Red Hat Enterprise Linux 8

Identity Management の計画

Identity Management の計画およびアクセス制御設定のガイド

Last Updated: 2020-09-07

Red Hat Enterprise Linux 8 Identity Management の計画

Identity Management の計画およびアクセス制御設定のガイド

法律上の通知

概要

本書は、Red Hat Enterprise Linux 8 における Identity Management サービスの計画を説明します。本書の現行バージョンには、厳選されたユーザーストーリーが含まれています。

目次

RED HAT ドキュメントへのフィードバック (英語のみ)

ご意見ご要望をお聞かせください。ドキュメントの改善点はございませんか。改善点を報告する場合は、以下のように行います。

- 特定の文章に簡単なコメントを記入する場合は、以下の手順を行います。
    1. ドキュメントの表示が **Multi-page HTML** 形式になっていて、ドキュメントの右上端に **Feedback** ボタンがあることを確認してください。
    2. マウスカーソルで、コメントを追加する部分を強調表示します。
    3. そのテキストの下に表示される **Add Feedback** ポップアップをクリックします。
    4. 表示される手順に従ってください。
- より詳細なフィードバックを行う場合は、Bugzilla のチケットを作成します。
    1. Bugzilla の Web サイトにアクセスします。
    2. Component で **Documentation** を選択します。
    3. **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記入してください。
    4. **Submit Bug** をクリックします。

第1章 RHEL における IDM およびアクセス制御の計画の概要

本セクションでは、Red Hat Enterprise Linux における ID 管理 (IdM) およびアクセス制御のオプションの概要を説明します。本セクションを読むことで、お使いの環境に合わせた計画の準備が行えます。

1.1. IDM の概要

本モジュールでは、Red Hat Enterprise Linux における Identity Management (IdM) の目的を説明します。また、IdM ドメイン (およびこのドメインに含まれるクライアントおよびサーバーのマシン) に関する基本的な情報も提供します。

Red Hat Enterprise Linux における IdM の目的

Red Hat Enterprise Linux の IdM は、Linux ベースのドメイン内で ID ストア、認証ポリシー、および認可ポリシーを一元管理する方法を提供します。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。Red Hat の SSO 技術は、このような IdP の例における Keycloak オープンソースプロジェクトに基づいています。

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。

関連資料

- 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
- Red Hat Enterprise Linux 8 Beta リリースノート

1.2. 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 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

ドメインを作成するのにインストールする最初のサーバーは **IdM マスター** または **マスターサーバー** になります。IdM マスターは、**マスター CA** サーバーと混同しないようにしてください。2 台のマシンで実行できます。

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役割を果たしています。

IdM トポロジーに統合 CA が含まれている場合は、1 台のサーバーに 証明書失効リスト (CRL) 生成マスター と CA Renewal Master の役割もあります。このサーバーは **マスター CA** です。

警告

マスター CA サーバーは、CA サブシステムの証明書および鍵の追跡と CRL の生成を行うドメインにある唯一のシステムであるため、IdM デプロイメントに重要です。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、冗長性および負荷分散のために、既存のサーバー (マスターサーバーまたは別のレプリカ) の **レプリカ** を作成することで、追加サーバーを作成します。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

レプリカと、そのレプリカを作成したサーバーは、CRL 生成マスターの役割を除き機能的に同じです。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。

1.3. 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 サーバーにおける ID およびポリシーの管理 (IdM の使用にかかわらず)

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 を使用しない場合は、以下のようになります。

- 各サーバーは別々に管理されます。
- 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。

1.4. 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。

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。公開鍵の暗号化は、証明書を使用して偽装の問題に対応します。

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 です。**ipa** IdM 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。

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。

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。

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。

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 マスタ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 で定義された値を上書きできます。

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。以下の質問に回答して、IdM デプロイメントを説明できます。

- IdM デプロイメントは、デプロイメントまたは実稼働デプロイメントをテストするか?
  - IdM サーバーは何台あるか?
- IdM デプロイメントに 統合 CA が含まれているか?
  - 存在する場合は、統合 CA 自己署名か、または外部署名であるか?
  - その場合、どのサーバーで CA ロール が利用できるか? KRA ロールは、どのサーバーで利用できるか?
- IdM デプロイメントに 統合 DNS が含まれているか?
  - その場合は、どのサーバーが DNS ロールを利用できるか?
- IdM デプロイメントは AD フォレスト と信頼関係にあるか?
  - その場合は、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 ロールを使用できるか?

IdM マスターおよびレプリカ

ipa-server-install コマンドを使用して IdM ドメインを作成する最初のサーバーは、**マスターサーバー** または **IdM マスター** と呼ばれています。
管理者は、**ipa-replica-install** コマンドを使用すると、マスターに加えて **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、作成元の IdM サーバーとの レプリカ合意 が作成され、残りの IdM への更新の受信および送信が可能になります。

マスターとレプリカの間に機能的な違いはありません。いずれも十分に機能する IdM サーバー になります。

別名 - **マスター**、**マスターサーバー**、**IdM マスターサーバー**

IdM マスター CA 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 つのサーバーに 証明書失効リスト (CRL) 生成マスター と CA 更新マスター の 2 つの役割があります。このサーバーは **マスター CA サーバー** になります。統合 CA のないデプロイメントには、マスター CA サーバーがありません。
別名 - **マスター CA**

重要

IdM マスター と **マスター CA サーバー** は、2 つの異なる用語です。たとえば、次のデプロイメントシナリオでは、1 台目のサーバーは IdM マスターで、レプリカはマスター CA サーバーになります。

1. 統合 CA のない環境で 1 台目の IdM サーバーをインストールします。
2. レプリカをインストールします。
3. CA をレプリカにインストールします。

このシナリオでは、1 台目のサーバーは IdM マスターで、レプリカはマスター CA サーバーになります。

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一覧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/master.example.com@EXAMPLE.COM**

エンティティー	命名規則	例
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 の役割」を参照してください。

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。

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

1.5. 関連情報

- Red Hat IdM に関する一般的な情報は、Red Hat カスタマーポータルの Red Hat Identity Management 製品ページ を参照してください。

第2章 レプリカトポロジーの計画

以下のセクションでは、ユースケースに適したレプリカトポロジーを決定するヒントを紹介します。

2.1. 高性能および災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

Identity Management (IdM) サービスの継続的な機能および高可用性は、リソースにアクセスするユーザーに不可欠です。ロードバランスを介して IdM インフラストラクチャーの継続的な機能および高可用性を実現する組み込みソリューションの 1 つは、マスターサーバーのレプリカサーバーを作成して中央ディレクトリーを複製することです。

IdM を使用すると、企業の組織構造を反映するために、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。このようにして、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パスが短くなります。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。

複数の冗長な IdM サーバーを維持し、それらを互いに複製させることも、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す。たとえば、1 台のサーバーに障害が発生しても、その他のサーバーがドメインにサービスを提供し続けます。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。

2.2. 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 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

ドメインを作成するのにインストールする最初のサーバーは **IdM マスター** または **マスターサーバー** になります。IdM マスターは、**マスター CA** サーバーと混同しないようにしてください。2 台のマシンで実行できます。

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役割を果たしています。

IdM トポロジーに統合 CA が含まれている場合は、1 台のサーバーに 証明書失効リスト (CRL) 生成マスター と CA Renewal Master の役割もあります。このサーバーは **マスター CA** です。

警告

マスター CA サーバーは、CA サブシステムの証明書および鍵の追跡と CRL の生成を行うドメインにある唯一のシステムであるため、IdM デプロイメントに重要です。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、冗長性および負荷分散のために、既存のサーバー (マスターサーバーまたは別のレプリカ) の **レプリカ** を作成することで、追加サーバーを作成します。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

レプリカと、そのレプリカを作成したサーバーは、CRL 生成マスターの役割を除き機能的に同じです。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。

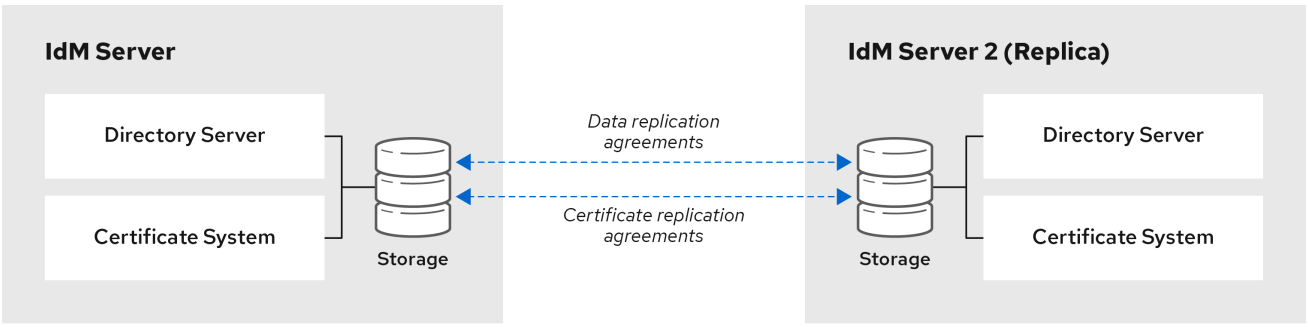
2.3. レプリカ合意

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。

レプリカ合意は常に双方向のものです。1 台目のサーバーから 2 台目のサーバーにデータが複製されるだけでなく、2 台目のサーバーから 1 台目のサーバーにもデータが複製されます。

IdM は、**マルチマスターレプリケーション** を使用します。マルチマスターレプリケーションでは、レプリカ合意に参加している全レプリカが更新を受け取るため、データマスターと見なされます。

図2.1 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構成されている場合は、証明書情報ではなく ID 情報だけが複製されます。

2.4. 適切なレプリカ数の決定

各データセンターに少なくとも 2 つのレプリカを設定 (必須要件ではありません)

データセンターは、たとえば、本社または地理的な位置 (領域) に置かれます。

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設定

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。

1 つの IdM ドメインに最大 60 台のレプリカを設定

Red Hat は、レプリカが最大 60 台含まれる環境をサポートしています。

2.5. トポロジー内でレプリカの接続

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、初期レプリカとマスター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行っても、大きな利点はありません。受信レプリカは、一度に 1 つのレプリカによってのみ更新でき、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、証明書のレプリケーションとドメインのレプリケーションの両方に適用されます。

1台のレプリカに対するレプリケーション合意が 4 つに制限される点について、2 つの例外があります。

- 特定のレプリカがオンラインでないか、応答していない場合はフェールオーバーパスが必要。
- 大規模デプロイメントでは、特定のノード間に追加の直接リンクが必要。

レプリケーション合意を多数構成すると、全体のパフォーマンスに影響を及ぼす場合があります。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、特定のレプリカは、受信更新と送信更新の間で changelog データベースファイルに対して競合が多くなる可能性があります。

レプリカごとに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およびレイテンシーが発生しないようにしてください。ただし、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、データセンター間のドメインレプリケーションが保証されます。

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保証されます。

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。

2.6. レプリカトポロジーの例

以下の図は、信頼できるトポロジーを作成するガイドラインに基づく Identity Management (IdM) トポロジーの例を示しています。

図2.2「レプリカトポロジーの例 1」には 4 つのデータセンターがあり、各データセンターに 4 つのサーバーがあります。このサーバーは、レプリカ合意に接続しています。

図2.2 レプリカトポロジーの例 1

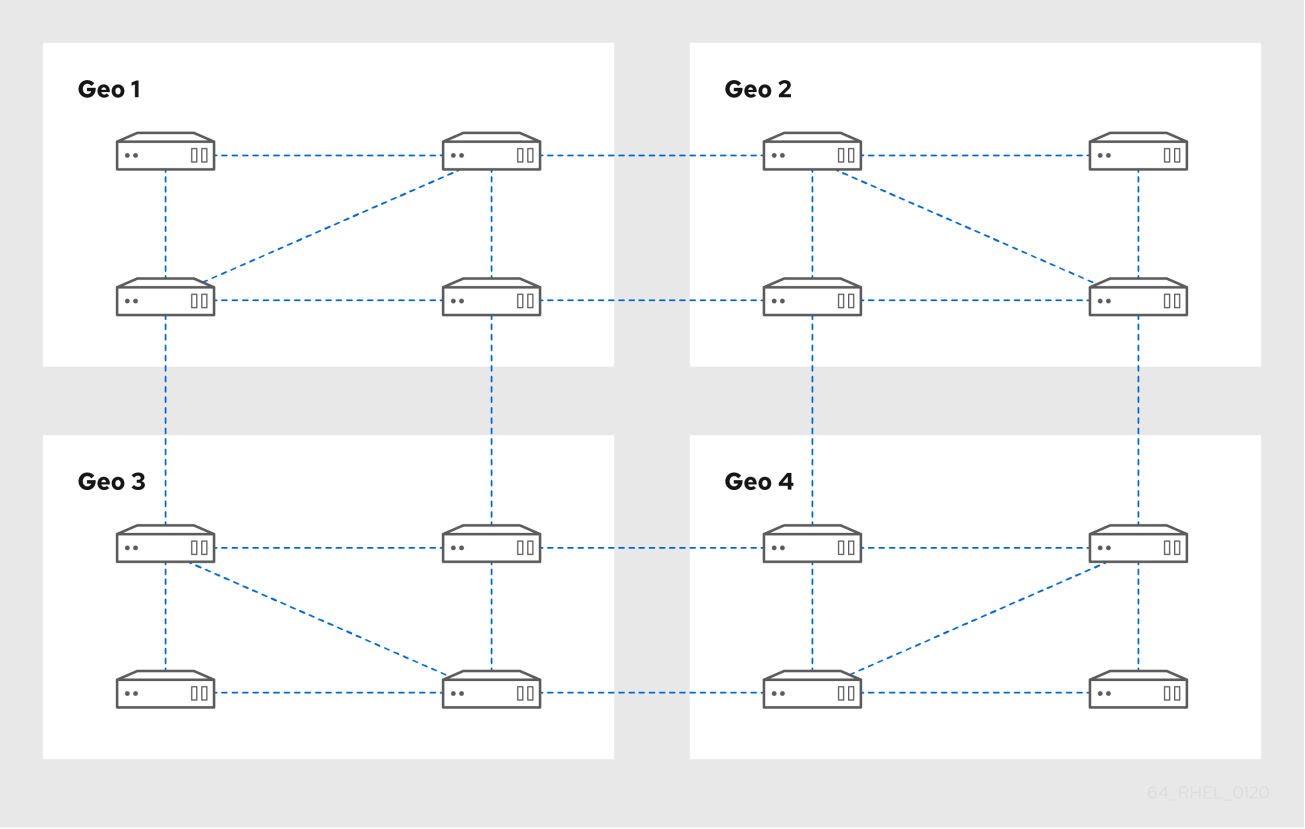
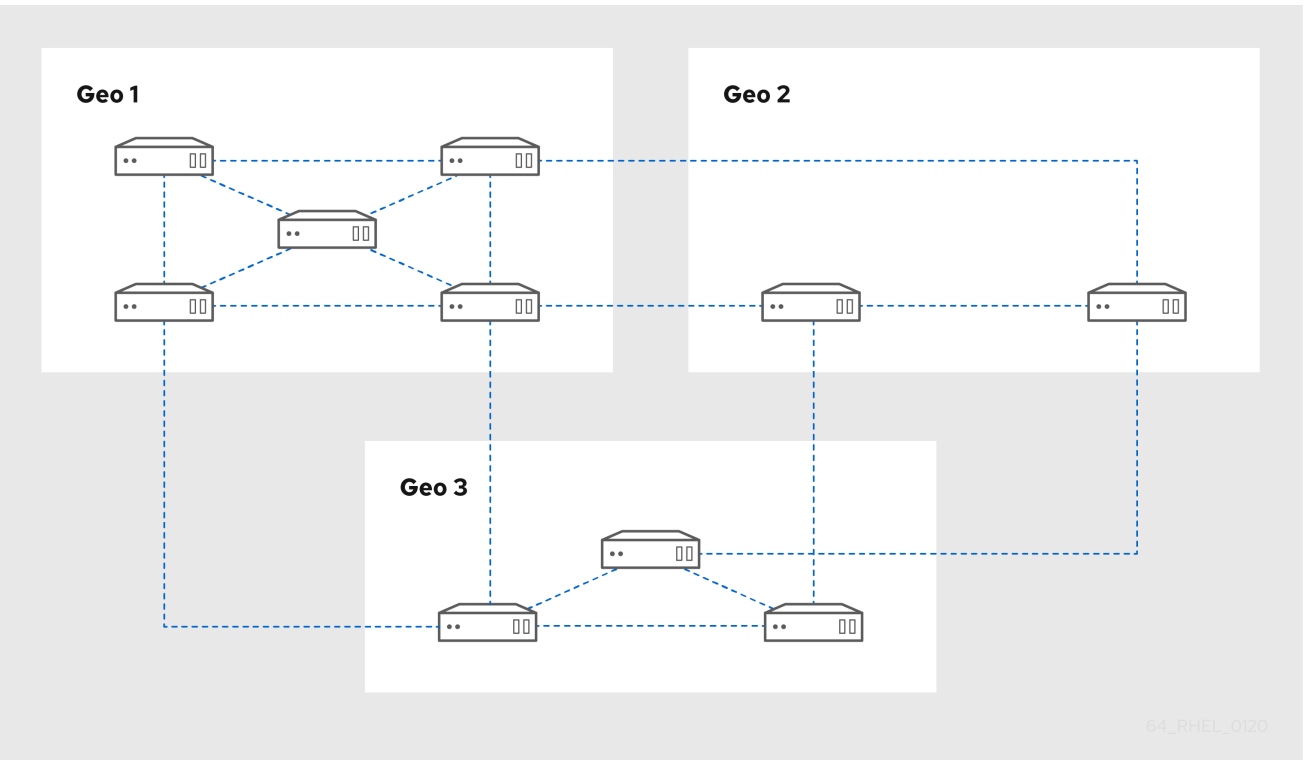


図2.3「レプリカトポロジーの例 2」には、所有するサーバー数が異なる 3 つのデータセンターが表示されます。このサーバーは、レプリカ合意に接続しています。

図2.3 レプリカトポロジーの例 2



2.7. 非表示のレプリカモード

デフォルトでは、新しいレプリカを設定すると、インストーラーは DNS にサービス (SRV) リソースレコードを自動的に作成します。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。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、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。

注記

非表示のレプリカ機能は、テクノロジープレビューとして Red Hat Enterprise Linux 8.1 以降で利用でき、サポート対象外となります。

非表示のレプリカは、主にクライアントを中断できる専用の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は、マスターまたはレプリカ上のすべての 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。

注記

- 新しいホストにある非表示のレプリカからバックアップを復元すると、常に非表示ではない (通常の) レプリカになります。
- クラスターで使用されるすべてのサーバーのロール、特に統合 CA が使用されている場合の認証局のロールは、バックアップでこのようなサービスを復元できるように、非表示のレプリカにインストールする必要があります。
- IdM バックアップの作成方法および操作方法は、「IdM のバックアップおよび復元」を参照してください。

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。レプリカを非表示としてインストールするには、**--hidden-replica** パラメーターを **ipa-replica-install** コマンドに渡します。

レプリカのインストールに関する詳細は、「Identity Management レプリカのインストール」を参照してください。

または、既存のレプリカの状態を変更することもできます。詳細は「Demoting or promoting hidden replicas」を参照してください。

第3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。

3.1. 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3.1 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 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。高度な DNS 機能の一部には対応していません。汎用の DNS サーバーとして使用することは意図されていません。	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。

3.2. 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。このセクションのガイドラインは、名前を正しく設定するのに役に立ちます。

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) で構成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。

第4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。

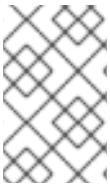
4.1. IDM サーバーで利用可能な CA サービス

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4.1 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、または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。

4.2. 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。

4.3. 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。

- 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、復元できなくなります。この場合は、新規 CA を設定して、新しい証明書をインストールする必要があります。

- 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。

推奨事項は、以下の表を参照してください。

表4.2 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

第5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。

- 直接統合の概要は「直接的な統合」を参照してください。
- 間接統合の概要は「間接的な統合」を参照してください。
- どちらを選択するかは「間接統合と直接統合の間の決定」を参照してください。

5.1. 直接的な統合

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。

5.2. 間接的な統合

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、同期を構成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。

5.3. 間接統合と直接統合の間の決定

本セクションのガイドラインは、どのタイプの統合がユースケースに合うかを判断するのに役に立ちます。

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。

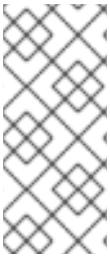
第6章 IDM と AD との間のフォレスト間の信頼の計画

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。

6.1. IDM と AD と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。

6.2. 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。

信頼エージェント

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

信頼コントローラー

Samba スイートも実行する信頼エージェント。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。
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。

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。

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、標準サーバーと通信するクライアントは、AD ユーザーおよびグループを解決できず、AD ユーザーを認証および認可することができません。

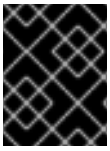
表6.1 信頼コントローラーおよび信頼エージェントが提供する機能の比較

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	はい	はい
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	はい	はい
信頼の管理 (たとえば、信頼合意の追加)	いいえ	はい

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。

6.3. 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。

6.4. 非 POSIX の外部グループおよび 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。

6.5. DNS の設定

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 構成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 ドメインと AD DNS ドメインとの間に重複がない

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。IdM クライアントを含む DNS ドメインが、AD に参加しているシステムを含む DNS ドメインと重複しないようにします。

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。

6.6. NETBIOS 名

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。

6.7. 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 2008 ~ Windows Server 2016
- ドメイン機能レベルの範囲 - Windows Server 2008 ~ Windows Server 2016

Identity Management (IdM) は、以下のオペレーティングシステムに対応しています。

- Windows Server 2008
- Windows Server 2008 R2
- Windows Server 2012
- Windows Server 2012 R2
- Windows Server 2016
- Windows Server 2019

6.8. AD サーバーの検出およびアフィニティーの設定

サーバー検出とアフィニティー設定は、Identity Management (IdM) クライアントが接続する Active Directory (AD) サーバーに影響を及ぼします。本セクションは、IdM と AD との間でフォレスト間の信頼関係がある環境で、検出とアフィニティーがどのように機能するかを説明します。

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。クライアントが確実にローカルサーバーと通信するようにするには、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。

6.9. IDM と AD への間接統合中に実行する操作

表6.2「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」は、Identity Management (IdM) 信頼コントローラーから Active Directory (AD) ドメインコントローラーに向けた、IdM から AD への信頼の作成時に実行される操作およびリクエストを示します。

表6.2 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
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6.3「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」は、AD ドメインコントローラーから IdM 信頼コントローラーに向けた、IdM から AD への信頼の作成時に実行される操作およびリクエストを示します。

表6.3 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

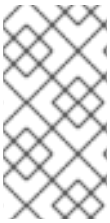
操作	使用プロトコル	目的
AD ドメインコントローラーに設定された IdM の DNS リゾルバーに対する DNS 解決	DNS	IdM 信頼コントローラーの IP アドレスを検出する

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 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
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7章 IDM のバックアップおよび復元

Red Hat Enterprise Linux Identity Management は、IdM システムを手動でバックアップして復元するソリューションを提供します。これは、データを損失したときに必要になる場合があります。

バックアップ時に、システムは IdM セットアップに関する情報を含むディレクトリーを作成して保存します。復元時に、このバックアップディレクトリーを使用して、元の IdM セットアップを復元できます。



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。サーバーの喪失による影響を緩和し、代替サーバーをクライアントに提供することで継続的な運用を保証するには、「レプリケーションによるサーバー損失の緩和」に記載されるようにレプリカトポロジーがあることを確認してください。

7.1. IDM バックアップタイプ

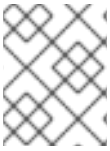
IdM は、サーバーのフルバックアップと、データのみのバックアップという 2 種類のバックアップを提供します。

バックアップタイプ	バックアップのコンテンツ	実行されたオンラインまたはオフライン	実施例
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 • LDAP データ交換形式 (LDIF) の LDAP データ	オフラインのみ。IdM サービスを一時的に停止する必要があります。	IdM デプロイメントのゼロからの再構築
データのみのバックアップ	• LDAP データ交換形式 (LDIF) の LDAP データ • レプリケーション変更ログ	オンラインまたはオフライン。	IdM データを以前の状態に復元

7.2. バックアップファイルの規則

デフォルトでは、IdM はバックアップを **/var/lib/ipa/backup/** ディレクトリーに保存します。このサブディレクトリーの命名規則は以下のとおりです。

- サーバーのフルバックアップ - GMT 時間で **ipa-full-YEAR-MM-DD-HH-MM-SS**
- データのみのバックアップ - GMT 時間で **ipa-data-YEAR-MM-DD-HH-MM-SS**

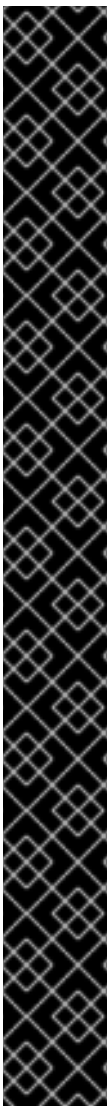


注記

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。

7.3. バックアップの作成

本セクションでは、**ipa-backup** コマンドを使用して、オフラインモードおよびオンラインモードでサーバーのフルバックアップと、データのみのバックアップを作成する方法を説明します。



重要

- デフォルトでは、**ipa-backup** はオフラインモードで実行され、すべての IdM サービスを停止します。バックアップが完了すると、サービスが自動的に起動します。
- サーバーのフルバックアップは、**常に** IdM サービスをオフラインで使用して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、オンラインのサービスで実行できます。
- デフォルトでは、バックアップは **/var/lib/ipa/backup/** ディレクトリーを含むファイルシステムに作成されます。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が推奨されます。
- 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、IdM クライアントに影響を及ぼさずに、非表示のレプリカでシャットダウンできます。
- サーバーの IdM バックアップは、そのサーバーにインストールされているサーバーロールのみを取得します。
  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 以外のレプリカのバックアップは CA データを **取得しません**。同様に、KRA が **インストールされていない** レプリカのバックアップも、KRA データを **取得しません**。
- IdM デプロイメントでビルトイン CA を使用する場合、CA なしのレプリカのバックアップでは、IdM デプロイメントを再構築するには不十分です。使用中の IdM サーバーロールがすべてインストールされているレプリカ (CA、KRA、DNS) にバックアップを作成してください。

ipa-backup コマンドの使用例

- 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。

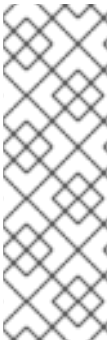
```
[root@server ~]# ipa-backup --data
```

- 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宛先を変更します。

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

詳細は「ipa-backup command fails to finish」 を参照してください。

検証手順

- バックアップディレクトリーには、バックアップが含まれるアーカイブが含まれます。

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

7.4. 暗号化 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。暗号化した IdM バックアップを作成するには、最初に GPG2 キーを作成する必要があります。

7.4.1. IdM バックアップを暗号化する GPG2 キーの作成

以下の手順では、**ipa-backup** ユーティリティーの GPG2 キーを生成する方法を説明します。

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: IPA Backup
Name-Comment: IPA Backup
Name-Email: root@example.com
Expire-Date: 0
%commit
%echo Finished creating standard key
EOF
```

3. 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** の内容に基づいて新規の GPG2 キーの生成を開始します。

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

a. GPG2 キーを保護するパスフレーズを入力します。

```
┌──────────────────────────────────────────────────────┐
│ Please enter the passphrase to                       │
│ protect your new key                                 │
│                                                      │
│ Passphrase: SecretPassphrase42                       │
│                                                      │
│  <OK>                          <Cancel>              │
└──────────────────────────────────────────────────────┘
```

b. 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: SecretPassphrase42                       │
│                                                      │
│  <OK>                          <Cancel>              │
└──────────────────────────────────────────────────────┘
```

c. これで新しい GPG2 キーが作成されました。

```
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 キーの一覧を表示します。

```
[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] IPA Backup (IPA Backup) <root@example.com>
```

関連情報

- GPG 暗号化とその使用に関する詳細は、GNU Privacy Guard の Web サイトを参照してください。

7.4.2. GPG2 で暗号化した IdM バックアップの作成

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。詳細は「IdM バックアップを暗号化する 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

検証手順

- バックアップディレクトリーに **.gpg** ファイル拡張子が付いた暗号化されたアーカイブが含まれるようにします。

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

関連情報

- バックアップの作成に関する詳細は、「バックアップの作成」を参照してください。

7.5. 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 は、問題を診断し、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をご利用になることが推奨されます。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サーバーロスからの復旧」を参照してください。

7.6. 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。

- バックアップが作成された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

- マルチマスター環境の IdM サーバーが復元されると、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。他のすべてのマスターサーバーは復元されたサーバーから再度初期化される必要があります。
- 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。

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。

7.7. バックアップからの IDM サーバーの復元

以下の手順では、IdM バックアップから IdM サーバーまたはその LDAP データを復元する方法を説明します。

図7.1 この例で使用されるレプリケーショントポロジー

表7.1 この例で使用されるサーバーの命名規則

サーバー名	機能
master1.example.com	バックアップから復元する必要があるサーバー
caReplica2.example.com	Master1.example.com に接続している認証局 (CA) レプリカ。
replica3.example.com	CaReplica2.example.com に接続しているレプリカ。

前提条件

- IdM サーバーの完全なサーバーまたはデータのみのバックアップは、**ipa-backup** ユーティリティーで生成されました。「バックアップの作成」を参照してください。

- 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. **lpa-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@master1 ~]# 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@master1 ~]# ipa-restore /mybackups/ipa-data-2020-02-01-05-30-00
     ```

     注記

     lpa-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** に追加します。

     ```
     [root@master1 ~]# 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
   master1.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. **lpa-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 master1.example.com to caReplica2.example.com
   Disabling CA replication agreement on master1.example.com to caReplica2.example.com
   Disabling replication agreement on caReplica2.example.com to master1.example.com
   Disabling replication agreement on caReplica2.example.com to replica3.example.com
   Disabling CA replication agreement on caReplica2.example.com to master1.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一覧を表示します。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@master1 ~]# ipa topologysegment-find domain
   ------------------
   2 segments matched
   ------------------
     Segment name: master1.example.com-to-caReplica2.example.com
     Left node: master1.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、**master1** からデータを使用して **caReplica2** の再初期化を実行します。

   ```
   [root@caReplica2 ~]# ipa-replica-manage re-initialize --from=master1.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一覧を表示します。

   ```
   [root@master1 ~]# ipa topologysegment-find ca
   -----------------
   1 segment matched
   -----------------
     Segment name: master1.example.com-to-caReplica2.example.com
     Left node: master1.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、**master1** からのデータを使用して **caReplica2** の **csreplica** の再初期化を実行します。

```
[root@caReplica2 ~]# ipa-csreplica-manage re-initialize --
from=master1.example.com
Directory Manager password:

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

6. 復元されたサーバー **master1.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

- man ページの **ipa-restore**(1) 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。

7.8. 暗号化されたバックアップからの復元

lpa-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、デフォルトでは GPG2 root キーリングと **gpg-agent** 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。「暗号化 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。詳細は「IdM バックアップを暗号化する GPG2 キーの作成」を参照してください。

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

2. **lpa-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:        │
│ "IPA Backup (IPA Backup) <root@example.com>"                         │
│ 2048-bit RSA key, ID BF28FFA302EF4557,                               │
│ created 2020-01-13.                                                  │
│                                                                      │
│                                                                      │
│ Passphrase: SecretPassPhrase42                                       │
│                                                                      │
│       <OK>                              <Cancel>                     │
└──────────────────────────────────────────────────────────────────────┘
```

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。