した場合は、IdM サーバーまたは LDAP コンテンツをバックアップ実行時の状態に復元できます。

以下は、IdM バックアップからの復元時の主要な考慮事項です。

- バックアップの作成元のサーバーの設定と一致するサーバー上でのみバックアップを復元できます。サーバーには以下の項目が **必要** です。
  - 同じホスト名
  - 同じ IP アドレス
  - 同じバージョンの IdM ソフトウェア
- 多数サーバーがある中で IdM サーバーを復元すると、復元されたサーバーは、IdM の唯一の情報ソースになります。他のサーバーはすべて、復元されたサーバーをもとに再度初期化する**必要があります**。
- 最後のバックアップ後に作成されたデータはすべて失われるため、通常のシステムメンテナンスには、バックアップと復元のソリューションを使用しないでください。
- サーバーが失われた場合は、バックアップから復元するのではなく、レプリカとしてサーバーを再インストールしてサーバーを再構築することが推奨されます。新規レプリカを作成すると、現在の作業環境のデータが保存されます。詳細は、サーバーでのレプリケーションによる損失の準備 を参照してください。
- バックアップ機能および復元機能はコマンドラインからのみ管理でき、IdM Web UI では使用できません。
- **/tmp** または **/var/tmp** ディレクトリーにあるバックアップファイルからは復元できません。IdM Directory Server は **PrivateTmp** ディレクトリーを使用しており、オペレーティングシステムで一般的に利用できる **/tmp** または **/var/tmp** ディレクトリーにはアクセスできません。

ヒント

バックアップから復元するには、バックアップの実行時にインストールされたものと同じバージョンのソフトウェア (RPM) がターゲットホストに必要になります。このため、Red Hat は、バックアップではなく、仮想マシンのスナップショットからの復元を行うことを推奨します。詳細は 仮想マシンスナップショットによるデータ損失からの復旧 を参照してください。

8.9. バックアップからの IDM サーバーの復元

IdM バックアップから IdM サーバーまたはその LDAP データを復元します。

図8.1 この例で使用されるレプリケーショントポロジー

表8.1 この例で使用されるサーバーの命名規則

サーバーのホスト名	機能
server1.example.com	バックアップから復元する必要があるサーバー

サーバーのホスト名	機能
caReplica2.example.com	**server1.example.com** ホストに接続した認証局 (CA) レプリカ。
replica3.example.com	**caReplica2.example.com** ホストに接続しているレプリカ。

前提条件

- **ipa-backup** ユーティリティーを使用して IdM サーバー全体のバックアップまたはデータのみのバックアップを生成している。See IdM バックアップの作成.
- バックアップファイルが **/tmp** または **/var/tmp** ディレクトリーにない。
- 完全なサーバーバックアップからサーバーの完全な復元を実行する前に、サーバーから IdM をアンインストール し、以前と同じサーバー設定を使用して IdM を 再インストール します。

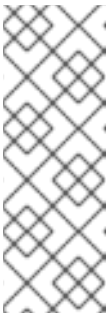
手順

1. **Ipa-restore** ユーティリティーを使用して、完全なサーバーまたはデータのみのバックアップを復元します。
   - バックアップディレクトリーがデフォルトの **/var/lib/ipa/backup/** の場合は、ディレクトリーの名前のみを入力します。

     ```
     [root@server1 ~]# ipa-restore ipa-full-2020-01-14-12-02-32
     ```

   - バックアップディレクトリーがデフォルトの場所にない場合は、完全パスを入力します。

     ```
     [root@server1 ~]# ipa-restore /mybackups/ipa-data-2020-02-01-05-30-00
     ```

     注記

     Ipa-restore ユーティリティーは、ディレクトリーに含まれるバックアップのタイプを自動的に検出し、デフォルトで同じタイプの復元を実行します。完全なサーバーバックアップからデータのみの復元を実行するには、**--data** オプションを **ipa-restore** コマンドに追加します。

     ```
     [root@server1 ~]# ipa-restore --data ipa-full-2020-01-14-12-02-32
     ```

2. Directory Manager パスワードを入力します。

   ```
   Directory Manager (existing master) password:
   ```

3. **Yes** を入力して、現在のデータをバックアップで上書きしていることを確認します。

   ```
   Preparing restore from /var/lib/ipa/backup/ipa-full-2020-01-14-12-02-32 on
   server1.example.com
   Performing FULL restore from FULL backup
   ```

```
Temporary setting umask to 022
Restoring data will overwrite existing live data. Continue to restore? [no]: yes
```

4. **Ipa-restore** ユーティリティーは、利用可能なすべてのサーバーでレプリケーションを無効にします。

```
Each master will individually need to be re-initialized or
re-created from this one. The replication agreements on
masters running IPA 3.1 or earlier will need to be manually
re-enabled. See the man page for details.
Disabling all replication.
Disabling replication agreement on server1.example.com to caReplica2.example.com
Disabling CA replication agreement on server1.example.com to caReplica2.example.com
Disabling replication agreement on caReplica2.example.com to server1.example.com
Disabling replication agreement on caReplica2.example.com to replica3.example.com
Disabling CA replication agreement on caReplica2.example.com to server1.example.com
Disabling replication agreement on replica3.example.com to caReplica2.example.com
```

その後、このユーティリティーは IdM サービスを停止し、バックアップを復元し、サービスを再起動します。

```
Stopping IPA services
Systemwide CA database updated.
Restoring files
Systemwide CA database updated.
Restoring from userRoot in EXAMPLE-COM
Restoring from ipaca in EXAMPLE-COM
Restarting GSS-proxy
Starting IPA services
Restarting SSSD
Restarting oddjobd
Restoring umask to 18
The ipa-restore command was successful
```

5. 復元されたサーバーに接続したすべてのレプリカを再初期化します。

   a. **domain** 接尾辞のレプリカトポロジーセグメントのリストを表示します。復元されたサーバーに関連するトポロジーセグメントを書き留めます。

```
[root@server1 ~]# ipa topologysegment-find domain
------------------
2 segments matched
------------------
  Segment name: server1.example.com-to-caReplica2.example.com
  Left node: server1.example.com
  Right node: caReplica2.example.com
  Connectivity: both

  Segment name: caReplica2.example.com-to-replica3.example.com
  Left node: caReplica2.example.com
  Right node: replica3.example.com
  Connectivity: both
----------------------------
Number of entries returned 2
----------------------------
```

b. 復元されたサーバーとともにすべてのトポロジーセグメントの **domain** 接尾辞を再初期化します。
この例では、**server1** からのデータで **caReplica2** の再初期化を実行します。

```
[root@caReplica2 ~]# ipa-replica-manage re-initialize --from=server1.example.com
Update in progress, 2 seconds elapsed
Update succeeded
```

c. 認証局データに移動し、**ca** 接尾辞のレプリケーショントポロジーセグメントのリストを表示します。

```
[root@server1 ~]# ipa topologysegment-find ca
-----------------
1 segment matched
-----------------
  Segment name: server1.example.com-to-caReplica2.example.com
  Left node: server1.example.com
  Right node: caReplica2.example.com
  Connectivity: both
----------------------------
Number of entries returned 1
----------------------------
```

d. 復元されたサーバーに接続されているすべての CA レプリカを再初期化します。
この例では、**server1** からのデータを使用して **caReplica2** の **csreplica** を再初期化します。

```
[root@caReplica2 ~]# ipa-csreplica-manage re-initialize --
from=server1.example.com
Directory Manager password:

Update in progress, 3 seconds elapsed
Update succeeded
```

6. 復元されたサーバー **server1.example.com** のデータですべてのサーバーが更新されるまで、レプリケーショントポロジーを介して、後続のレプリカを再初期化します。
この例では、**caReplica2** からのデータで、**replica3** の **domain** 接尾辞を再初期化することのみが必要になります。

```
[root@replica3 ~]# ipa-replica-manage re-initialize --from=caReplica2.example.com
Directory Manager password:

Update in progress, 3 seconds elapsed
Update succeeded
```

7. すべてのサーバーで SSSD のキャッシュをクリアし、無効なデータによる認証の問題を回避します。

a. SSSD サービスを停止します。

```
[root@server ~]# systemctl stop sssd
```

b. SSSD からキャッシュされたコンテンツをすべて削除します。

```
[root@server ~]# sss_cache -E
```

c. SSSD サービスを起動します。

```
[root@server ~]# systemctl start sssd
```

d. サーバーを再起動します。

関連情報

- **ipa-restore**(1) の man ページでは、復元中の複雑なレプリケーションシナリオの処理方法が詳細に説明されています。

8.10. 暗号化されたバックアップからの復元

この手順では、暗号化された IdM バックアップから IdM サーバーを復元します。**Ipa-restore** ユーティリティーは、IdM バックアップが暗号化されているかどうかを自動的に検出し、GPG2 root キーリングを使用して復元します。

前提条件

- GPG 暗号化 IdM バックアップ。GPG2 で暗号化した IdM バックアップの作成 を参照してください。
- LDAP Directory Manager のパスワード
- GPG キーの作成時に使用されるパスフレーズ

手順

1. GPG2 キーの作成時にカスタムキーリングの場所を使用した場合は、**$GNUPGHOME** 環境変数がそのディレクトリーに設定されていることを確認します。GPG2 キーの作成 を参照してください。

```
[root@server ~]# echo $GNUPGHOME
/root/backup
```

2. **Ipa-restore** ユーティリティーにバックアップディレクトリーの場所を指定します。

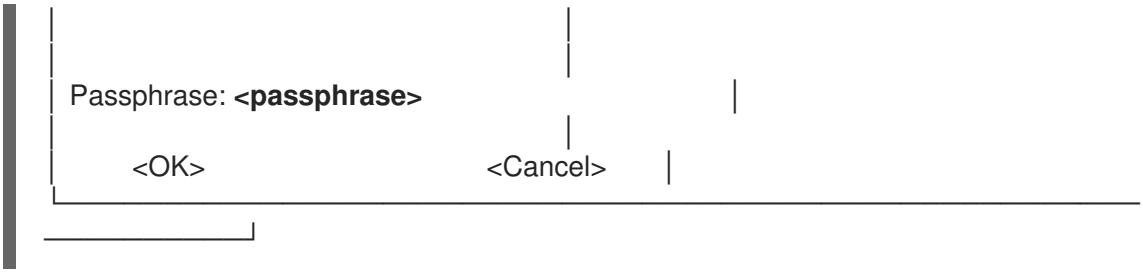
```
[root@server ~]# ipa-restore ipa-full-2020-01-13-18-30-54
```

a. Directory Manager パスワードを入力します。

```
Directory Manager (existing master) password:
```

b. GPG キーの作成時に使用したパスフレーズを入力します。

```
┌──────────────────────────────────────────────────────────────────────┐
│ Please enter the passphrase to unlock the OpenPGP secret key:        │
│ "GPG User (first key) <root@example.com>"                            │
│ 2048-bit RSA key, ID BF28FFA302EF4557,                               │
│ created 2020-01-13.                                                  │
```



3. 復元されたサーバーに接続されているすべてのレプリカを再初期化します。バックアップからの IdM サーバーの復元 を参照してください。

第9章 ANSIBLE PLAYBOOK を使用した IDM サーバーのバックアップおよび復元

ipabackup Ansible **ロール** を使用して、IdM サーバーのバックアップを自動化し、サーバーと Ansible コントローラー間でバックアップファイルを転送して、バックアップから IdM サーバーを復元できます。

9.1. ANSIBLE を使用した IDM サーバーのバックアップの作成

Ansible Playbook の **ipabackup** ロールを使用して、IdM サーバーのバックアップを作成し、それを IdM サーバーに保存できます。

前提条件

- 次の要件を満たすように Ansible コントロールノードを設定している。
  - Ansible バージョン 2.14 以降を使用している。
  - Ansible コントローラーに **ansible-freeipa** パッケージがインストールされている。
  - **~/MyPlaybooks/** ディレクトリーに、IdM サーバーの完全修飾ドメイン名 (FQDN) を使用して Ansible インベントリーファイル を作成している (この例の場合)。
  - この例では、**secret.yml** Ansible vault に **ipaadmin_password** が保存されていることを前提としています。
- ターゲットノード (**ansible-freeipa** モジュールが実行されるノード) が、IdM クライアント、サーバー、またはレプリカとして IdM ドメインに含まれている。

手順

1. **~/MyPlaybooks/** ディレクトリーに移動します。

```
$ cd ~/MyPlaybooks/
```

2. **/usr/share/doc/ansible-freeipa/playbooks** ディレクトリーにある **backup-server.yml** ファイルのコピーを作成します。

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/backup-server.yml backup-my-server.yml
```

3. **backup-my-server.yml** Ansible Playbook ファイルを開いて編集します。
4. **hosts** 変数をインベントリーファイルのホストグループに設定して、ファイルを調整します。この例では、**ipaserver** ホストグループに設定します。

```
---
- name: Playbook to backup IPA server
  hosts: ipaserver
  become: true

  roles:
  - role: ipabackup
    state: present
```

5. ファイルを保存します。
6. Playbook ファイルとインベントリーファイルを指定して Ansible Playbook を実行します。

```
$ ansible-playbook --vault-password-file=password_file -v -i ~/MyPlaybooks/inventory
backup-my-server.yml
```

検証手順

1. バックアップした IdM サーバーにログインします。
2. バックアップが **/var/lib/ipa/backup** ディレクトリーにあることを確認します。

```
[root@server ~]# ls /var/lib/ipa/backup/
ipa-full-2021-04-30-13-12-00
```

関連情報

- **ipabackup** ロールを使用する他の Ansible Playbook の例は、以下を参照してください。
  - **/usr/share/doc/ansible-freeipa/roles/ipabackup** ディレクトリーの **README.md** ファイル
  - **/usr/share/doc/ansible-freeipa/playbooks/** ディレクトリー

9.2. ANSIBLE を使用した ANSIBLE コントローラーへの IDM サーバーのバックアップの作成

Ansible Playbook の **ipabackup** ロールを使用して、IdM サーバーのバックアップを作成し、それを Ansible コントローラーに自動的に転送できます。バックアップファイル名は、IdM サーバーのホスト名で始まります。

前提条件

- 次の要件を満たすように Ansible コントロールノードを設定している。
  - Ansible バージョン 2.14 以降を使用している。
  - Ansible コントローラーに **ansible-freeipa** パッケージがインストールされている。
  - **~/MyPlaybooks/** ディレクトリーに、IdM サーバーの完全修飾ドメイン名 (FQDN) を使用して Ansible インベントリーファイル を作成している (この例の場合)。
  - この例では、**secret.yml** Ansible vault に **ipaadmin_password** が保存されていることを前提としています。
- ターゲットノード (**ansible-freeipa** モジュールが実行されるノード) が、IdM クライアント、サーバー、またはレプリカとして IdM ドメインに含まれている。

手順

1. バックアップを保存するために、Ansible コントローラーのホームディレクトリーにサブディレクトリーを作成します。

```
$ mkdir ~/ipabackups
```

2. ~/**MyPlaybooks**/ ディレクトリーに移動します。

```
$ cd ~/MyPlaybooks/
```

3. /**usr**/**share**/**doc**/**ansible-freeipa**/**playbooks** ディレクトリーにある **backup-server-to-controller.yml** ファイルのコピーを作成します。

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/backup-server-to-controller.yml backup-my-server-to-my-controller.yml
```

4. **backup-my-server-to-my-controller.yml** ファイルを開いて編集します。

5. 以下の変数を設定してファイルを調整します。

   a. **hosts** 変数を、インベントリーファイルのホストグループに設定します。この例では、**ipaserver** ホストグループに設定します。

   b. **(オプション)** IdM サーバー上にバックアップのコピーを保持するには、次の行のコメントを解除します。

```
# ipabackup_keep_on_server: true
```

6. デフォルトでは、バックアップは Ansible コントローラーの現在の作業ディレクトリーに保存されます。ステップ 1 で作成したバックアップディレクトリーを指定するには、**ipabackup_controller_path** 変数を追加し、それを /**home**/**user**/**ipabackups** ディレクトリーに設定します。

```
---
- name: Playbook to backup IPA server to controller
  hosts: ipaserver
  become: true
  vars:
    ipabackup_to_controller: true
    # ipabackup_keep_on_server: true
    ipabackup_controller_path: /home/user/ipabackups

  roles:
  - role: ipabackup
    state: present
```

7. ファイルを保存します。

8. Playbook ファイルとインベントリーファイルを指定して Ansible Playbook を実行します。

```
$ ansible-playbook --vault-password-file=password_file -v -i ~/MyPlaybooks/inventory backup-my-server-to-my-controller.yml
```

検証手順

- バックアップが Ansible コントローラーの /**home**/**user**/**ipabackups** ディレクトリーにあることを確認します。

```
[user@controller ~]$ ls /home/user/ipabackups
server.idm.example.com_ipa-full-2021-04-30-13-12-00
```

関連情報

- **ipabackup** ロールを使用する他の Ansible Playbook の例は、以下を参照してください。
  - **/usr/share/doc/ansible-freeipa/roles/ipabackup** ディレクトリーの **README.md** ファイル
  - **/usr/share/doc/ansible-freeipa/playbooks/** ディレクトリー

9.3. ANSIBLE を使用した IDM サーバーのバックアップの ANSIBLE コントローラーへのコピー

Ansible Playbook を使用して、IdM サーバーのバックアップを IdM サーバーから Ansible コントローラーにコピーできます。

前提条件

- 次の要件を満たすように Ansible コントロールノードを設定している。
  - Ansible バージョン 2.14 以降を使用している。
  - Ansible コントローラーに **ansible-freeipa** パッケージがインストールされている。
  - **~/MyPlaybooks/** ディレクトリーに、IdM サーバーの完全修飾ドメイン名 (FQDN) を使用して Ansible インベントリーファイル を作成している (この例の場合)。
  - この例では、**secret.yml** Ansible vault に **ipaadmin_password** が保存されていることを前提としています。
- ターゲットノード (**ansible-freeipa** モジュールが実行されるノード) が、IdM クライアント、サーバー、またはレプリカとして IdM ドメインに含まれている。

手順

1. バックアップを保存するために、Ansible コントローラーのホームディレクトリーにサブディレクトリーを作成します。

```
$ mkdir ~/ipabackups
```

2. ~/**MyPlaybooks/** ディレクトリーに移動します。

```
$ cd ~/MyPlaybooks/
```

3. **/usr/share/doc/ansible-freeipa/playbooks** ディレクトリーにある **copy-backup-from-server.yml** ファイルのコピーを作成します。

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/copy-backup-from-server.yml copy-backup-from-my-server-to-my-controller.yml
```

4. **copy-my-backup-from-my-server-to-my-controller.yml** ファイルを開いて編集します。

5. 以下の変数を設定してファイルを調整します。

   a. **hosts** 変数を、インベントリーファイルのホストグループに設定します。この例では、**ipaserver** ホストグループに設定します。

   b. **ipabackup_name** 変数を、Ansible コントローラーにコピーする IdM サーバー上の **ipabackup** の名前に設定します。

   c. デフォルトでは、バックアップは Ansible コントローラーの現在の作業ディレクトリーに保存されます。ステップ 1 で作成したディレクトリーを指定するには、**ipabackup_controller_path** 変数を追加し、それを **/home/user/ipabackups** ディレクトリーに設定します。

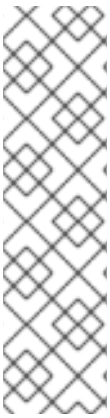
```
---
- name: Playbook to copy backup from IPA server
  hosts: ipaserver
  become: true
  vars:
    ipabackup_name: ipa-full-2021-04-30-13-12-00
    ipabackup_to_controller: true
    ipabackup_controller_path: /home/user/ipabackups

  roles:
  - role: ipabackup
    state: present
```

6. ファイルを保存します。

7. Playbook ファイルとインベントリーファイルを指定して Ansible Playbook を実行します。

```
$ ansible-playbook --vault-password-file=password_file -v -i ~/MyPlaybooks/inventory copy-backup-from-my-server-to-my-controller.yml
```



注記

すべて の IdM バックアップをコントローラーにコピーするには、Ansible Playbook の **ipabackup_name** 変数を **all** に設定します。

```
  vars:
    ipabackup_name: all
    ipabackup_to_controller: true
```

たとえば、**/usr/share/doc/ansible-freeipa/playbooks** ディレクトリーの Ansible Playbook **copy-all-backups-from-server.yml** を参照してください。

検証手順

- バックアップが Ansible コントローラーの **/home/user/ipabackups** ディレクトリーにあることを確認します。

```
[user@controller ~]$ ls /home/user/ipabackups
server.idm.example.com_ipa-full-2021-04-30-13-12-00
```

関連情報

- **/usr/share/doc/ansible-freeipa/roles/ipabackup** ディレクトリーの **README.md** ファイル
- **/usr/share/doc/ansible-freeipa/playbooks/** ディレクトリー

9.4. ANSIBLE を使用した IDM サーバーのバックアップの ANSIBLE コントローラーから IDM サーバーへのコピー

Ansible Playbook を使用して、IdM サーバーのバックアップを Ansible コントローラーから IdM サーバーにコピーできます。

前提条件

- 次の要件を満たすように Ansible コントロールノードを設定している。
  - Ansible バージョン 2.14 以降を使用している。
  - Ansible コントローラーに **ansible-freeipa** パッケージがインストールされている。
  - **~/MyPlaybooks/** ディレクトリーに、IdM サーバーの完全修飾ドメイン名 (FQDN) を使用して Ansible インベントリーファイル を作成している (この例の場合)。
  - この例では、**secret.yml** Ansible vault に **ipaadmin_password** が保存されていることを前提としています。
- ターゲットノード (**ansible-freeipa** モジュールが実行されるノード) が、IdM クライアント、サーバー、またはレプリカとして IdM ドメインに含まれている。

手順

1. **~/MyPlaybooks/** ディレクトリーに移動します。

```
$ cd ~/MyPlaybooks/
```

2. **/usr/share/doc/ansible-freeipa/playbooks** ディレクトリーにある **copy-backup-from-controller.yml** のコピーを作成します。

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/copy-backup-from-controller.yml copy-backup-from-my-controller-to-my-server.yml
```

3. **copy-my-backup-from-my-controller-to-my-server.yml** ファイルを開いて編集します。
4. 以下の変数を設定してファイルを調整します。
   a. **hosts** 変数を、インベントリーファイルのホストグループに設定します。この例では、**ipaserver** ホストグループに設定します。
   b. **ipabackup_name** 変数を、IdM サーバーにコピーする Ansible コントローラー上の **ipabackup** の名前に設定します。

```
---
- name: Playbook to copy a backup from controller to the IPA server
  hosts: ipaserver
  become: true

  vars:
```

```
  ipabackup_name: server.idm.example.com_ipa-full-2021-04-30-13-12-00
  ipabackup_from_controller: true

roles:
- role: ipabackup
  state: copied
```

5. ファイルを保存します。
6. Playbook ファイルとインベントリーファイルを指定して Ansible Playbook を実行します。

```
$ ansible-playbook --vault-password-file=password_file -v -i ~/MyPlaybooks/inventory copy-backup-from-my-controller-to-my-server.yml
```

関連情報

- **/usr/share/doc/ansible-freeipa/roles/ipabackup** ディレクトリーの **README.md** ファイル
- **/usr/share/doc/ansible-freeipa/playbooks/** ディレクトリー

9.5. ANSIBLE を使用した IDM サーバーからのバックアップの削除

Ansible Playbook を使用して、IdM サーバーからバックアップを削除できます。

前提条件

- 次の要件を満たすように Ansible コントロールノードを設定している。
  - Ansible バージョン 2.14 以降を使用している。
  - Ansible コントローラーに **ansible-freeipa** パッケージがインストールされている。
  - **~/MyPlaybooks/** ディレクトリーに、IdM サーバーの完全修飾ドメイン名 (FQDN) を使用して Ansible インベントリーファイル を作成している (この例の場合)。
  - この例では、**secret.yml** Ansible vault に **ipaadmin_password** が保存されていることを前提としています。
- ターゲットノード (**ansible-freeipa** モジュールが実行されるノード) が、IdM クライアント、サーバー、またはレプリカとして IdM ドメインに含まれている。

手順

1. ~/MyPlaybooks/ ディレクトリーに移動します。

```
$ cd ~/MyPlaybooks/
```

2. **/usr/share/doc/ansible-freeipa/playbooks** ディレクトリーにある **remove-backup-from-server.yml** ファイルのコピーを作成します。

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/remove-backup-from-server.yml remove-backup-from-my-server.yml
```

3. **remove-backup-from-my-server.yml** ファイルを開いて編集します。

4. 以下の変数を設定してファイルを調整します。

   a. **hosts** 変数を、インベントリーファイルのホストグループに設定します。この例では、**ipaserver** ホストグループに設定します。

   b. **ipabackup_name** 変数を、IdM サーバーから削除する **ipabackup** の名前に設定します。

```
---
- name: Playbook to remove backup from IPA server
  hosts: ipaserver
  become: true

  vars:
    ipabackup_name: ipa-full-2021-04-30-13-12-00

  roles:
  - role: ipabackup
    state: absent
```

5. ファイルを保存します。

6. Playbook ファイルとインベントリーファイルを指定して Ansible Playbook を実行します。

```
$ ansible-playbook --vault-password-file=password_file -v -i ~/MyPlaybooks/inventory
remove-backup-from-my-server.yml
```

注記

IdM サーバーから **すべて** の IdM バックアップを削除するには、Ansible Playbook の **ipabackup_name** 変数を **all** に設定します。

```
vars:
  ipabackup_name: all
```

たとえば、**/usr/share/doc/ansible-freeipa/playbooks** ディレクトリーの Ansible Playbook **remove-all-backups-from-server.yml** を参照してください。

関連情報

- **/usr/share/doc/ansible-freeipa/roles/ipabackup** ディレクトリーの **README.md** ファイル
- **/usr/share/doc/ansible-freeipa/playbooks/** ディレクトリー

9.6. ANSIBLE を使用したサーバーに保存されているバックアップからの IDM サーバーの復元

Ansible Playbook を使用して、サーバーのホストに保存されているバックアップから IdM サーバーを復元できます。

前提条件

- 次の要件を満たすように Ansible コントロールノードを設定している。
  - Ansible バージョン 2.14 以降を使用している。

- Ansible コントローラーに **ansible-freeipa** パッケージがインストールされている。
- **~/MyPlaybooks/** ディレクトリーに、IdM サーバーの完全修飾ドメイン名 (FQDN) を使用して Ansible インベントリーファイル を作成している (この例の場合)。
- この例では、**secret.yml** Ansible vault に **ipaadmin_password** が保存されていることを前提としています。

- ターゲットノード (**ansible-freeipa** モジュールが実行されるノード) が、IdM クライアント、サーバー、またはレプリカとして IdM ドメインに含まれている。
- LDAP Directory Manager のパスワードを知っている必要があります。

手順

1. **~/MyPlaybooks/** ディレクトリーに移動します。

```
$ cd ~/MyPlaybooks/
```

2. **/usr/share/doc/ansible-freeipa/playbooks** ディレクトリーにある **restore-server.yml** ファイルのコピーを作成します。

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/restore-server.yml restore-my-server.yml
```

3. Ansible Playbook の **restore-my-server.yml** を開いて編集します。
4. 以下の変数を設定してファイルを調整します。
   a. **hosts** 変数を、インベントリーファイルのホストグループに設定します。この例では、**ipaserver** ホストグループに設定します。
   b. **ipabackup_name** 変数は、復元する **ipabackup** の名前に設定します。
   c. **ipabackup_password** 変数は LDAP Directory Manager パスワードに設定します。

```
---
- name: Playbook to restore an IPA server
  hosts: ipaserver
  become: true

  vars:
    ipabackup_name: ipa-full-2021-04-30-13-12-00
    ipabackup_password: <your_LDAP_DM_password>

  roles:
  - role: ipabackup
    state: restored
```

5. ファイルを保存します。
6. Playbook ファイルとインベントリーファイルを指定して Ansible Playbook を実行します。

```
$ ansible-playbook --vault-password-file=password_file -v -i ~/MyPlaybooks/inventory
restore-my-server.yml
```

関連情報

- **/usr/share/doc/ansible-freeipa/roles/ipabackup** ディレクトリーの **README.md** ファイル
- **/usr/share/doc/ansible-freeipa/playbooks/** ディレクトリー

9.7. ANSIBLE を使用した ANSIBLE コントローラーに保存されているバックアップから IDM サーバーの復元

Ansible Playbook を使用して、Ansible コントローラーに保存されているバックアップから IdM サーバーを復元できます。

前提条件

- 次の要件を満たすように Ansible コントロールノードを設定している。
  - Ansible バージョン 2.14 以降を使用している。
  - Ansible コントローラーに **ansible-freeipa** パッケージがインストールされている。
  - **~/MyPlaybooks/** ディレクトリーに、IdM サーバーの完全修飾ドメイン名 (FQDN) を使用して Ansible インベントリーファイル を作成している (この例の場合)。
  - この例では、**secret.yml** Ansible vault に **ipaadmin_password** が保存されていることを前提としています。
- ターゲットノード (**ansible-freeipa** モジュールが実行されるノード) が、IdM クライアント、サーバー、またはレプリカとして IdM ドメインに含まれている。
- LDAP Directory Manager のパスワードを知っている必要があります。

手順

1. **~/MyPlaybooks/** ディレクトリーに移動します。

```
$ cd ~/MyPlaybooks/
```

2. **/usr/share/doc/ansible-freeipa/playbooks** ディレクトリーにある **restore-server-from-controller.yml** ファイルのコピーを作成します。

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/restore-server-from-controller.yml restore-my-server-from-my-controller.yml
```

3. **restore-my-server-from-my-controller.yml** ファイルを開いて編集します。
4. 以下の変数を設定してファイルを調整します。
   a. **hosts** 変数を、インベントリーファイルのホストグループに設定します。この例では、**ipaserver** ホストグループに設定します。
   b. **ipabackup_name** 変数は、復元する **ipabackup** の名前に設定します。
   c. **ipabackup_password** 変数は LDAP Directory Manager パスワードに設定します。

```
---
```

```
- name: Playbook to restore IPA server from controller
  hosts: ipaserver
  become: true

  vars:
    ipabackup_name: server.idm.example.com_ipa-full-2021-04-30-13-12-00
    ipabackup_password: <your_LDAP_DM_password>
    ipabackup_from_controller: true

  roles:
  - role: ipabackup
    state: restored
```

5. ファイルを保存します。

6. Playbook ファイルとインベントリーファイルを指定して Ansible Playbook を実行します。

```
$ ansible-playbook --vault-password-file=password_file -v -i ~/MyPlaybooks/inventory
restore-my-server-from-my-controller.yml
```

関連情報

- **/usr/share/doc/ansible-freeipa/roles/ipabackup** ディレクトリーの **README.md** ファイル
- **/usr/share/doc/ansible-freeipa/playbooks/** ディレクトリー

第10章 IDM と RED HAT 製品の統合

IdM と統合する他の Red Hat 製品のドキュメントを紹介します。IdM ユーザーがサービスにアクセスできるように、これらの製品を設定することができます。

Ansible Automation Platform

Setting up LDAP authentication

OpenShift Container Platform

LDAP アイデンティティープロバイダーの設定

OpenStack Platform

OpenStack Identity (keystone) と Red Hat Identity Manager (IdM) の統合

Satellite

Red Hat Identity Management の使用

Single Sign-On

SSSD および FreeIPA Identity Management の統合

仮想化

Configuring an external LDAP provider

第11章 IDM ドメインで RHEL 9 WEB コンソールにシングルサインオンを設定

RHEL 9 Web コンソールでの Identity Management (IdM) が提供する SSO (シングルサインオン) 認証を使用する方法を学びます。

利点:

- IdM ドメインの管理者は、RHEL 9 Web コンソールを使用して、ローカルマシンを管理できます。
- IdM ドメインに Kerberos チケットがあると、Web コンソールにアクセスする際にログイン認証情報を指定する必要がなくなりました。
- IdM ドメインが認識しているすべてのホストは、RHEL 9 Web コンソールのローカルインスタンスから SSH 経由でアクセスできます。
- 証明書設定は必須ではありません。コンソールの Web サーバーでは、IdM 認証局が発行した証明書に自動的に切り替わり、ブラウザーに許可されます。

本章は、RHEL Web コンソールにログインするために SSO を設定する手順を説明します。

1. RHEL 9 Web コンソールを使用して IdM ドメインにマシンを追加します。
   詳細はWeb コンソールで IdM ドメインに RHEL 9 システムを参加させる を参照してください。
2. 認証に Kerberos を使用する場合は、マシンで Kerberos チケットを取得する必要があります。
   詳細は、Kerberos 認証を使用した Web コンソールへのログイン を参照してください。
3. IdM サーバーの管理者が、任意のホストで任意のコマンドを実行できます。
   詳細は、管理者の sudo で IdM サーバーのドメイン管理者にアクセス可能に を参照してください。

前提条件

- RHEL Web コンソールが RHEL 9 システムにインストールされている。
  詳細は、Web コンソールのインストール を参照してください。
- RHEL Web コンソールを使用して IdM クライアントがシステムにインストールされている。
  詳細は IdM クライアントのインストール を参照してください。

11.1. WEB コンソールを使用した RHEL 9 システムの IDM ドメインへの参加

Web コンソールを使用して、Red Hat Enterprise Linux 9 システムを Identity Management (IdM) ドメインに参加させることができます。

前提条件

- IdM ドメインが実行中で参加するクライアントから到達可能
- IdM ドメインの管理者認証情報がある。

手順

1. RHEL Web コンソールにログインします。

詳細は、Web コンソールへのログイン を参照してください。

2. **Overview** タブの **Configuration** フィールドで、**Join Domain** をクリックします。
3. **ドメイン参加** ダイアログボックスの **ドメインアドレス** フィールドに、IdM サーバーのホスト名を入力します。
4. **ドメイン管理者名** フィールドで、IdM 管理アカウントのユーザー名を入力します。
5. **Domain administrator password** にパスワードを追加します。
6. **参加** をクリックします。

検証手順

1. システムが IdM ドメインに参加していると、RHEL 9 Web コンソールにエラーが表示されず、**システム** 画面でドメイン名を確認できます。
2. ユーザーがドメインのメンバーであることを確認するには、Terminal ページをクリックし、**id** コマンドを実行します。

```
$ id
euid=548800004(example_user) gid=548800004(example_user)
groups=548800004(example_user) context=unconfined_u:unconfined_r:unconfined_t:s0-
s0:c0.c1023
```

関連情報

- Identity Management の計画
- Identity Management のインストール
- IdM ユーザーグループのホストとアクセス制御ルールの管理

11.2. KERBEROS 認証を使用した WEB コンソールへのログイン

Kerberos 認証を使用するように RHEL 9 システムを設定します。

重要

SSO を使用した場合は、通常、Web コンソールに管理者権限がありません。これは、パスワードがない sudo を設定した場合に限り機能します。Web コンソールは、対話的に sudo パスワードを要求しません。

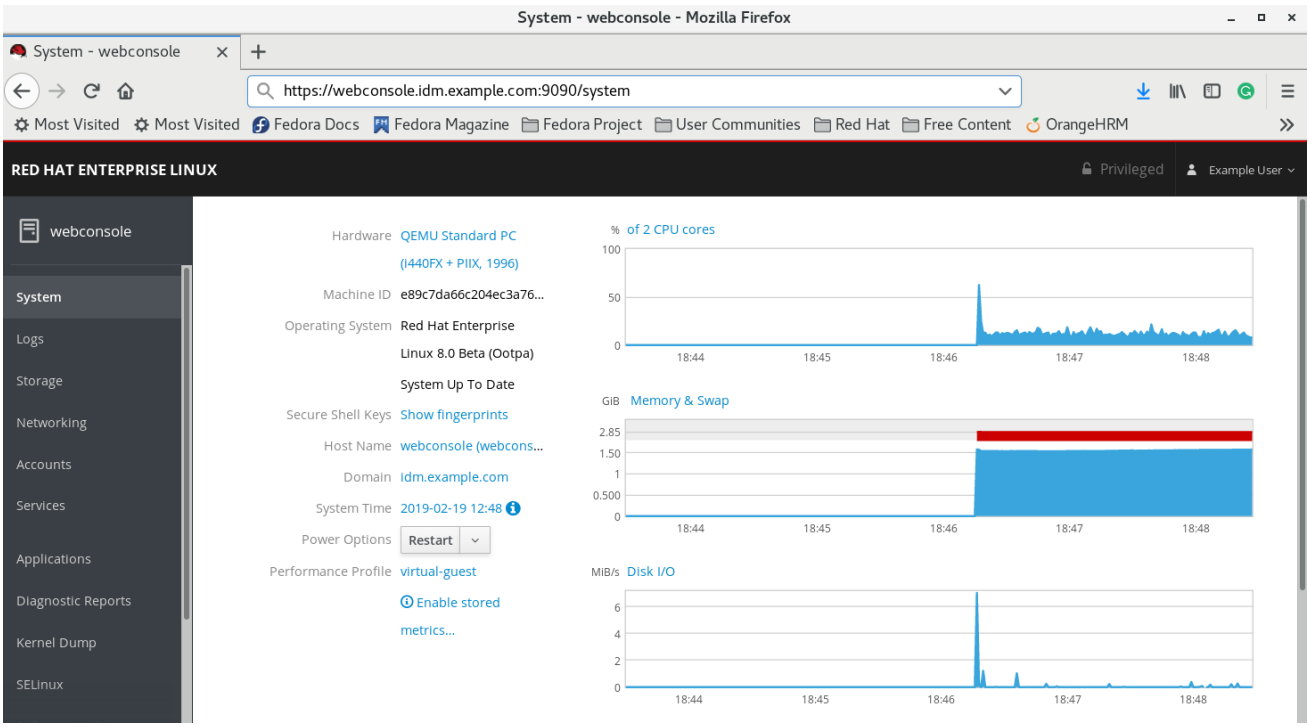
前提条件

- 稼働中で、会社の環境で到達可能な IdM ドメイン
  詳細はWeb コンソールで IdM ドメインに RHEL 9 システムを参加させる を参照してください。
- リモートシステムで、RHEL Web コンソールで接続して管理する **cockpit.socket** サービスを有効にしている。
  詳細は、Web コンソールのインストール を参照してください。
- システムが、SSSD クライアントが管理する Kerberos チケットを使用しない場合は、**kinit** ユーティリティーを使用して手動でチケットを要求してみる。

手順

https://dns_name:9090 から、RHEL Web コンソールにログインします。

この時点で、RHEL Web コンソールへの接続に成功しており、設定を開始できます。



11.3. 管理者の SUDO で IDM サーバーのドメイン管理者にアクセス可能に

RHEL Web コンソールを使用すると、ドメイン管理者が Identity Management (IdM) ドメイン内の任意のホストで任意のコマンドを使用できるようにすることができます。

これを可能にするために、IdM サーバーのインストール時に自動的に作成された **admins** ユーザーグループに sudo がアクセスできるようにします。グループで **ipa-advise** スクリプトを実行すると、**admins** グループに追加されたすべてのユーザーに sudo アクセス権が付与されます。

前提条件

- サーバーが、IdM 4.7.1 以降を実行している。

手順

1. IdM サーバーに接続します。
2. ipa-advise スクリプトを実行します。

```
$ ipa-advise enable-admins-sudo | sh -ex
```

コンソールにエラーが表示されない場合、**admins** グループには IdM ドメイン内のすべてのマシンに対する sudo 権限があります。

第12章 IDM DIRECTORY SERVER の RFC サポート

Identity Management (IdM) の Directory Server コンポーネントは、多くの LDAP 関連の Requests for Comments (RFC) をサポートしています。

関連情報

- Directory Server の RFC サポート
- Directory Server の計画と設計