Red Hat Enterprise Linux 7 Windows 統合ガイド

Linux システムの Active Directory 環境との統合

Ella Deon Ballard　Tomáš Čapek　Aneta Petrová

Red Hat Enterprise Linux 7 Windows 統合ガイド

Linux システムの Active Directory 環境との統合

Ella Deon Ballard
Red Hat Customer Content Services
dlackey@redhat.com

Tomáš Čapek
Red Hat Customer Content Services
tcapek@redhat.com

Aneta Petrová
Red Hat Customer Content Services
apetrova@redhat.com

法律上の通知

概要

異種の IT 環境には、シームレスな通信が必要な各種のドメインやオペレーティングシステムが含まれています。Red Hat Enterprise Linux は、Linux を Microsoft Windows の Active Directory (AD) に緊密に統合するための複数の方法を提供します。この統合は、複数のユーザー、グループ、サービス、またはシステムを含む複数の異なるドメインオブジェクトに対して実行できます。本書では、軽量 AD パススルー認証から本格的な Kerberos で信頼されるレルムまでの様々な統合シナリオについても説明します。

目次

第1章 Active Directory と Linux 環境の統合方法

IT 環境にはそれぞれの構造があり、IT 環境内のシステムは目的別に配置されます。2 つの別々のインフラストラクチャーを統合するには、それぞれの環境のインフラストラクチャーの目的を判断し、それらがどのように、またどこで相互に作用するかを理解する必要があります。

1.1. Windows 統合の定義

Windows 統合は、Linux 環境と Windows 環境間でどのような相互作用が必要かによってその意味がかなり異なります。この統合は、個々の Linux システムを Windows ドメインに登録すること、Linux ドメインを Windows ドメインのピアに設定すること、または単にこれらの環境間で情報をコピーすることを意味します。

Windows ドメインと Linux システム間にはいくつかの接点があります。これらの接点では、異なるドメインオブジェクト (ユーザー、グループ、システム、サービス) の識別とその識別に使用されるサービスが主に実行されます。

ユーザー識別子および認証

- ユーザーアカウントが置かれる場所: Windows (AD ドメイン) 上で実行される中央の認証システムか、または Linux 上で実行される中央のアイデンティティーおよび認証サーバーか?
- Linux システムのユーザーの認証方法: ローカル Linux 認証システムか、または Window 上で実行される中央認証システムか?
- ユーザーのグループメンバーシップの設定方法: グループメンバーシップの判別方法は?
- ユーザーの認証方法: ユーザー名/パスワードのペア、Kerberos チケット、証明書、またはこれらのメソッドの組み合わせが使用されるのか?
- Linux マシンのサービスへのアクセスに必要な POSIX 属性の保存方法: これらの属性は Windows ドメインで設定されるか、Linux システムでローカルに設定されるか、または動的にマップされるか (UID/GID 番号と Windows SID)?
- どのユーザーがどのリソースにアクセスするか: Windows で定義されたユーザーは Linux リソースにアクセスできるか? Linux で定義されたユーザーは Windows リソースにアクセスできるか?

ほとんど環境では、Active Directory ドメインがユーザー情報の中央ハブになります。Linux システムが認証要求のためにユーザー情報にアクセスするには何らかの経路が必要になります。ここでは、そのユーザー情報を取得する**方法** にはどのようなものがあり、そのユーザー情報の内、外部システムが利用できる情報はどの程度あるかという点を考えることができます。また、Linux システム (POSIX 属性) および Linux ユーザー (特定のアプリケーション管理者) に必要な情報とその情報が管理される方法との間には一定のバランスが必要です。

ホストおよびサービスプリンシパル

- どのリソースがアクセスされるか?
- どの認証プロトコルが必要か?
- Kerberos チケットはどのように取得されるか? SSL 証明書はどのように要求され、検証されるか?
- ユーザーは単一ドメイン、または Linux ドメインと Windows ドメインの両方にアクセスする必要があるか?

DNS ドメイン、クエリーおよび名前解決

- DNS 設定をどのように行うか?
- 単一 DNS ドメインがあるか? 複数のサブドメインがあるか?
- システムのホスト名はどのように解決されるか?
- サービス検出はどのように設定されるか?

セキュリティーポリシー

- アクセス制御の指示が設定される場所は?
- 各ドメインに設定される管理者は?

変更管理

- システムがドメインに追加される頻度はどの程度か?
- DNS サービスなど、Windows 統合の関連要素についての基礎的な設定が変更される場合、それらの変更はどのように伝播されるか?
- 設定はドメイン関連のツールまたはプロビジョニングシステムで維持されるか?
- 統合パスには Windows サーバー上のアプリケーションまたは設定が追加で必要か?

ドメイン内の統合される要素と同様に、その統合がどのように維持されるかも重要な点になります。環境内に頻繁に更新されるシステムが多数含まれる場合には、手作業に大きく依存する特定の統合方法は保守の面で機能しない可能性があります。

以下のセクションでは、Windows との統合についての主要なシナリオを概略します。直接的な統合では、Linux システムは Active Directory に追加の中継なしに接続されます。一方、間接的な統合ではアイデンティティーサーバーが使用されます。このサーバーは Linux システムを中央で管理し、その環境全体をサーバー対サーバーレベルで Active Directory に接続します。

1.2. 直接的な統合

Linux システムを Active Directory (AD) に接続するには 2 つのコンポーネントが必要です。1 つのコンポーネントは、中央のアイデンティティーおよび認証ソース (この場合は AD) と対話します。もう 1 つのコンポーネントは、利用可能なドメインを検出し、正しい認証ソースを使用するように 1 つ目のコンポーネントを設定します。情報を取得し、AD に対して認証を実行するために使用できるオプションは複数あります。それらには以下が含まれます。

ネイティブ LDAP と Kerberos PAM および NSS モジュール

これらのモジュールには、**nss_ldap**、**pam_ldap**、および **pam_krb5** が含まれます。PAM および NSS モジュールはすべてのアプリケーションプロセスにロードされるので、それらは実行環境に直接影響を与えます。キャッシュやオフラインサポート、またはアクセス資格情報の保護などがない場合は、NSS および PAM 用に基本的な LDAP および Kerberos モジュールを使用することは、機能的に制限があるために推奨されません。

Samba Winbind

Samba Winbind の使用は、Linux システムを AD に接続する従来の方法でした。Winbind は Linux システム上で Windows クライアントをエミュレートし、AD サーバーに通信できます。System Security Services Daemon (SSSD) の最新バージョンでは Samba Winbind と SSSD 間に機能的なキャップはなくなり、SSSD は Winbind の置き換えとして使用できるようになりました。Winbind を依然として使用する必要があるケースも稀にありますが、一般的には Winbind が第一のオプションとして使用されることはなくなりました。

System Security Services Daemon (SSSD)

SSSD の主な機能として、システムにキャッシュおよびオフラインサポートを提供する共通フレームワークから、リモートのアイデンティティーおよび認証リソースにアクセスする機能があります。SSSD は高度に設定可能であり、PAM および NSS 統合を提供するだけでなく、中央サーバーから取得されるコアおよび拡張ユーザーデータと共にローカルユーザーを保存するデータベースを提供します。SSSD は、 Active Directory であれ、Red Hat Enterprise Linux の Identity Management (IdM) であれ、または汎用的な LDAP および/または Kerberos サーバーであれ、ユーザーが選択するアイデンティティーサーバーに Linux システムを接続する際に推奨されるコンポーネントです。

Winbind から SSSD に切り替える主な理由には、SSSD が直接的な統合および間接的な統合の両方に利用でき、多額の移行コストなしにある統合アプローチを別の統合アプローチに切り替えることができる点があります。Linux システムを AD に直接的に統合するために SSSD または Winbind を設定する際の最も便利な方法として、**`realmd`** サービスを使用することができます。このサービスを使用することにより、呼び出し元は、標準的な方法でネットワークの認証およびドメインのメンバーシップを設定することができます。**`realmd`** サービスは、アクセス可能なドメインおよびレルムについての情報を自動的に検出し、ドメインまたはレルムに参加するために詳細な設定を必要としません。

直接的な統合は、Linux システムを AD 環境に導入する簡単な方法です。ただし、Linux システムのシェアが拡大すると、通常デプロイメントにおいてホストベースのアクセス制御、sudo、または SELinux ユーザーのマッピングなどのアイデンティティー関連のポリシーをより効果的に一元管理する必要が生じます。最初は Linux システムのこれらの分野の設定はローカル設定ファイルで維持することができますが、システムの数が増えると、Red Hat Satellite などのプロビジョニングシステムを使用する方が、設定ファイルの配信と管理をより簡単に行うことができます。ただし、この方法では設定ファイルを変更してからファイルを配信することによるオーバーヘッドが生じます。直接的な統合における拡張が予想されない場合は、次のセクションで説明する間接的な統合を検討するとよいでしょう。

1.3. 間接的な統合

間接的な統合の主な利点は、Active Directory (AD) ドメインのユーザーが Linux システムおよびサービスに透過的にアクセスできるようにすると共に、Linux システムとそれらのシステムに関するポリシーを一元的に管理できる点にあります。この間接的な統合には、以下のような 2 つの異なるアプローチがあります。

信頼ベースのソリューション

推奨されるアプローチとしては、Red Hat Enterprise Linux の Identity Management (IdM) を Linux システムを制御する中央サーバーとして利用し、AD とのクロスレルム Kerberos 信頼を設定し、AD のユーザーがログオンおよびシングルサインオンを使用して Linux システムおよびリソースにアクセスできるようにする方法があります。このソリューションでは、Kerberos 機能を使用して異なるアイデンティティーソース間の信頼を設定します。IdM は自らを別個のフォレストとして AD に表示し、AD でサポートされるフォレストレベルの信頼の利点を活用します。

複雑な環境では、単一の IdM フォレストは複数の AD フォレストに接続することができます。このセットアップにより、組織内の異なる業務/機能をより効果的に分離することができます。AD 管理者はユーザーおよびユーザー関連のポリシーに焦点を当て、Linux 管理者は Linux インフラストラクチャーを全面的に管理します。このケースでは、IdM で制御される Linux レルムは AD リソースドメインまたはレルムに類似しますが、Linux システムがこれに組み込まれています。

注記

Windows では、すべてのドメインが Kerberos レルムであると同時に DNS ドメインになります。ドメインコントローラーで管理されるすべてのドメインには、独自の専用 DNS ゾーンが設定されている必要があります。IdM がフォレストとして AD によって信頼される場合にも同じことが当てはまります。AD は IdM に独自の DNS ドメインがあることを期待します。信頼のセットアップが機能するには、DNS ドメインを Linux 環境の専用ドメインとして設定する必要があります。

同期ベースのソリューション

これは信頼ベースソリューションの代替ソリューションで、IdM または Red Hat Directory Server (RHDS) でも利用できるユーザー同期機能を使用します。この同期により、ユーザーアカウント (RHDS の場合はグループアカウントも含む) を AD から IdM または RHDS に同期させることができます。ただし、このアプローチには以下を含む一連の制約があります。

- ユーザーの重複
- パスワードを同期する必要。これには AD ドメインのすべてのドメインコントローラーに特殊なコンポーネントが必要になります。
- パスワードを取り込むことができること。すべてのユーザーは初回にパスワードを手動で変更する必要があります。
- 同期は単一ドメインのみに対応する。
- IdM または RHDS の 1 つのインスタンスにデータを同期するのに使用できる AD のドメインコントローラーは 1 つのみである。

統合シナリオによってはユーザーの同期オプションしか選択できない場合がありますが、一般的には同期アプローチがクロスレルムの信頼ベース統合よりも奨励されることはありません。

パート I. 単一 Linux システムの Active Directory ドメインへの追加

第2章 Active Directory を SSSD のアイデンティティープロバイダーとして使用する

System Security Services Daemon (SSSD) は、複数の異なるアイデンティティーおよび認証プロバイダーへのアクセスを提供します。このサービスは、ローカルシステムをより大きなバックエンドシステムに関連付けます。単純な LDAP ディレクトリー、Active Directory (AD) のドメイン、Red Hat Enterprise Linux の Identity Management (IdM)、または Kerberos レルムなどがこれに相当します。

SSSD は、認証情報を取得するために ID ストアに接続してから、ユーザーおよび資格情報のローカルキャッシュを作成するためにこれを使用する方法を設定します。また、SSSD はグループ情報を引き込むこともできます。認可情報は、IdM の HBAC (Host-Based Access Control) および AD の GPO (グループポリシーオブジェクト) を使用して SSSD によって収集されます。

2.1. SSSD について

SSSD サービスは、ローカルアプリケーションと任意の設定済みデータストア間の仲介役として機能します。この 2 者間の関係により、管理者には数多くの利点がもたらされます。

- **識別および認証サーバーへの負荷を軽減。** すべてのアプリケーションサービスが識別サーバーに直接接続することを試行するのではなく、それぞれのローカルアプリケーションが SSSD に接続してから、識別サーバーへの接続、またはそのキャッシュの検査が行われます。
- **オフライン認証のオプション。** SSSD は、リモートサービスから取得するユーザー ID (オプションとしてユーザー資格情報も含む) のキャッシュを維持します。これにより、ユーザーはリモート識別サーバーまたはローカルマシンがオフラインの場合でも認証を行うことができます。
- **単一ユーザーアカウント。** ユーザーは 2 つ以上のユーザーアカウントを持つことができます。たとえば、ローカルシステム用のアカウントと組織上のシステムのアカウントを持つことができます。これは、仮想プライベートネットワーク (VPN) に接続するために必要です。SSSD はキャッシュおよびオフライン認証をサポートするので、リモートユーザーはそれぞれのローカルマシンに対する認証を行うだけでネットワークリソースに接続でき、その後は SSSD がそれらのネットワーク資格情報を維持します。

2.1.1. SSSD 設定

SSSD はシステムをより大規模な外部の ID サービスに接続するローカルサービスです。この接続は SSSD 設定ファイルに **ドメイン** を設定して実行されます。それぞれのドメインは異なる外部データソースを表します。複数のドメインは全体として **アイデンティティープロバイダー** を常に表します。これは、ユーザー情報を指定し、オプションで認証またはパスワード変更などの異なる操作用に他のプロバイダーを定義します。

注記

SSSD は、すべてのユーザー ID を別々の外部アイデンティティーソースに維持できるようにします。Windows 統合の場合は通常 AD ドメインがユーザーアカウントを管理するために使用されます。ローカルシステムのユーザーを作成したり、それらを AD のユーザーアカウントと同期させる必要はありません。SSSD は Windows ID を使用し、Windows ユーザーがローカルシステムおよびローカルサービスにアクセスできるようにします。

さらに SSSD は、システム上のどのサービスが資格情報やユーザーアカウントをキャッシュするために SSSD を使用するかを定義します。これらは、Name Service Switch (NSS) および Pluggable Authentication Modules (PAM) など、高レベルのアプリケーションによって使用される基礎的なセキュリティーサービスに関連します。

例2.1 単純な `sssd.conf` ファイル

```
[sssd]
domains = WIN.EXAMPLE.COM
services = nss, pam
config_file_version = 2

[domain/WINDOWS]
id_provider = ad
auth_provider = ad
access_provider = ad
```

2.1.2. Active Directory ドメイン設定

例2.1「単純な **`sssd.conf`** ファイル」 に示されるように、SSSD 設定ファイルには 2 つの主なセクションがあります。最初のセクションは SSSD サービス (**`[sssd]`**) を設定し、2 つ目のセクションはアイデンティティードメイン (**`[domain/NAME]`**) を設定します。さらに **`[nss]`** または **`[pam]`** など、SSSD をアイデンティティーキャッシュとして使用するシステムサービスを設定するための追加のセクションが含まれる場合があります。

デフォルトでは、アイデンティティープロバイダー (**`id_provider`**) および認可プロバイダー (**`access_provider`**) オプションのみを設定する必要があります。**`id_provider`** オプションは、他のタイプまたはサーバーが設定されていない場合に認証 (**`auth_provider`**) およびパスワードプロバイダー (**`chpass_provider`**) オプションに使用されます。**`ad`** 値を使用すると、Active Directory を任意の種類のプロバイダーに設定できます。

```
[domain/AD_EXAMPLE]
id_provider = ad
auth_provider = ad
access_provider = ad
chpass_provider = ad

ad_server = dc1.example.com
# only needed if DNS discovery is not working
ad_hostname = client.example.com
# only needed if the host name of the client machine is incorrect
ad_domain = example.com
# only needed if AD domain is named differently than SSSD domain
```

接続情報は、使用する Active Directory サーバーを識別するために必要です。基本設定のほかにも、Active Directory アイデンティティープロバイダーは Active Directory 環境用に設定したり、POSIX 属性、ローカルシステム上の Windows SID のマッピング、フェイルオーバーサーバー、ホームディレクトリーなどのアカウント情報を使用するかどうかなどの特定機能を使用できるように設定できます。

Active Directory 固有の設定パラメーターのほかにも、すべての LDAP ドメインプロバイダーを Active Directory プロバイダーで使用できます。詳細の一覧は、sssd-ldap および sssd-ad の man ページで参照できます。

汎用 LDAP プロバイダー設定には、Active Directory プロバイダーの設定に使用できる数多くのオプション

があります。**ad** 値の使用は、Active Directory の指定プロバイダーを設定するためにパラメーターおよび値を自動的に引き込むためのショートカットになります。たとえば、アクセスプロバイダーのショートカットは以下のようになります。

```
access_provider = ad
```

汎用 LDAP パラメーターを使用すると、この設定は以下のように拡張します。

```
access_provider = ldap
ldap_access_order = expire
ldap_account_expire_policy = ad
```

これらの設定すべては、**ad** プロバイダータイプを使用して暗黙的に設定されます。

2.2. SSSD の環境

ほとんどの場合、SSSD は NIS および Winbind などの Windows 統合に使用される古いアイデンティティー管理サービスの置き換えとして使用できます。SSSD はローカルシステムのサービスであるため、システム数が少ない環境の場合にのみこれを手動で設定することができます。

SSSD Active Directory ドメインの初期設定を準備するために使用できるツールがあります。**realmd** スイートは、すべての基礎となる設定ファイルを自動で編集します。このツールは設定の編集を単純化しますが、各システムで別々に実行する必要があります。IdM サーバーはクライアントが Active Directory-IdM 間の信頼に基づいて機能するように設定できますが、これには設定済みかつ実行中の IdM Linux ドメインと設定済みの信頼環境が必要です。

2.3. SSSD を Active Directory 環境に統合する方法

2.3.1. ローカルシステム上の Active Directory アイデンティティー

Windows と Linux では、システムユーザーを処理する方法に構造上の違いがあります。Active Directory で使用されるユーザースキーマと標準の LDAPv3 ディレクトリーサービスにも大きな違いがあります。Active Directory アイデンティティープロバイダーを SSSD と共に使用する場合、Active Directory 形式のユーザーを新規 SSSD ユーザーに対して調整する必要があります。これは以下の 2 つの方法で実行することができます。

- SSSD の ID マッピングは Active Directory セキュリティー ID (SID) と Linux で生成される UID 間のマップを作成します。ID マッピングのオプションは、Active Directory に追加のパッケージや設定が不要なため、ほとんどの環境について最も単純なオプションになります。
- Unix サービスは Windows ユーザーおよびグループエントリーの POSIX 属性を管理できます。これには Active Directory 環境内により多くの設定および情報が必要になりますが、このサービスにより特定の UID/GID 値および他の POSIX 属性に対して管理上の制限を強化できます。

Active Directory はユーザーエントリーおよび属性を、ローカルディレクトリーから **グローバルカタログ** に複製できます。これにより、フォレスト内の他のドメインでその情報が利用できるようになります。パフォーマンスの点では、グローバルカタログのレプリケーションは、SSSD でユーザーおよびグループについての情報を取得するための推奨される方法で、これにより SSSD はトポロジー内のすべてのドメインのすべてのユーザーデータにアクセスできます。その結果、SSSD は Active Directory グローバルカタログでユーザーまたはグループ情報を照会する必要のあるアプリケーションで使用することができます。

2.3.1.1. セキュリティー ID マッピングについて

ID マッピングのメカニズム

Linux/Unix システムは、ローカルユーザー ID 番号 (UID) およびグループ ID 番号 (GID) を使用して、システム上のユーザーを識別します。これらの **UID:GID** 番号は、たとえば **501:501** のような単純な整数になります。

Microsoft Windows と Active Directory は、ユーザー、グループ、およびマシンを識別するために異なるユーザー ID 構造を使用します。*セキュリティー ID* (SID) は、セキュリティーバージョン、発行局のタイプ、マシン、および ID 自体を特定する複数の異なるセグメントで構成されます。3 番目から 6 番目のブロックはマシン ID です。

```
S-1-5-21-3623811015-3361044348-30300820-1013
```

最後のブロックは特定のエンティティーを特定する **相対 ID** (RID) です。

```
S-1-5-21-3623811015-3361044348-30300820-1013
```

使用可能な ID 番号の範囲は常に SSSD に割り当てられます。これはローカルの範囲であり、この範囲はすべてのマシンについて同一です。デフォルトで、この範囲は 10,000 セクションに分割され、各セクションには 200,000 ID が割り当てられます。

新規の Active Directory ドメインが検出されると、SID はハッシュ化されます。次に SSSD は Active Directory ドメインに割り当てる ID セクションを判別するために、ハッシュのモジュラスと使用できるセクションの数を取ります。これは、同じ ID 範囲をすべてのクライアントマシンの同じ Active Directory ドメインに割り当てられるよう ID セクションを割り当てる一貫した方法です。

```
|    AD      |    AD      |          |
| domain 1   | domain 2   |   ...    |
|____________|____________|__________|
|  slice 1   |  slice 2   |   ...    |
min ID                          max ID
```

注記

すべてのクライアントが ID マッピングに SSSD を使用する限り、マッピングの整合性は保たれます。ただし、一部のクライアントが異なるソフトウェアを使用する場合、同じマッピングアルゴリズムが使用されていることを確認するか、または明示的な POSIX 属性を使用してください。

ID マッピングパラメーター

ID マッピングは、デフォルトで AD プロバイダーで有効にされます。**ldap_id_mapping** パラメーターはマッピングを有効にし、**ldap_schema** パラメーターはどの LDAP 属性をどの SSSD 属性にマップするかを設定します。

注記

ID マッピングが有効にされると、***uidNumber*** および ***gidNumber*** 属性は無視されます。これにより、すべての手動で割り当てられる値の使用が回避されます。**いずれかの** 値を手動で割り当てられなければならない場合は、**すべての** 値を手動で割り当てる必要があり、その場合は ID マッピングを無効にする必要があります。

ユーザーのマッピング

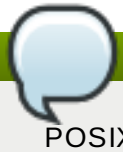
Active Directory ユーザーによるローカルシステムへのログイン試行の初回時に、そのユーザーのエントリーは SSSD キャッシュに作成されます。リモートユーザーはシステムユーザーと同様にセットアップされます。

- ユーザーのシステム UID は、そのユーザーの SID および該当するドメインの ID 範囲に基づいて作成されます。
- UID と同一のユーザーの GID が作成されます。
- シェル属性が SSSD 設定に基づいて使用されます。
- ユーザーが Active Directory ドメインの任意のグループに属する場合、SSSD はユーザーを Linux システム上のそれらのグループに追加するために SID を使用します。

2.3.1.2. SSSD および POSIX 属性について

Active Directory は、***uidNumber***、***gidNumber***、***unixHomeDirectory*** および ***loginShell*** などの POSIX 属性を作成し、保存するために設定できます。すべてのユーザー属性と同様に、それらは元々はローカルドメインに保存されますが、グローバルカタログに複製することもできます。それらがグローバルカタログに入ると、SSSD や ID 情報を取得するために SSSD を使用するすべてのアプリケーションでこれらを使用できるようになります。

属性の複製はパフォーマンス面で利点がありますが、必須ではありません。SSSD はPOSIX 属性があるかどうかを検出しようとしますが、これらがない場合には SSSD は POSIX 属性のグローバルカタログへの複製を要求する代わりに、LDAP ポート上で個々のドメインコントローラーに直接接続します。



注記

POSIX 属性がサーバー上で定義されている場合でも、ID マッピングを使用することができることに注意してください。この場合、SSSD は POSIX 属性を無視します。

パフォーマンスの最大化のために既存の POSIX 属性を使用するには、以下を確認します。

- POSIX 属性を Active Directory のグローバルカタログに公開する
- Active Directory ドメインエントリーに **`ldap_id_mapping = False`** を設定して SSSD の ID マッピングを無効にする

2.3.1.3. SSSD による CIFS 共有へのアクセス

SSSD では Windows セキュリティー ID (SID) と POSIX ID 間の ID マッピングを処理することができます。したがって、SSSD クライアントは Common Internet File System (CIFS) 共有にアクセスでき、これを制限なく使用できます。

注記

適切なアクセス制御を行うために CIFS 共有を使用するには、Windows SID を Linux POSIX UID および GID に変換する必要があります。以前は Winbind のみがこの機能を提供していましたが、現在 SSSD クライアントではこの実行において SSSD と共に Winbind を実行する必要がなくなりました。

CIFS ファイル共有プロトコルは Windows マシンで幅広く導入されています。SSSD は、Identity Management と Active Directory 間の信頼のある環境で、標準 Linux ファイルシステムの場合と同様の CIFS のシームレスな使用を可能にします。SSSD クライアントがシステム識別情報に使用する SID から ID または SID から名前のアルゴリズムは CIFS 共有でも使用できるようになりました。たとえば、アクセス制御リスト (ACL) にアクセスする際には、SSSD と並行して Winbind を実行する必要がなくなりました。

CIFS 共有にアクセスするには、Winbind ではなく SSSD を使用することをお勧めします。IdM クライアントは、AD ユーザーを UNIX ユーザーにマップするためにデフォルトで SSSD を使用します。CIFS マッピングに SSSD を使用することにより、IdM クライアントが Winbind を使用した場合に生じる可能性のある不整合なマッピングを防ぐことができます。クライアントが AD ドメインに直接参加する AD に直接統合された環境において、Linux クライアントが一般的な AD ユーザーマッピングのために Winbind ではなく SSSD を使用する場合、クライアントも CIFS のマッピングサービスとして SSSD を使用する必要があります。

サーバー側では、SSSD は SID と POSIX ID のマッピングを有効にし、クライアントに CIFS 共有へのアクセスを提供します。ただし、サーバー側の Winbind は依然として NT LAN Manager (NTLM) または NetBIOS 名参照を使用した認証のサポートなど、SSSD よりも多くのサービスを提供します。しかし IdM ドメインでは Kerberos 認証と DNS 名参照が利用できるため、この違いが IdM クライアントについて何らかの支障になることはありません。

現在 CIFS へのアクセスに SSSD を使用しているか、または Winbind を使用しているかを確認するには、**alternatives** ツールを使用します。

```
# alternatives --display cifs-idmap-plugin
cifs-idmap-plugin - status is auto.
 link currently points to /usr/lib/cifs-utils/cifs_idmap_sss.so
/usr/lib/cifs-utils/cifs_idmap_sss.so - priority 20
/usr/lib/cifs-utils/idmapwb.so - priority 10
Current `best' version is /usr/lib/cifs-utils/cifs_idmap_sss.so.
```

SSSD プラグイン (**cifs_idmap_sss.so**) がインストールされている場合、このプラグインはデフォルトで Winbind プラグイン (**idmapwb.so**) よりも優先されます。

別のプラグインに切り替えるには、**alternatives --set cifs-idmap-plugin** コマンドを実行し、プラグインへのパスを指定します。たとえば、Winbind に切り替えるには、以下を実行します。

```
# alternatives --set cifs-idmap-plugin /usr/lib/cifs-utils/idmapwb.so
```

重要

IdM クライアントでは常に SSSD プラグインを使用することをお勧めします。

Winbind プラグインに切り替える必要がある場合、Winbind がシステム上で実行中であることを確認します。同様に SSSD に再度切り替える場合も SSSD が実行中であることを確認します。

2.3.1.4. Active Directory ユーザーおよび範囲取得検索

Microsoft Active Directory には ***MaxValRange*** の属性があり、これは、複数値属性の返される値の上限を設定します。これは **範囲取得** 検索の拡張機能です。基本的には複数のミニ検索を実行し、すべての一致が返されるまでそれぞれの検索は特定の範囲内での結果のサブセットを返します。

たとえば、***member*** 属性を検索する際に、各エントリーに複数の値があり、その属性に複数のエントリーがある場合があります。1500 件 以上の一致結果があった場合、***MaxValRange*** は一度に表示される数を制限します。特定の属性には新たなフラグセットが付けられ、その結果がセット内のどの範囲にあるかを示します。

```
attribute:range=low-high:value
```

たとえば、検索結果を 100 件から 500 件にするには、以下のようにします。

```
member:range=99-499: cn=John Smith...
```

SSSD は、Active Directory プロバイダーによる範囲検索をユーザーおよびグループ管理の一部としてサポートします。この使用に追加の設定は不要です。

SSSD の検索ベースがカスタムフィルターまたはスコープを指定する場合、検索の設定に利用できる一部の LDAP プロバイダー属性 (***ldap_user_search_base*** など) は範囲取得と共に使用できません。Active Directory プロバイダードメインで検索ベースを設定する際には、どの検索が範囲取得をトリガーするかに留意してください。

2.3.2. Linux クライアントおよび Active Directory DNS サイト

SSSD はローカル Linux システムをより大規模な Active Directory 環境に接続します。これには、Linux クライアントが正常に統合されるように、SSSD が Active Directory フォレスト内の利用可能な設定を認識し、それらの設定で機能できるようにする必要があります。

Active Directory フォレストは、数多くの異なるドメインコントローラー、ドメインおよびサブドメイン、および物理サイトを含む非常に大きなフォレストである場合があります。クライアントのパフォーマンスを向上させるために、Active Directory は特殊なタイプの DNS レコードを使用して同じドメイン内の物理的に異なる場所にあるドメインコントローラーを識別します。クライアントは最も近くにあるドメインコントローラーに接続されます。

Active Directory は通常の DNS SRV レコードを拡張し、ドメインコントローラーの特定の物理的な場所またはサイトを識別します。SSSD などのクライアントは、独自のサイト設定に基づいて使用するドメインコントローラーを判別します。SSSD は、最初に Active Directory ドメインでサイト設定を、次にドメインコントローラーの DNS レコードを照会することにより、使用するドメインコントローラーを判別することができます。

1. SSSD は Active Directory ドメインに接続し、通常の DNS 検出で利用可能なドメインコントローラーの検索を試行します。
2. SSSD は、DNS ドメイン、ドメイン SID およびバージョンを検索するドメインコントローラーに LDAP 検索を送信します。

   ```
   (&(&(DnsDomain=ad.domain)(DomainSid=S-1-5-21-1111-2222-3333))
   (NtVer=0x01000016))
   ```

   これは、クライアントのサイトが設定されている場合にその情報を取得するために使用されます。
3. サイトがクライアント用に設定されている場合、返信にはサイト名 (**site-name._sites.**) などのプライマリーサーバーの拡張された DNS SRV レコードが含まれます。

   ```
   _tcp._ldap.site-name._sites.domain.name
   ```

   バックアップサーバーのレコードも、標準 SRV レコードとして送信されます。

```
_tcp._ldap.domain.name
```

サイトが設定されていない場合、標準 SRV レコードがすべてのプライマリーおよびバックアップサーバー用に送信されます。

4. SSSD はプライマリーサーバーおよびフォールバックサーバーの一覧を取得します。

2.4. ID マッピングを使用した Active Directory ドメインの設定

Active Directory ドメインの設定における最も簡単な設定は、すべてのプロバイダー (アイデンティティー、アクセス、パスワード) に **ad** 値を使用する方法です。さらにデフォルトの RFC 2307 を使用する代わりに、ユーザーおよびグループエントリーのネイティブの Active Directory スキーマをロードできます。

他の設定も、一般的な LDAP プロバイダー設定 [1] および Active Directory 固有の設定で利用できます [2]。これらには、特定のユーザーまたはグループサブツリーの LDAP フィルターの設定、認証のフィルター、および特定アカウント設定値が含まれます。一部の追加設定については、「追加の設定例」で説明されています。

注記

以下の手順は Active Directory ドメインの手動設定の説明であることに注意してください。**realmd** を使用する場合は、以下のステップ 3 から 7 については **realm join** コマンドで自動的に実行できます。詳細は、3章*realmd を使用した Active Directory ドメインへの接続*を参照してください。

1. Active Directory および Linux システムの両方に環境が適切に設定されていることを確認してください。
   - サービス検出が SSSD で使用される場合はとくに、名前解決が適切に設定されている必要があります。
   - Kerberos が適切に機能するには、両方のシステムの時計が同期している必要があります。
2. Linux クライアントでは、Active Directory ドメインをクライアントの DNS 設定に追加し、ドメインの SRV レコードを解決できるようにします。

```
search adserver.example.com
nameserver 198.68.72.1
```

3. Linux システムを Active Directory クライアントとしてセットアップし、Active Directory ドメイン内にこれを登録します。これは Linux システムに Kerberos および Samba サービスを設定して実行されます。
   a. Kerberos をセットアップし、Active Directory Kerberos レルムを使用します。
      a. Kerberos クライアント設定ファイルを開きます。

```
[root@server ~]# vim /etc/krb5.conf
```

      b. **[logging]** および **[libdefaults]** セクションを設定し、それらが Active Directory レルムに接続されるようにします。

```
[logging]
 default = FILE:/var/log/krb5libs.log

[libdefaults]
 default_realm = EXAMPLE.COM
 dns_lookup_realm = true
 dns_lookup_kdc = true
 ticket_lifetime = 24h
 renew_lifetime = 7d
 rdns = false
 forwardable = yes
```

自動検出が SSSD で使用されない場合、**[realms]** および **[domain_realm]** セクションも設定し、Active Directory サーバーを明示的に定義します。

b. Active Directory サーバーに接続するように Samba サーバーを設定します。

a. Samba 設定ファイルを開きます。

```
[root@server ~]# vim /etc/samba/smb.conf
```

b. Active Directory ドメイン情報を **[global]** セクションに設定します。

```
[global]
   workgroup = EXAMPLE
   client signing = yes
   client use spnego = yes
   kerberos method = secrets and keytab
   log file = /var/log/samba/%m.log
   password server = AD.EXAMPLE.COM
   realm = EXAMPLE.COM
   security = ads
```

c. Linux マシンを Active Directory ドメインに追加します。

a. Windows 管理ユーザーの Kerberos 資格情報を取得します。

```
[root@server ~]# kinit Administrator
```

b. **net** コマンドを使用して、マシンをドメインに追加します。

```
[root@server ~]# net ads join -k
Joined 'server' to dns domain 'example.com'
```

これにより、新規のキータブファイル **/etc/krb5.keytab** が作成されます。

システムのキーを一覧表示し、ホストのプリンシパルがあることを確認します。

```
[root@server ~]# klist -k
```

4. 必要な場合は **oddjob-mkhomedir** パッケージをインストールし、SSSD が Active Directory ユーザーのホームディレクトリーを作成できるようにします。

```
[root@server ~]# yum install oddjob-mkhomedir
```

5. **authconfig** を使用してシステム認証のために SSSD を有効にします。**--enablemkhomedir** を使用して SSSD でホームディレクトリーを作成できるようにします。

```
[root@server ~]# authconfig --update --enablesssd --enablesssdauth --enablemkhomedir
```

6. SSSD 設定ファイルを開きます。

```
[root@rhel-server ~]# vim /etc/sssd/sssd.conf
```

7. Active Directory ドメインを設定します。

   a. **[sssd]** セクションで、アクティブなドメインの一覧に Active Directory ドメインを追加します。これは SSSD 設定ファイルの **[domain/NAME]** に設定されるドメインエントリーの名前になります。

   さらに **pac** をサービスの一覧に追加します。これにより、SSSD を使用して Active Directory ドメインとの通信に使用されるチケットの MS-PAC 情報を設定し、使用することができます。

```
[sssd]
config_file_version = 2
domains = ad.example.com
services = nss, pam, pac
```

   b. Active Directory ドメインのファイルの下部に新規のドメインセクションを作成します。このセクションには **domain/ad.example.com** などの **domain/NAME** 形式が使用されます。それぞれのプロバイダーについては、値を **ad** に設定し、接続する特定 Active Directory インスタンスの接続情報を指定します。

```
[domain/ad.example.com]
id_provider = ad
auth_provider = ad
chpass_provider = ad
access_provider = ad
```

   c. 資格情報のキャッシュを有効にします。これにより、ユーザーは Active Directory ドメインが使用できない場合でも、キャッシュされた情報を使用してローカルシステムにログインできます。

```
cache_credentials = true
```

8. SSH サービスを再起動して、新規の PAM 設定をロードします。

```
[root@server ~]# systemctl restart sshd.service
```

9. 設定ファイルを変更した後に SSSD を再起動します。

```
[root@rhel-server ~]# systemctl restart sssd.service
```

2.5. POSIX 属性を使用した Active Directory ドメインの設定

SSSD で Active Directory 定義の POSIX 属性を使用する際には、それらの属性をグローバルカタログに複

製してパフォーマンスの向上を図ることが推奨されています。それらの属性はグローバルカタログに入ると、SSSD、およびアイデンティティー情報に SSSD を使用するアプリケーションで利用できるようになります。さらに POSIX 属性が使用される場合、新規設定をローカルに作成するのではなく、POSIX属性が Active Directory から使用されるよう ID マッピングを SSSD で無効にする必要があります。

他の設定も、一般的な LDAP プロバイダー設定 [3] および Active Directory 固有の設定で利用できます [4]。これらには、特定のユーザーまたはグループサブツリーの LDAP フィルターの設定、認証のフィルター、および特定アカウント設定値が含まれます。一部の追加設定については、「追加の設定例」 で説明されています。

注記

以下の手順は Active Directory ドメインの手動設定の説明であることに注意してください。**realmd** を使用する場合は、以下のステップ 4 から 11 については **realm join** コマンドで自動的に実行できます。詳細は、3章*realmd を使用した Active Directory ドメインへの接続*を参照してください。

1. Active Directory および Linux システムの両方に環境が適切に設定されていることを確認してください。
   - サービス検出が SSSD で使用される場合はとくに、名前解決が適切に設定されている必要があります。
   - Kerberos が適切に機能するには、両方のシステムの時計が同期している必要があります。
2. Active Directory ドメインで、POSIX 属性がグローバルカタログに複製されるように設定します。
   a. すべてのプライマリーおよび子ドメインコントローラーに **Identity Management for UNIX Components** をインストールします。これにより、POSIX 属性および関連するスキーマがユーザーアカウントで利用可能になります。これらの属性は、**Properties** メニューの **UNIX Attributes** タブで利用できます。
   b. Active Directory スキーマスナップインをインストールし、グローバルカタログに複製される属性を追加します。
   c. 関連する POSIX 属性 (***uidNumber***、***gidNumber***、***unixHomeDirectory***、および ***loginShell***) について、**Properties** メニューを開き、**Replicate this attribute to the Global Catalog** チェックボックスを選択してから **OK** をクリックします。
3. Linux クライアントでは、Active Directory ドメインをクライアントの DNS 設定に追加し、ドメインの SRV レコードを解決できるようにします。

```
search adserver.example.com
nameserver 198.68.72.1
```

4. Linux システムを Active Directory クライアントとしてセットアップし、Active Directory ドメイン内にこれを登録します。これは Linux システムに Kerberos および Samba サービスを設定して実行されます。
   a. Kerberos をセットアップし、Active Directory Kerberos レルムを使用します。
      a. Kerberos クライアント設定ファイルを開きます。

```
[root@server ~]# vim /etc/krb5.conf
```

b. **[logging]** および **[libdefaults]** セクションを設定し、それらが Active Directory レルムに接続されるようにします。

```
[logging]
 default = FILE:/var/log/krb5libs.log

[libdefaults]
 default_realm = EXAMPLE.COM
 dns_lookup_realm = true
 dns_lookup_kdc = true
 ticket_lifetime = 24h
 renew_lifetime = 7d
 rdns = false
 forwardable = yes
```

自動検出が SSSD で使用されない場合、**[realms]** および **[domain_realm]** セクションも設定し、Active Directory サーバーを明示的に定義します。

b. Active Directory サーバーに接続するように Samba サーバーを設定します。

a. Samba 設定ファイルを開きます。

```
[root@server ~]# vim /etc/samba/smb.conf
```

b. Active Directory ドメイン情報を **[global]** セクションに設定します。

```
[global]
   workgroup = EXAMPLE
   client signing = yes
   client use spnego = yes
   kerberos method = secrets and keytab
   log file = /var/log/samba/%m.log
   password server = AD.EXAMPLE.COM
   realm = EXAMPLE.COM
   security = ads
```

c. Linux マシンを Active Directory ドメインに追加します。

a. Windows 管理ユーザーの Kerberos 資格情報を取得します。

```
[root@server ~]# kinit Administrator
```

b. **net** コマンドを使用して、マシンをドメインに追加します。

```
[root@server ~]# net ads join -k
Joined 'server' to dns domain 'example.com'
```

これにより、新規のキータブファイル **/etc/krb5.keytab** が作成されます。

c. システムのキーを一覧表示し、ホストのプリンシパルがあることを確認します。

```
[root@server ~]# klist -ke
```

d. ユーザーが **ldapsearch** を使用してグローバルカタログを検索できるかどうかをテストします。

```
[root@server ~]# ldapsearch -H
ldap://server.ad.example.com:3268 -Y GSSAPI -N -b
"dc=ad,dc=example,dc=com" "(&(objectClass=user)
(sAMAccountName=aduser))"
```

5. **sssd-ad** パッケージをインストールします。

```
[root@server ~]# yum install sssd-ad
```

6. SSSD サービスを起動します。

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

7. SSSD 設定ファイルを開きます。

```
[root@rhel-server ~]# vim /etc/sssd/sssd.conf
```

8. Active Directory ドメインを設定します。

   a. **[sssd]** セクションで、アクティブなドメインの一覧に Active Directory ドメインを追加します。これは SSSD 設定ファイルの **[domain/NAME]** に設定されるドメインエントリーの名前になります。

   b. Active Directory ドメインのファイルの下部に新規のドメインセクションを作成します。このセクションには **domain/ad.example.com** などの **domain/NAME** 形式が使用されます。それぞれのプロバイダーについては、値を **ad** に設定し、接続する特定 Active Directory インスタンスの接続情報を指定します。

```
[domain/ad.example.com]
id_provider = ad
auth_provider = ad
chpass_provider = ad
access_provider = ad
```

   c. ID マッピングを無効にします。これにより、SSSD は Windows SID に基づいて **UID:GID** 番号を作成するのではなく、グローバルカタログで POSIX 属性を検索するように指示されます。

```
# disabling ID mapping
ldap_id_mapping = False
```

   d. ホームディレクトリーおよびログインシェルがユーザーアカウントで設定される場合、テンプレートに基づいて属性を作成するのではなく POSIX 属性を使用するよう SSSD を設定するため、以下の行をコメントアウトします。

```
# Comment out if the users have the shell and home dir set on
the AD side
#default_shell = /bin/bash
#fallback_homedir = /home/%d/%u
```

   e. Active Directory ユーザーの短縮名または完全修飾ユーザー名を使用するかどうかを設定します。複雑なトポロジーでは、曖昧さを避けるために完全修飾名の使用が必要になる場合があります。

```
# Comment out if you prefer to user shortnames.
use_fully_qualified_names = True
```

f. 資格情報のキャッシュを有効にします。これにより、ユーザーは Active Directory ドメインが使用できない場合でも、キャッシュされた情報を使用してローカルシステムにログインできます。

```
cache_credentials = true
```

9. SSSD 設定ファイルのファイル権限および所有者を設定します。

```
[root@server ~]# chown root:root /etc/sssd/sssd.conf
[root@server ~]# chmod 0600 /etc/sssd/sssd.conf
[root@server ~]# restorecon /etc/sssd/sssd.conf
```

10. 必要な場合は **oddjob-mkhomedir** パッケージをインストールし、SSSD が Active Directory ユーザーのホームディレクトリーを作成できるようにします。

```
[root@server ~]# yum install oddjob-mkhomedir
```

11. **authconfig** を使用してシステム認証のために SSSD を有効にします。**--enablemkhomedir** を使用して SSSD でホームディレクトリーを作成できるようにします。

```
[root@server ~]# authconfig --update --enablesssd --enablesssdauth --
enablemkhomedir
```

12. SSH サービスを再起動して、新規の PAM 設定をロードします。

```
[root@server ~]# systemctl restart sshd.service
```

13. 設定ファイルを変更した後に SSSD を再起動します。

```
[root@rhel-server ~]# systemctl restart sssd.service
```

authconfig を使用すると、アイデンティティーソースとして SSSD を使用するように NSS および PAM 設定ファイルが自動的に設定されます。

たとえば **nsswitch.conf** ファイルには、ユーザー、グループ、およびサービス情報のソースとして SSSD (**sss**) が追加されます。

```
passwd:         files sss
group:          files sss
...
services:       files sss
...
netgroup:       files sss
```

異なる **pam.d** ファイルが、**/etc/pam.d/system-auth** および **/etc/pam.d/password-auth** ファイルの各 **pam_unix.so** 行の下に **pam_sss.so** モジュールの行を追加します。

```
auth         sufficient    pam_sss.so use_first_pass
...
account      [default=bad success=ok user_unknown=ignore] pam_sss.so
```

```
...
password     sufficient    pam_sss.so use_authtok
...
session      optional      pam_mkhomedir.so
session      optional      pam_sss.so
```

2.6. 追加の設定例

2.6.1. アカウント設定

Linux ユーザーの場合、特定のシステム設定が新規ユーザー用にデフォルトで設定されます。これらのシステム設定は、Windows ユーザーアカウントで設定されないか、または Linux システムと互換性のないシステムに設定される可能性があります。これらには、ユーザーホームディレクトリーおよびデフォルトユーザーシェルの 2 分野での設定が含まれます。

2.6.1.1. ユーザーホームディレクトリーの設定

Red Hat Enterprise Linux には PAM ライブラリー (**pam_oddjob_mkhomedir.so**) があります。このライブラリーは、ユーザーの初回ログイン時にユーザーディレクトリーを自動的に作成します。これには、ユーザーの Linux システムへの初回ログイン時の Active Directory ユーザーも含まれます。

SSSD を使用すると、ユーザーディレクトリーの形式はアイデンティティープロバイダーから取得されます。アイデンティティープロバイダーに Linux システムの形式とは異なるホームディレクトリーがある場合や、プロバイダーが値を指定しない場合は、設定に指定されるテンプレートを使用してホームディレクトリーの属性値を設定するよう SSSD を設定することができます。このテンプレートは、NSS サービスセクションにグローバルに設定するか、またはドメインごとに設定することができます。以下の 2 つのパラメーターを使用できます。

- ***fallback_homedir***: アイデンティティープロバイダーによる指定がない場合はテンプレートを指定します。
- ***override_homedir***: アイデンティティープロバイダーに設定される情報にかかわらず、使用するテンプレートを設定します。

上記のいずれの場合も、ログイン名の **%u** およびドメイン名の**%d** などのテンプレート内の変数を使用できます。

```
[nss]
fallback_homedir = /home/%u
...
[domain/AD_EXAMPLE]
id_provider = ad
auth_provider = ad
...
override_homedir = /home/%d/%u
```

2.6.1.2. ユーザーシェルの設定

デフォルトで、SSSD はアイデンティティープロバイダーからユーザーシェルについての情報を取得するよう試行します。Active Directory および LDAPv3 スキーマのいずれでも、これは ***loginShell*** 属性で定義されます。ただし、これはオプション属性のため、すべてのユーザーに定義されない可能性があります。Active Directory ユーザーの場合は、定義されたログインシェルが Linux システムで許可されない可能性があります。

SSSD 設定のシェルを処理するには数多くの方法があります。

- シェルが指定されていない場合は、***shell_fallback*** を使用してフォールバック値を使用します。
- ***allowed_shells*** および ***vetoed_shells*** を使用して許可されたシェル、またはブラックリストに入れられたシェルの一覧を設定します。
- ***default_shell*** を使用してデフォルト値を設定します。
- 別の値がアイデンティティープロバイダーに指定されている場合でも、***override_shell*** を使用して使用する値を設定します。

注記

allowed_shells、**vetoed_shells**、および **shell_fallback** パラメーターは、ドメインごとにではなく、グローバル設定としてのみ設定できます。ただし、これらのパラメーターはローカルシステムのユーザーには影響を与えず、SSSD アイデンティティープロバイダーで取得される外部ユーザーにのみ影響を与えます。**/bin/rbash** などの一般的な設定の使用はほとんどの外部ユーザーの場合に役に立ちます。

デフォルト値はドメインごとに設定できますが、シェルのホワイトリストおよびブラックリストなどの一部の値は NSS サービス設定でグローバルに設定する必要があります。以下が例になります。

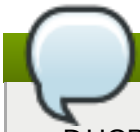
```
[nss]
shell_fallback = /bin/sh
allowed_shells =  /bin/sh,/bin/rbash,/bin/bash
vetoed_shells =  /bin/ksh
...

[domain/AD_EXAMPLE]
id_provider = ad
auth_provider = ad
...
default_shell = /bin/rbash
```

2.6.2. 動的 DNS 更新の有効化 (Active Directory のみ)

Active Directory は、そのクライアントが DNS レコードを自動的に更新することを許可します。さらに、Active Directory は DNS レコードをアクティブに維持し、非アクティブなレコードのタイムアウト (エイジング) および削除 (清掃) などを実行し、これらのレコードの更新状態を維持できます。DNS の清掃機能については AD 側ではデフォルトで有効にされないことに注意してください。

SSSD は、DNS レコードを更新して Linux システムが Windows クライアントを模倣できるようにします。さらにレコードが非アクティブとマークされて DNS レコードから削除されることを防ぎます。動的 DNS 更新が有効にされると、クライアントの DNS レコードが以下のタイミングで複数回更新されます。

- アイデンティティープロバイダーがオンラインになる時点 (常時)
- Linux システムが再起動する時点 (常時)
- 指定された間隔 (オプションの設定)



注記

DHCP リースと同じ間隔を設定することができます。この場合、Linux クライアントはリースの更新後に更新されます。

DNS 更新は、DNS の Kerberos/GSSAPI (GSS-TSIG) を使用して Active Directory サーバーに送信されます。これはセキュアな接続のみを有効にする必要があることを意味します。

動的 DNS 設定は各ドメインに対して設定されます。以下が例になります。

```
[domain/ad.example.com]
id_provider = ad
auth_provider = ad
chpass_provider = ad
access_provider = ad

ldap_schema = ad

dyndns_update = true
dyndns_refresh_interval = 43200
dyndns_update_ptr = true
dyndns_ttl = 3600
```

表2.1 動的 DNS 更新のオプション

オプション	説明	形式
dyndns_update	DNS サーバーをクライアント IP アドレスで動的に更新するかどうかを設定します。これには安全な更新が必要であり、その他の動的 DNS 設定が有効になるように **true** に設定する必要があります。デフォルト値は **true** です。	ブール値
dyndns_ttl	クライアントの DNS レコードの有効時間を設定します。デフォルト値は 3600 秒です。	整数
dyndns_refresh_interval	プロバイダーがオンラインになる際の更新のほかに、自動的な DNS 更新を実行する頻度を設定します。デフォルト値は 86400 秒 (24 時間) です。	整数
dyndns_update_ptr	クライアントがその DNS レコードを更新する際に PTR レコードを更新するかどうかを設定します。デフォルト値は **true** です。	ブール値

2.6.3. アクセス制御でのフィルターの使用

Active Directory アクセスプロバイダーは認可情報のソースとして使用されます。以下の設定パラメーターオプションは他のいくつかの汎用 LDAP パラメーターの組み合わせです。

```
access_provider = ad
```

これは以下の LDAP パラメーターを設定する場合と同じになります。

```
access_provider = ldap
ldap_access_order = expire
ldap_account_expire_policy = ad
```

LDAP フィルターを使用して、アクセスを付与するユーザーアカウントを識別するための追加のオプションがあります。まずアカウントはフィルターに一致しており、それらは ***access_provider = ad*** 設定で暗黙的に設定されている期限チェックをパスしている必要があります。たとえば、以下は管理者グループに属し、***unixHomeDirectory*** 属性を持つユーザーのみがアクセス制御チェックに一致するものとして設定しています。

```
access_provider = ad
ad_access_filter = (&(memberOf=cn=admins,ou=groups,dc=example,dc=com)
(unixHomeDirectory=*))
```

[1] **sssd-ldap** man ページを参照してください。

[2] **sssd-ad** man ページを参照してください。

[3] **sssd-ldap** man ページを参照してください。

[4] **sssd-ad** man ページを参照してください。

第3章 realmd を使用した Active Directory ドメインへの接続

realmd システムは、アイデンティティードメインを検出し、このドメインに参加するための明確で簡単な方法を提供します。このシステム自体がドメインに接続することはありませんが、SSSD や Winbind といった基礎となる Linux システムサービスがドメインに接続できるように設定します。

3.1. realmd について

*2章Active Directory を SSSD のアイデンティティープロバイダーとして使用する*では、ローカルシステムおよびバックエンドのアイデンティティープロバイダーとしての Active Directory でシステムセキュリティーサービスデーモン (SSSD) を使用する方法を説明しています。それぞれの使用可能なアイデンティティープロバイダーおよび SSSD 自体には数多くの異なる設定パラメーターがあります。すべてのドメイン情報は事前に利用可能にしておく必要があり、その後に SSSD がローカルシステムを Active Directory に統合できるよう SSSD 設定で適切にフォーマットされる必要があります。これは複雑なタスクになり得ますが、**realmd** はその設定を単純化します。これは利用可能な Active Directory および Red Hat Enterprise Linux Identity Management ドメインを識別するためにサービス検出を実行し、ドメインに参加してユーザーアクセスを管理します。SSSD は基礎となるサービスとして複数のドメインをサポートするため、**realmd** も複数のドメインを検出し、サポートすることができます。

3.1.1. ドメインのタイプ

realmd システムは、以下のタイプのアイデンティティードメインを検出します。

- Microsoft Active Directory
- Red Hat Enterprise Linux Identity Management

realmd を使用して Active Directory または Identity Management ドメインに参加することができます。**realmd** は、必要なシステム設定ファイルおよびサービスを適切に設定します。

3.1.2. サポートされるドメインクライアント

realmd システムは、所定のアイデンティティーレルムに接続するために使用される必要のあるクライアントサービスを自動的に設定します。サポートされるクライアントには以下の 2 つがあります。

- Red Hat Enterprise Linux Identity Management と Microsoft Active Directory 用の SSSD
- Microsoft Active Directory 用の Winbind

3.2. realmd コマンド

realmd システムには、ドメインにおけるシステム登録管理と、ローカルシステムリソースへのアクセスを許可するドメインユーザーの設定という 2 つの主なタスク分野があります。**realmd** の中央ユーティリティーは **realm** と呼ばれ、このユーティリティーは主としてアクションおよびアクションの実行対象であるレルムを指定します。

```
realm command arguments
```

例:

```
realm join ad.example.com
realm permit username
```

表3.1 realmd コマンド

コマンド	説明
Realm コマンド	
discover	ネットワーク上のドメインの検出スキャンを実行します。
join	システムを指定されたドメインに追加します。
leave	指定されたドメインからシステムを削除します。
list	システム用に設定されたすべてのレルム、または検出され、設定されたすべてのレルムを一覧表示します。
ログインコマンド	
permit	設定されたレルム内の指定されたユーザーまたはすべてのユーザーがローカルシステムにアクセスできるようアクセスを有効にします。
deny	設定されたレルム内の指定ユーザーまたはすべてのユーザーがローカルシステムにアクセスする際のアクセスを制限します。

3.3. Active Directory ドメインの検出およびドメインへの参加

3.3.1. ドメインの検出

検出プロセスは **discover** コマンドで処理されます。このコマンドは、詳細なレルム設定と、システムのレルムへの登録のためにインストールする必要のあるパッケージの一覧を返します。

注記

realm discover コマンドの使用にあたっては NetworkManager が実行中である必要があることに注意してください。このコマンドはとくに NetworkManager の D-Bus インターフェースに依存しています。システムが NetworkManager を使用しない場合は、たとえば **realm discover ad.example.com** のようにコマンドのレルム名を指定します。

```
[root@server ~]# realm discover
ad.example.com
  type: active-directory
  realm-name: AD.EXAMPLE.COM
  domain-name: ad.example.com
  configured: kerberos-member
  server-software: active-directory
  client-software: sssd
  required-package: oddjob
  required-package: oddjob-mkhomedir
  required-package: sssd
  login-formats: %D\%U
  login-policy: allow-realm-logins
```

realmd は Active Directory ドメインおよび Identity Management ドメインの両方を検出できます。これらがどちらも環境にある場合は、検出結果を特定のサーバータイプに制限することができます。

```
[root@server ~]# realm discover --server-software=active-directory
```

また、ドメインコントローラーのホスト名または IP アドレスを使用して、特定ドメインの検出を実行することもできます。

```
[root@server ~]# realm discover ad.example.com
```

3.3.2. Active Directory ドメインへの参加

join コマンドにはレルム名のみが必要になります。

```
[root@server ~]# realm join ad.example.com
See: journalctl REALMD_OPERATION=r1088239.6316
realm: Joined ad.example.com domain
```

これはデフォルトの Windows 管理者として join を実行し、ほとんどの環境ではパスワードのプロンプトを出します。このコマンドでは、**-U** オプションを使用し、別のユーザーで Active Directory 環境に接続することもできます。

```
[root@server ~]# realm join ad.example.com -U AD.EXAMPLE.COM\jsmith
```

Kerberos が Linux システム上で適切に設定されている場合、join 操作は認証用の Kerberos チケットを使用して実行することもできます。**realmd** システムは **-U** オプションを使用して、使用するプリンシパルを選択したり、デフォルトの資格情報キャッシュまたは **KRB5_CCACHE** 変数を使用したりすることができます。

```
[root@server ~]# kinit jsmith
[root@server ~]# realm join ad.example.com -U jsmith
```

join の実行時に、**realmd** システムは DNS SRV レコードをチェックします。

```
_ldap._tcp.domain.example.com. // for IdM records
_ldap._tcp.dc._msdcs.domain.example.com. // for Active Directory records
```

DNS SRV レコードは Active Directory が設定される際にデフォルトで作成されます。これにより、サービス検出でレコードを検出することができます。**realmd** は、ネットワーク上の LDAP サーバーを検出するために DHCP で割り当てられたドメインを使用します。

実際の join コマンドは、以下のステップを実行して、ローカルシステムサービスと Active Directory ドメイン内のエントリーの両方を設定します。

1. 指定されたレルムについて検出スキャンを実行します。
2. システムをドメインに参加させるために必要なパッケージをインストールします。これには、SSSD および PAM ホームディレクトリーのジョブパッケージが含まれます。パッケージの自動インストールでは **PackageKit** スィートが実行中である必要があることに注意してください。

注記

PackageKit が無効な場合に、システムは足りないパッケージを尋ねるプロンプトを出します。それらのパッケージは、**yum** ユーティリティーを使用して手動でインストールする必要があります。

3. 異なるユーザーが **-U** オプションで指定されていない限り、Active Directory ドメインに管理者として参加することを試行します。コマンドは最初は資格情報なしに接続することを試行しますが、必要な場合はパスワードのプロンプトを出します。

注記

Active Directory では、管理者アカウントは **Administrator** であり、IdM では **admin** になります。

4. いったんドメインに接続されると、ディレクトリー内のシステムのアカウントエントリーが作成されます。
5. **/etc/krb5.keytab** ホストキータブファイルを作成します。
6. SSSD のドメインを設定し、サービスを再起動します。
7. PAM 設定および **/etc/nsswitch.conf** ファイルでシステムサービスのドメインユーザーを有効にします。

検出の検索で返される属性の 1 つは **login-policy** です。これは、ドメインユーザーが join の完了時にすぐにログインできるかどうかを示します。ログインがデフォルトで許可されない場合は、**permit** コマンドにより手動で許可することができます。詳細は、「Active Directory からのユーザーログイン管理」を参照してください。

3.3.3. Active Directory ドメインからのシステムの削除

万一システムを Active Directory ドメインから削除する必要がある場合、削除は **leave** コマンドで実行できます。これにより、SSSD およびローカルシステムからドメイン設定が削除されます。

```
[root@server ~]# realm leave ad.example.com
```

このコマンドは、デフォルトの管理者アカウント (アイデンティティー管理の admin、Active Directory の Administrator) として削除を実行します。スクリプトはパスワードのプロンプトを出すか、またはシステムがドメインに参加している方法によっては、異なるユーザーが操作を実行することを要求する可能性があります。ユーザーは、**-U** オプションで指定することができます。

```
[root@server ~]# realm leave ad.example.com -U AD.EXAMPLE.COM\jsmith
```

注記

クライアントがドメインから出ても、コンピューターオブジェクトは削除されません。ローカルクライアントの設定のみが解除されます。削除を実行する場合は、**--remove** オプションを指定してコマンドを実行します。

3.3.4. ドメインの一覧表示

list コマンドは、システムのすべての設定済みドメイン、およびそのドメインの詳細およびデフォルトの設定を一覧表示します。この内容は、すでにシステム設定内にあるドメインの場合のみレルム検出で返される情報と同じになります。

```
[root@server ~]# realm list
linux.example.com
  type: kerberos
  realm-name: LINUX.EXAMPLE.COM
  domain-name: linux.example.com
```

```
  configured: kerberos-member
  server-software: ipa
  client-software: sssd
  required-package: ipa-client
  required-package: oddjob
  required-package: oddjob-mkhomedir
  required-package: sssd
  login-formats: %U
  login-policy: allow-realm-logins
```

--all オプションには、設定されたドメインのほかに検出されたドメイン (Active Directory、Identity Management、および Kerberos) が含まれます。**--name-only** は、設定の詳細なしのドメイン名に結果を制限します。

```
[root@server ~]# realm list --all --name-only
linux.example.com
example.com
ad.example.com
```

3.4. Active Directory からのユーザーログイン管理

デフォルトでは、ドメインユーザーのログインポリシーはドメイン自体で定義されます。この設定は、クライアント側のアクセス制御が使用されるようにレルム設定で上書きすることができます。つまり、ローカルポリシーはログインできるユーザーのみを定義します。マシンが複数のドメインに参加する場合は、それらのドメインの 1 つのみがドメインアクセス制御を適用でき、他のドメインではクライアント側のアクセス制御を使用する必要があります。

realmd コマンドにより、基本的なアクセス許可またはアクセス拒否のルールを特定ドメインのユーザーに設定することができます。これらの権限は、クライアント側のアクセス制御を適用している場合にのみ指定できます。

注記

上記のアクセスルールは、システムへのすべてのアクセスを許可するか、または全くアクセスを許可しないかのいずれかになります。詳細なアクセスルールは、特定のシステムリソース上か、またはドメインに設定する必要があります。

アクセスルールを設定するコマンドには以下の 2 つがあります。

- **realm deny** コマンドは、単にレルム内のすべてのユーザーにアクセスが付与されないようにします。このコマンドは **--all** オプションと一緒に使用します。
- 一方 **realm permit** コマンドでは、**--all** を使用してすべてのユーザーにアクセスを付与するか、指定されたユーザーのみに付与するか、または **-x** を使用して指定されたユーザーの権限を取り消すことができます。

たとえば、以下のコマンドは **ad.example.com** 内のすべてのユーザーに対して許可ルールを追加してから、**jsmith** ユーザーのログイン権限を取り消します。

```
[root@server ~]# realm permit ad.example.com --all
[root@server ~]# realm permit ad.example.com -x AD.EXAMPLE.COM\jsmith
```

重要

permit -x でアクセスを拒否する代わりに、**permit** を使用してアクセスを許可することが推奨されます。一部のユーザーに対してアクセスを拒否し、それ以外のすべてのユーザーにアクセスを許可する代わりに、明確に選択したユーザーまたはグループにアクセスを許可する方がはるかに安全であると言えます。

現時点で SSSD は利用可能なサブドメインについて **realmd** に通知することができず、ユーザーログインにはドメイン名が含まれる必要があるため、アクセスの許可はプライマリードメインのユーザーに対しては可能ですが、信頼されるドメインのユーザーに対しては実行できません。

3.5. デフォルトユーザー設定の追加

/etc/realmd.conf 設定ファイルでは、グローバルなログインユーザー設定のカスタム設定を追加できます。一部の POSIX 属性は Windows ユーザーアカウントに設定できないか、またはローカルシステムの他のユーザーとは異なるものに設定される可能性があります。関連する分野として以下のような 2 つの分野があります。

- ユーザーホームディレクトリー
- デフォルトユーザーシェル

ユーザー設定は **/etc/realmd.conf** ファイルの **[users]** セクションで定義されます。

- ***default-shell*** パラメーターには、任意のサポートされているシステムシェルを使用できます。
- ***default-home*** パラメーターは、レルムに指定がない場合にホームディレクトリーの作成に使用するテンプレートを設定します。共通の形式は **/home/%d/%u** です。ここで、**%d** はドメイン名で、**%u** はユーザー名です。

例:

```
[users]
default-home = /home/%u
default-shell = /bin/bash
```

3.6. Active Directory ドメインエントリーの追加設定

それぞれの個別ドメインには、**/etc/realmd.conf** 設定ファイルのレルムエントリーにカスタム設定を加えることができます。各レルムには、独自の設定セクションを設けることができます。

```
[realm.name]
attribute = value
attribute = value
```

各属性は、それらを設定ファイルに手動で追加するか、またはシステムがレルムに参加する際にそれらを引数として渡すことによって設定できます。

表3.2 レルム設定オプション

パラメーター	説明

パラメーター	説明
computer-ou	コンピューターアカウントをドメインに追加するためのディレクトリーの場所を設定します。これは、root エントリーに関連する完全 DN または RDN にすることができます。サブツリーはすでに存在している必要があります。
user-principal	システムのホストプリンシパルを作成するかどうかを設定します。
automatic-id-mapping	動的 ID マッピングを有効にするか、またはマッピングを無効にして Active Directory で設定される POSIX 属性を使用するかどうかを設定します。
manage-system	特定のログインポリシーがローカルシステム内か、または Active Directory で設定されるかどうかを設定します。

以下の例は ID マッピングを無効にし、ホストプリンシパルを有効にし、かつシステムを指定されたサブツリーに追加します。

```
[domain.example.com]
computer-ou = OU=Linux Computers,DC=domain,DC=example,DC=com
user-principal = yes
automatic-id-mapping = no
manage-system = no
```

システムがドメインに参加する際に、これらの同じパラメーターを渡すことができます。

```
[root@server ~]# realm join --computer-ou="ou=Linux Computers," --
automatic-id-mapping=no --user-principal=yes
```

第4章 Samba、Kerberos、および Winbind の使用

複数ユーティリティーで構成される Samba の標準 Windows 相互運用機能スイートは、Linux システムを Windows クライアントとして表示させることにより、これらのシステムを Active Directory 環境に参加させることができます。システム統合の方法として、Samba は Linux クライアントを Active Directory Kerberos レルムに参加させ、その ID ストアとして Active Directory を使用することを可能にします。

Winbind は、統一されたログインを提供する Samba スィートのコンポーネントです。これは、Microsoft RPC コール、Pluggable Authentication Module (PAM)、および Name Service Switch (NSS) の UNIX 実装を使用して Windows ドメインユーザーが UNIX システム上の UNIX ユーザーとして表示され、機能するようにします。

4.1. Samba および Active Directory 認証について

Samba のコア機能は、ファイルおよびプリンター共有、および関連操作を実行するためにクライアントサーバーのネットワークを構築しますが、本章では、Samba を使用した Windows との対話の 1 つの側面、つまり Active Directory を使用して Linux クライアントを認証する点を重点的に説明します。

4.1.1. Samba、Kerberos および Active Directory ドメイン

Active Directory は、Kerberos レルムおよび DNS ドメインを含む、Windows 環境の数多くのサーバーのドメインコントローラーです。Samba は、Kerberos、DNS、NTLMSSP、または DCE/RPC を含む Active Directory で使用される幅広い範囲のプロトコルをサポートします。Active Directory の統合とは、Kerberos を Active Directory 内のネイティブセキュリティーコンテキストとして使用するセキュリティー環境を設定することを意味します。

Active Directory をドメインコントローラーとして使用するには、Linux クライアント上でいくつかの異なるシステムサービスを設定する必要があります。

- Samba: ユーザーおよび認証用
- DNS: Active Directory サーバーをネームサーバーとして設定
- Kerberos: Active Directory KDC の使用
- PAM: Winbind の使用
- NSS: Winbind の使用

4.1.1.1. Samba

Samba サーバーを Active Directory ドメインに参加させる方法には複数の方法があります。SMB/CIFS ネットワークでは、ユーザーレベルと共有レベルの 2 種類のセキュリティーがあります。Samba はユーザーレベルのセキュリティーを使用するための 4 つの方法を提供します。それらの方法は *セキュリティーモード* と総称されます。Windows の統合で重要になるのはこの内の 2 つの方法のみです。

- **`ads`** は、ローカル Samba サーバーを Active Directory ドメイン内のドメインメンバーとして設定します。さらに、LDAP クエリーおよび Kerberos 認証の内部使用のサポートも有効にします。これは優先されるセキュリティーモードです。
- **`domain`** は、DCE/RPC プロトコルを使用して、Samba サーバーを Active Directory ドメイン内のドメインメンバーサーバーとして設定します。

必要な設定は **`/etc/samba/smb.conf`** の **`[global]`** セクションに置かれます。必須の設定には、セキュリティータイプ (***`security`***)、DNS 検出で解決される Active Directory Kerberos レルムの名前 (***`realm`***)、および Samba ワークグループ (***`workgroup`***) が含まれます。

```
#================== Global Settings ====================

[global]
    workgroup = ADEXAMPLE
    security = ads
    realm = ADEXAMPLE.COM

...
```

4.1.1.2. Kerberos

Kerberos は Active Directory サーバーをその KDC として使用するように設定する必要があります。これにより、ユーザーは認証に Kerberos チケットを使用することができます。さらに Samba は、Winbind による Kerberos プリンシパルの管理が可能な Active Directory Kerberos レルムを使用できるよう設定する必要があります。

Active Directory レルムは、**/etc/krb5.conf** ファイルの **[libdefaults]** セクションのデフォルトドメインとして、さらに **[realms]** セクションの KDC として設定される必要があります。**[domain_realm]** セクションでは Active Directory ドメインを定義する必要があります。

シームレスな Kerberos の使用を可能にするには、Winbind Kerberos ロケータープラグインが *samba-winbind-krb5-locator* パッケージからインストールされていることを確認してください。これにより、Winbind およびそのユーザーすべて、また Kerberos ライブラリーとそのユーザーすべてが常に同じ KDC を使用するようになります。

```
[libdefaults]
...

  default_realm = ADEXAMPLE.COM

[realms]
  ADEXAMPLE.COM = {
    kdc = kdc.adexample.com
}

[domain_realm]
 adexample.com = ADEXAMPLE.COM
  .adexample.com = ADEXAMPLE.COM
```

4.1.1.3. DNS

ローカル DNS サービスは、ドメインコントローラーとして Active Directory を使用できるように設定する必要があります。DNS は Kerberos のホスト名およびドメインの解決のために必要です。多くのシステムでは、Active Directory をネームサーバーとして設定しなくても Samba と Active Directory の統合が十分に機能するよう DNS が適切に設定されていますが、Active Directory をネームサーバーとして使用することにより、解決に至るまでの潜在的な問題を防ぐことができます。またドメインは **search** ディレクティブとして追加し、Active Directory ドメインが検索および検出用に使用されるようにする必要があります。

DNS 設定は **/etc/resolv.conf** ファイルに設定されます。

```
nameserver 1.2.3.4
search adexample.com
```

4.1.1.4. PAM および NSS

PAM および NSS は、ローカルのアプリケーションが Active Directory によって提供される Kerberos 資格情報を使用できるようにします。これにより、システムアプリケーションおよびドメインユーザーのシングルサインオンが有効になります。使いやすさの点、また資格情報のオフラインキャッシュその他の機能の点では、Winbind を使用することをお勧めします。

PAM の場合、Winbind ライブラリーは認証、アカウント、パスワード、およびオプションのセッション管理用に設定されます。これは **/etc/pam.d/system-auth** ファイルに設定されます。

```
auth        required      pam_env.so
auth        sufficient    pam_unix.so nullok try_first_pass
auth        requisite     pam_succeed_if.so uid >= 500 quiet
auth sufficient pam_winbind.so use_first_pass
auth        required      pam_deny.so

account     required      pam_unix.so broken_shadow
account     sufficient    pam_localuser.so
account     sufficient    pam_succeed_if.so uid < 500 quiet
account [default=bad success=ok user_unknown=ignore] pam_winbind.so
account     required      pam_permit.so

password    requisite     pam_cracklib.so try_first_pass retry=3 type=
password    sufficient    pam_unix.so sha512 shadow nullok try_first_pass
use_authtok
password sufficient pam_winbind.so use_authtok
password    required      pam_deny.so

session     optional      pam_keyinit.so revoke
session     required      pam_limits.so
session     [success=1 default=ignore] pam_succeed_if.so service in crond
quiet use_uid
session     required      pam_unix.so
session     optional      pam_krb5.so
session optional pam_winbind.so use_first_pass
```

もう 1 つの重要な設定ファイルは **/etc/security/pam_winbind.conf** です。この中には、Kerberos 認証、オフライン認証、またはホームディレクトリーの自動作成を含む各種のパラメーターおよびデフォルト値が設定されます。詳細は、pam_winbind.conf(5) man ページを参照してください。

NSS については、Winbind をオプションとして設定することにより、Active Directory をパスワード、シャドウ (ユーザー) およびグループ用に使用することができます。さらに、ホストの設定に使用するために **WINS** サービスオプションを追加することもできます。アカウントを調べる際には、常に **files** を最初の位置に使用してください。この設定により、ローカルシステムユーザーおよびサービスによるログインやリソースへのアクセスが可能になります。

NSS 設定は **/etc/nsswitch.conf** ファイルに設定されます。

```
passwd:     files winbind
shadow:     files winbind
group:      files winbind

hosts:      files dns wins
```

注記

PAM および NSS は、Active Directory との統合用に手動で設定することができないことに注意してください。その代わりに、**authconfig** ユーティリティーを使用します。詳細は、「**authconfig** を使用したドメインメンバーの設定」を参照してください。

4.1.2. Winbind および Samba を使用した認証

ファイルの管理には、適切な所有権の設定と適切な関係者にアクセスを制御するという 2 つの重要なタスクが含まれます。これらはいずれも、どのようにユーザーを効果的に識別し、認証できるかという点と関連します。Winbind は 3 つの相互に関連する機能を提供します。

- ローカル PAM 設定を使用したユーザーの認証
- NSS 参照を使用した ID、ユーザー名およびグループの解決
- マップされた Active Directory SID およびローカル UID/GID 番号のデータベースの作成

Winbind は Samba の一部であり、Active Directory ドメインに直接接続されます。ローカルの Linuxシステムは、AD に保存される詳細なマシンアカウントで表される、Windows で認識可能な完全なドメインメンバーになります。PAM および NSS は、ローカルシステム上のユーザー識別に Winbind を使用するように設定されます。

Windbind の使用に関連する他の側面には以下が含まれます。

- Winbind は主にマシンアカウントの資格情報 (Active Directory のマシンアカウントとしての Linux マシンの表示) を維持します。他の機能の中でも、この機能はマシンアカウント資格情報を更新するか、またはパスワードポリシーのローカルストアを更新する (またはこれに準拠させる) ために使用できます。
- Winbind は POSIX 属性を RFC 2307 属性の形式または「Microsoft Services for Unix」拡張機能 (バージョン 3.5 とバージョン 3.0) の形式でサポートします。詳細は、idmap_ad(8) man ページを参照してください。
- ドメインへの参加は、Samba が提供するユーティリティーを使って実行されます (**net ads join** などのコマンドを使用)。Kerberos チケットの管理は、チケットの更新およびチケットの再取得を含め、Winbind によって実行されます。
- **smb.conf** ファイルのみが ID マッピングを定義する場所になります。

注記

Red Hat Enterprise Linux では、Active Directory との直接的な統合の有効な代替方法として SSSD を使用することをお勧めします。詳細は、*2章Active Directory を SSSD のアイデンティティープロバイダーとして使用する* を参照してください。

4.2. 設定ファイル、オプションおよびパッケージの要約

表4.1 システム設定ファイル、必須オプションおよび必須パッケージ

サービス	設定ファイル	必須パラメーター	必須パッケージ

Samba	**/etc/samba/smb.conf**	`[global]`
		`workgroup = ADEXAMPLE`
		`security = ads`
		`realm = ADEXAMPLE.COM`
			samba

Winbind	**/etc/security/pam_winbind.conf**		samba-winbind

Kerberos	**/etc/krb5.conf**	`[libdefaults]`
		`default_realm = ADEXAMPLE.COM`
		
		`[realms]`
		`ADEXAMPLE.COM = {`
		`kdc = kdc.adexample.com`
		`}`
		
		`[domain_realm]`
		`adexample.com = ADEXAMPLE.COM`
		`.adexample.com = ADEXAMPLE.COM`
			krb5-workstation

PAM	**/etc/pam.d/system-auth** または **/etc/pam.d/system-auth-ac** (authconfig を使用)	**auth** sufficient pam_winbind.so use_first_pass **account** [default=bad success=ok user_unknown=ignore] pam_winbind.so **password** sufficient pam_winbind.so use_authtok **session** optional pam_winbind.so use_first_pass	

NSS	**/etc/nsswitch.conf**	`#required`
		`passwd: files winbind`
		`shadow: files winbind`
		`group: files winbind`
		
		`#optional`
		`hosts: files dns wins`	

DNS	**/etc/resolv.conf**	`nameserver IPaddress` `search domainName`	

4.3. authconfig を使用したドメインメンバーの設定

「設定ファイル、オプションおよびパッケージの要約」に概略されているすべての設定は、DNS 設定の例外を除き、**authconfig** ユーティリティーを使用して自動的に実行されます。さらに、設定ファイルは **authconfig** によってバックアップされます。

4.3.1. authconfig の引数および設定パラメーター

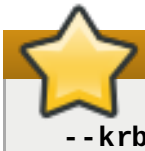
認証設定 (Authentication Configuration) ユーティリティーは、Winbind をローカルシステムの認証ストアとして設定するために使用する際に、Samba、Kerberos および Active Directory 統合の必要な設定ファイルを自動的に更新します。表4.2「authconfig 引数および設定ファイルパラメーター」は、それぞれのニマンドオプションと共に設定されるパラメーターを示しています。

表4.2 authconfig 引数および設定ファイルパラメーター

サービス	CLI オプション	GUI フィールド	設定ファイル	設定パラメーター
Samba	--smbsecurity	Security Model	/etc/samba/smb.conf	security
Samba	--smbworkgroup	Winbind Domain	/etc/samba/smb.conf	workgroup

▸ Samba
▸ Kerberos
	--smbrealm	Winbind ADS Realm	▸ Samba
			  ◦ /etc/samba/smb.conf
			▸ Kerberos
			  ◦ /etc/krb5.conf
				▸ Samba
				  ◦ [global] の realm
				▸ Kerberos
				  ◦ [libdefaults] の default_realm
				  ◦ [realms] の realm entry (**REALMNAME = {...}**)

Kerberos	--smbservers	Winbind Domain Controllers	/etc/krb5.conf	[realms] の realm entry (例: **REALMNAME {...}**) の KDC
Kerberos	--krb5realm		/etc/krb5.conf	[domain_realm] の domain entry
PAM	--enablewinbindauth		/etc/pam.d/system-auth	auth, account, password, sessions
NSS	--enablewinbind		/etc/nsswitch.conf	passwd, shadow, group
NSS	--enablewins		/etc/nsswitch.conf	hosts
Winbind	--enablecache			
Winbind	--enablewinbindkrb5			
Winbind	--enablewinbindoffline			



重要

--krb5realm オプションの値は、ドメインを適切に設定するために**--smbrealm** に指定される値と同一である必要があります。

4.3.2. `authconfig` を使用した Active Directory 認証の CLI 設定

1. *samba-winbind* パッケージをインストールします。これは Samba サービスの Windows 統合機能に必要ですが、デフォルトではインストールされません。

```
[root@server ~]# yum install samba-winbind
```

2. *krb5-workstation* パッケージをインストールします。これは Kerberos レルムに接続し、プリンシパルおよびチケットを管理するために必要です。

```
[root@server ~]# yum install krb5-workstation
```

3. *samba-winbind-krb5-locator* パッケージをインストールします。これには、ローカル Kerberos ライブラリーが Samba および Winbind が使用するものと同じ KDC を使用できるようにするシステム Kerberos ライブラリーのプラグインが含まれます。

```
[root@server ~]# yum install samba-winbind-krb5-locator
```

4. Active Directory ドメインをネームサーバーとして、または検索に使用できるように **`/etc/resolv.conf`** ファイルで DNS 設定を編集します。

```
nameserver 1.2.3.4
search adexample.com
```

5. **`authconfig`** ユーティリティーは、新規設定の定義と設定変更に使用できるため、所定の時間に起動しなければならないオプションなどの要件は設定しません。

   以下の例では、Samba、Kerberos、PAM および NSS に必要なすべてのパラメーターを示しています。さらに、オフラインアクセスを許可する Winbind のオプションや、システムアカウントの機能を継続させるためのローカルシステムのオプションも含まれます。このコマンド例は複数の行に分割され、読みやすくするために注釈が付けられています。

```
[root@server ~]# authconfig
      // NSS
      --enablewinbind
      --enablewins
      // PAM
      --enablewinbindauth
      // Samba
      --smbsecurity ads
      --smbworkgroup=ADEXAMPLE
      --smbrealm ADEXAMPLE.COM
      // Kerberos
      --smbservers=ad.example.com
      --krb5realm=ADEXAMPLE.COM
      // winbind
      --enablewinbindoffline
      --enablewinbindkrb5
      --winbindtemplateshell=/bin/sh
      // general
      --winbindjoin=admin
      --update
```

```
        --enablelocauthorize
        --savebackup=/backups

[/usr/bin/net join -w ADEXAMPLE -S ad.example.com -U admin]
```

--winbindjoin オプションは **net join** コマンドを自動的に実行し、システムを Active Directory ドメインに追加します。

--enablelocalauthorize オプションは、**/etc/passwd** ファイルをチェックするためのローカル認可操作を設定します。これにより、ローカルアカウントを使用してユーザーおよび Active Directory ドメインを認証することができます。

注記

--savebackup オプションは、推奨されますが必須のオプションではありません。設定ファイルに変更を加える前に、これを指定されるディレクトリーにバックアップします。設定エラーがある場合や設定が後で変更される場合、**authconfig** はバックアップを使用して変更を元に戻すことができます。

4.3.3. authconfig GUI を使用した Active Directory 認証の設定

authconfig GUI にある設定オプションの数は CLI のオプション数よりも少なくなります。たとえば、ドメインに参加するために Samba、NSS、Winbind を設定することはできますが、Kerberos または PAM は設定できません。UI を使用する場合は、後者については手動で設定する必要があります。

注記

authconfig コマンドラインユーティリティーはデフォルトでインストールされますが、GUI は、デフォルトでは利用できない *authconfig-gtk* パッケージを要求します。

1. **samba-winbind** パッケージをインストールします。これは Samba サービスの Windows 統合機能に必要ですが、デフォルトではインストールされません。

```
[root@se yum install samba-winbind
```

2. **krb5-workstation** パッケージをインストールします。これは Kerberos レルムに接続し、プリンシパルとチケットを管理するために必要です。

```
[root@se yum install krb5-workstation
```

3. デフォルトレルムとしてのActive Directory Kerberos レルム、およびローカルシステムの KDC を設定します。

```
[root@se vim /etc/krb5.conf

[libdefaults]
...

  default_realm PLE.COM
```

```
[realms]
  ADEXAMPLE.COM
    kdc = kdc.adcom
}

[domain_realm]
 adexample.com =LE.COM
  .adexample.comMPLE.COM
```

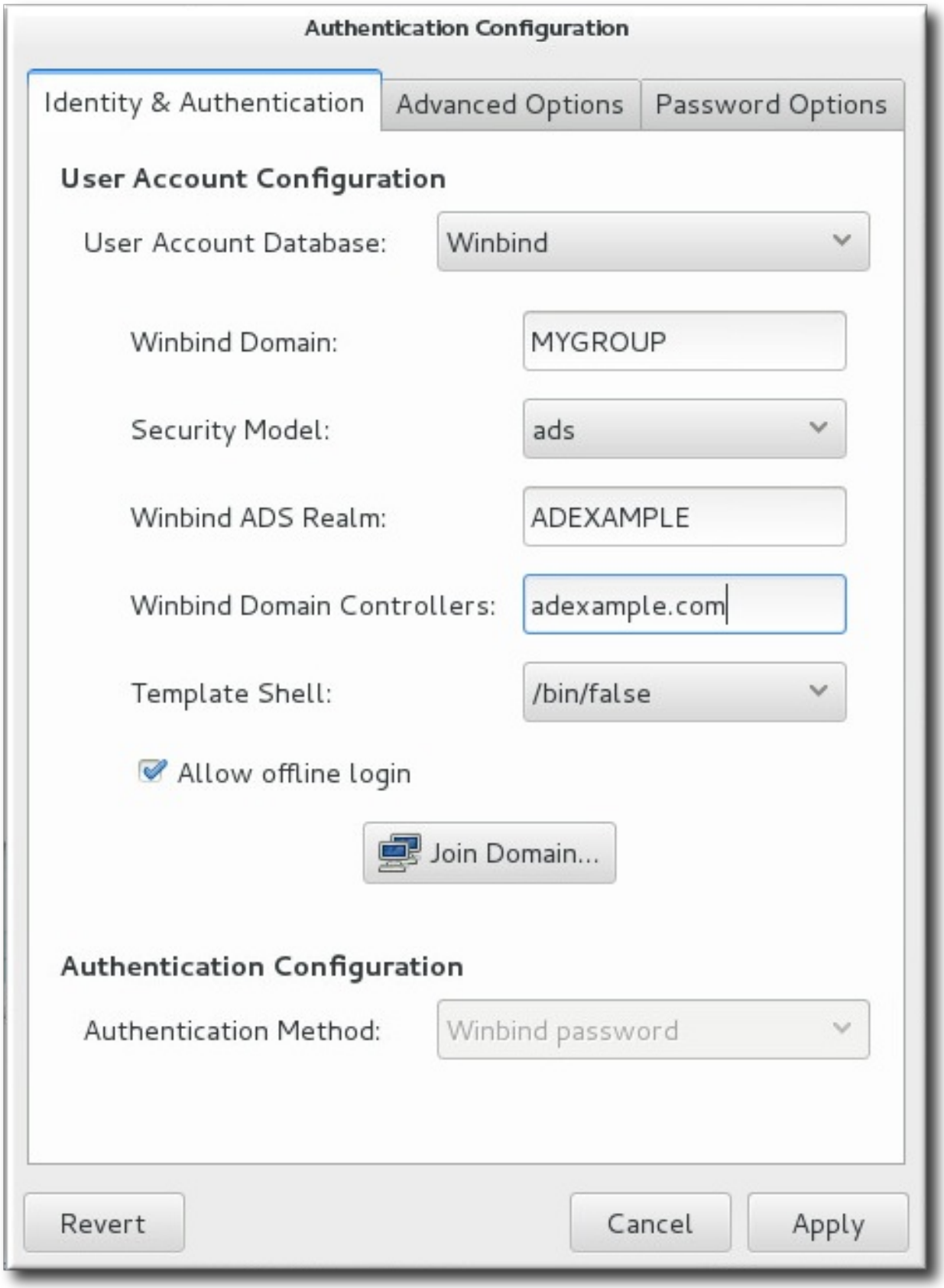
4. Active Directory ドメインをネームサーバーとして、または検索に使用できるように **/etc/resolv.conf** ファイルで DNS 設定を編集します。

```
nameserver 1.2.3
search adexample
```

5. 認証設定ツール (Authentication Configuration Tool) を開きます。

```
[root2se authconfig-gtk
```

6. **Identity & Authentication** タブで、**User Account Database** ドロップダウンメニューから **Winbind** を選択します。



7. Microsoft Active Directory ドメインコントローラーに接続するために必要な情報を設定します。

   - **`Winbind Domain`** では Windows ワークグループを指定します。このフィールドのエントリーには、**`DOMAIN`** のように Windows 2000 形式を使用する必要があります。
   - **`Security Model`** では、Samba クライアントを使用するためにセキュリティーモデルを設定します。Active Directory Server レルムで Samba がドメインメンバーとして機能するように設定するための正確な値は **`ads`** です。

- **Winbind ADS Realm** では、Samba サーバーが参加する Active Directory レルムを指定します。
- **Winbind Domain Controllers** では、ドメインコントローラーが使用するホスト名または IP アドレスを指定します。
- **Template Shell** では、Windows ユーザーアカウントの設定に使用するログインシェルを設定します。この設定はオプションになります。
- **Allow offline login** は、認証情報をローカルキャッシュに保存できるようにします。キャッシュは、システムのオフライン時に、ユーザーがシステムリソースに対して認証を試行する際に参照されます。

8. **Join Domain** ボタンをクリックして **net ads join** コマンドを実行し、Active Directory ドメインに参加します。この操作はドメインへの即時参加を目的としています。この設定を保存してから、後で **net ads join** コマンドを手動で実行することもできます。
9. **Apply** ボタンをクリックして設定を保存します。

パート II. Linux ドメインと Active Directory ドメインの統合

第5章 Active Directory および Identity Management によるクロスレルム信頼の作成

Kerberos は **信頼されるレルム** の設定を許可します。各レルムには、それぞれのリソースとユーザーが含まれますが、信頼関係により、いずれの信頼されたレルムのユーザーもチケットを取得し、実際のメンバーであるかのようにピアレルムのマシンまたはサービスに接続できます。

Windows と Linux ドメインが LDAP サービス、DNS 管理、さらには Kereros レルムを実装する方法は異なるため、Active Directory と Linux ドメイン間の直接的な信頼を手動で設定することは容易ではありません。IdM を使用した信頼関係では、Kerberos 信頼および DNS マッピングを一元的に定義し、これを設定するため、Active Directory ユーザーは資格情報の単一セットを使用して Linux ホストおよびサービスに完全に透過的な方法でアクセスできます。

5.1. 信頼について

Kerberos には、2 つの別個のレルム間の関係を作成する機能があります。これは、*クロスレルム信頼* と呼ばれています。これらのレルムでは共有チケットおよびキーを作成し、1 つのレルムのメンバーが両方のレルムのメンバーとして認識されるようにします。つまり、1 つのレルムがもう 1 つのレルムを信頼することになります。

クロスレルム Kerberos 信頼は Kerberos レルムに制限されます。ただし、Active Directory または Red Hat Enterprise Linux の IdM などのアイデンティティー管理には、ドメイン定義に Kerberos 以外のサービス (とくに LDAP および DNS) が含まれます。IdM 信頼の場合は、Kerberos レルムだけではなく、ドメイン全体で信頼される関係を設定することができます。

5.1.1. 信頼関係のアーキテクチャー

Active Directory および Identity Management の両方は、Kerberos、LDAP、DNS、または証明書サービスなどの各種のコアサービスを管理します。これら 2 つの異なる環境を透過的に統合するには、すべてのコアサービスが相互にシームレスに対話できる必要があります。

コアサービスは、Kerberos レルムおよび DNS ドメインという 2 つの主要なポイントで相互に作用します。証明書ストア、LDAP エントリーおよびその他のサービスは、Active Directory および IdM 用に別個に管理できます。それらのサービスが交差するポイントは、アイデンティティーが認証されるポイント (Kerberos) とドメイン間でクエリーをルーティングするメカニズム (DNS) が機能するポイントになります。

信頼は 2 つのドメイン間のアイデンティティー/アクセス関係を設定します。Active Directory 環境は複雑になり得るので、サブドメイン、ルートドメイン、フォレスト、および外部ドメイン間には Active Directory 信頼の複数の異なるタイプや取り決めが存在することになります。信頼は、あるドメインまたはレルムから別のドメインまたはレルムへのパスです。アイデンティティーおよび情報がドメイン間で移動することは *信頼フロー* と言います。

基本的に、信頼フローは 1 方向のみになります。*信頼されるドメイン* にはユーザーが含まれ、*信頼する側のドメイン* はリソースへのアクセスを許可します。信頼の下では、ユーザーは信頼する側のドメインのリソースにアクセスできますが、信頼する側のドメインのユーザーは、信頼されるドメインのリソースにアクセスすることはできません。図5.1「基本的な一方向の信頼」 では、レルム A はレルム B によって信頼されていますが、レルム B はレルム A によって信頼されていません。信頼の方向は一方向です。

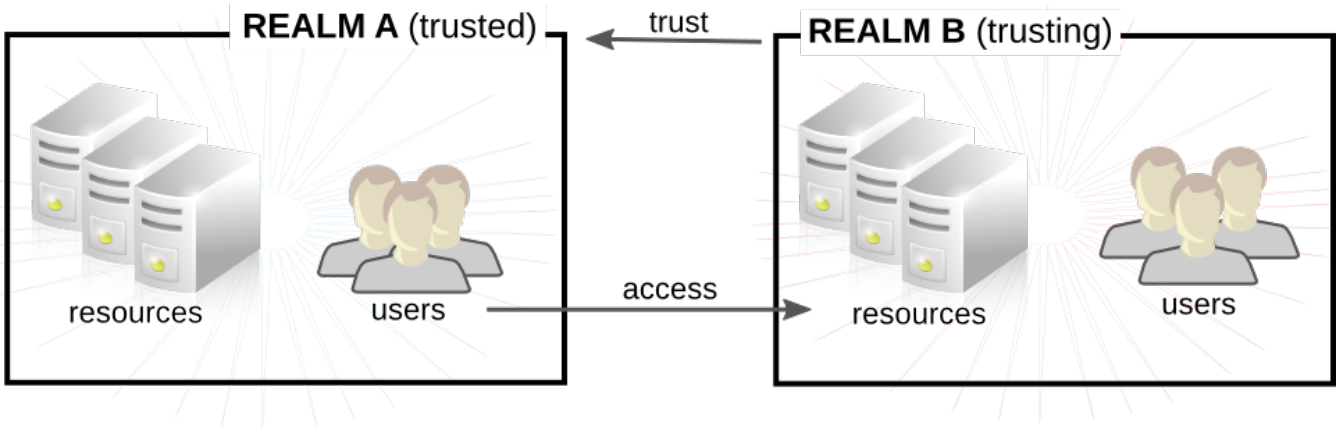


図5.1 基本的な一方向の信頼

信頼は *推移する* こともあります。つまり、ドメインはもう 1 つのドメインまたはその 2 つ目のドメインによって信頼されるその他のドメインを信頼します。信頼は *推移しない* ようにすることもできます。これは、信頼を明示的に組み込まれたドメインに制限することを意味します。

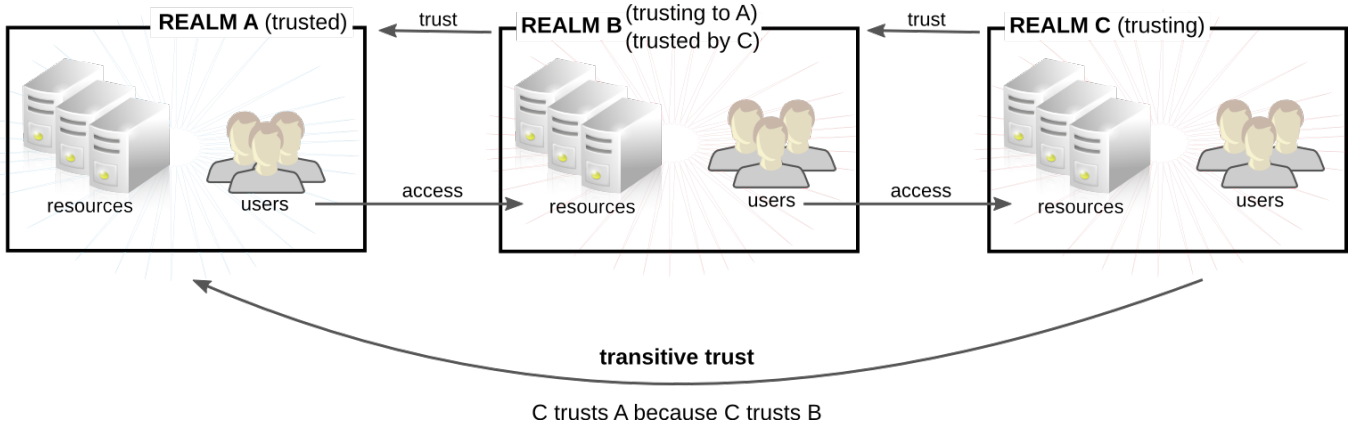


図5.2 推移的な信頼

Active Directory フォレスト内では、ドメイン間の信頼関係はデフォルトでは通常、双方向で推移的です。2 つの AD フォレスト間の信頼は 2 つのフォレストルートドメイン間の信頼であるため、この信頼は双方向または一方向にすることができます。フォレスト間の信頼の推移性は明示的です。つまり、フォレストのルートドメインにつながる AD フォレスト内のドメイン信頼は、フォレスト間信頼に基づいて推移します。ただし、あるフォレスト間信頼と別のフォレスト間信頼の間では信頼は推移しません。明示的なフォレスト間信頼は、ある AD フォレストのルートドメインと別の AD フォレストのルートドメイン間で設定される必要があります。

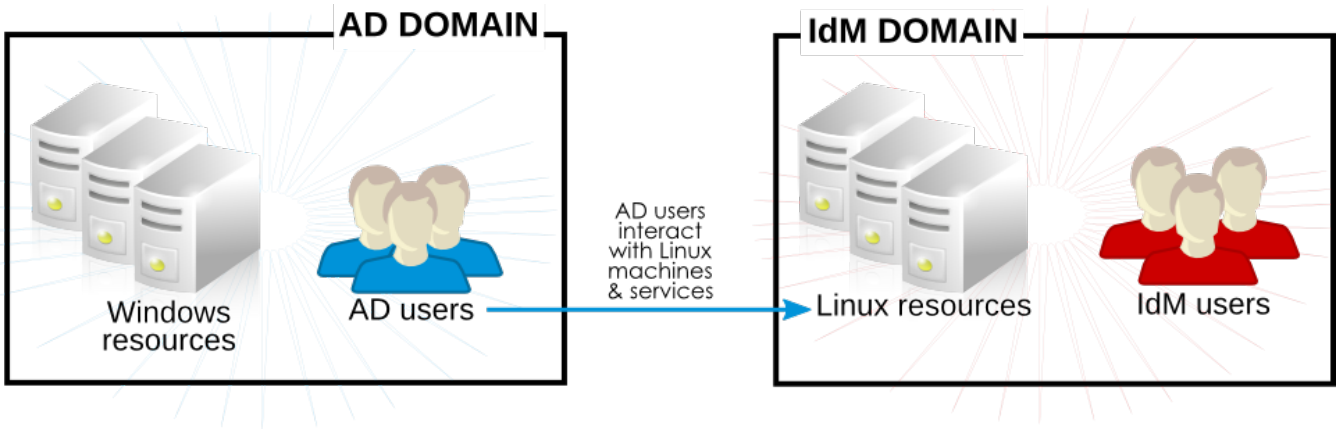


図5.3 信頼の方向

5.1.2. Active Directory セキュリティーオブジェクトおよび信頼

信頼の下では、ユーザーは外部ドメインのリソースにアクセスします。*信頼パス* は、アクセスを取得する個々のケースでセキュアな通信が行われるドメインのシーケンスです。Active Directory 信頼は、アイデンティティーを評価するために NTLM (NT LAN Manager) を使用します。Identity Management 信頼を含むレルム信頼は、特権付きアクセス証明書を作成し、送信し、検証するために Kerberos を使用し、外部アプリケーション用に Kerberos チケットを作成します。

アプリケーションが認証要求を処理するために使用するプロトコルは、NTLM または Kerberos のいずれかにすることができます。これらのプロトコルはどちらも Active Directory の Net Logon 層と対話し、ここからドメインオブジェクトにアクセスするためのプロセスは同じになります。それぞれの Active Directory サーバーは、すべてのローカルで定義されたセキュリティーポリシーが含まれる *ローカルセキュリティー機関* (LSA) を維持し、ローカルユーザーおよび識別子、チケットおよび PAC、およびその他のセキュリティーデータを識別するための方法を提供します。

信頼の下での Active Directory および IdM のすべての Kerberos 通信は GSS-API を使用します。ローカルセキュリティー機関のほかにも、大規模な Active Directory 設定が必要になります。ドメインシステムコントローラーは、ドメイン内のすべてのオブジェクトのユーザーおよびグループ情報を含むすべてのセキュリティー情報を保持します。ドメインのルートには、フォレスト全体にあるすべてのユーザー、グループ、およびオブジェクトのグローバルカタログがあります。信頼が設定されていると、Windows ユーザーについての情報はシステムコンテナーまたはグローバルカタログから取得できます。

IdM は複数の異なる Active Directory フォレストとの信頼関係の一部に組み込むこともできます。いったん信頼が設定されると、同じコマンドと手順が実行して他のフォレストとの信頼を後で追加することができます。IdM は複数のまったく無関係のフォレストを同時に信頼できるため、異なる相互に関連性のない Active Directory フォレストのユーザーが同じ共有 IdM ドメイン内のリソースにアクセスできるようになります。

5.1.3. IdM における信頼アーキテクチャー

Identity Management 側では、IdM サーバーは Active Directory アイデンティティーを認識でき、かつアクセス制御のためにグループメンバーシップを適切に処理できる必要があります。Microsoft PAC (MS-PAC、Privilege Account Certificate) にはユーザーについての必要な情報が含まれます。これには、ユーザーのセキュリティー ID、ドメインユーザー名、およびグループメンバーシップが含まれます。Identity Management には、Kerberos チケットの PAC でデータを分析するための 2 つのコンポーネントがあります。

- SSSD は、Active Directory 上の ID 検索を実行し、認可のためにユーザーおよびグループセキュリティー識別子 (SID) を取得します。さらに SSSD は、ユーザー、グループ、およびユーザーのチケット情報をキャッシュし、Kerberos および DNS ドメインをマップします。

- Identity Management (Linux ドメイン管理) は、Active Directory ユーザーを、IdM ポリシーおよびアクセスのために IdM グループと関連付けます。

注記

SELinux、sudo、およびホストベースのアクセス制御など、Linux ドメイン管理のアクセス制御ルールおよびポリシーは Identity Management で定義され、適用されます。Active Directory 側で設定されるいずれのアクセス制御ルールも IdM では評価または使用されます。関連する Active Directory 設定はグループメンバーシップのみになります。

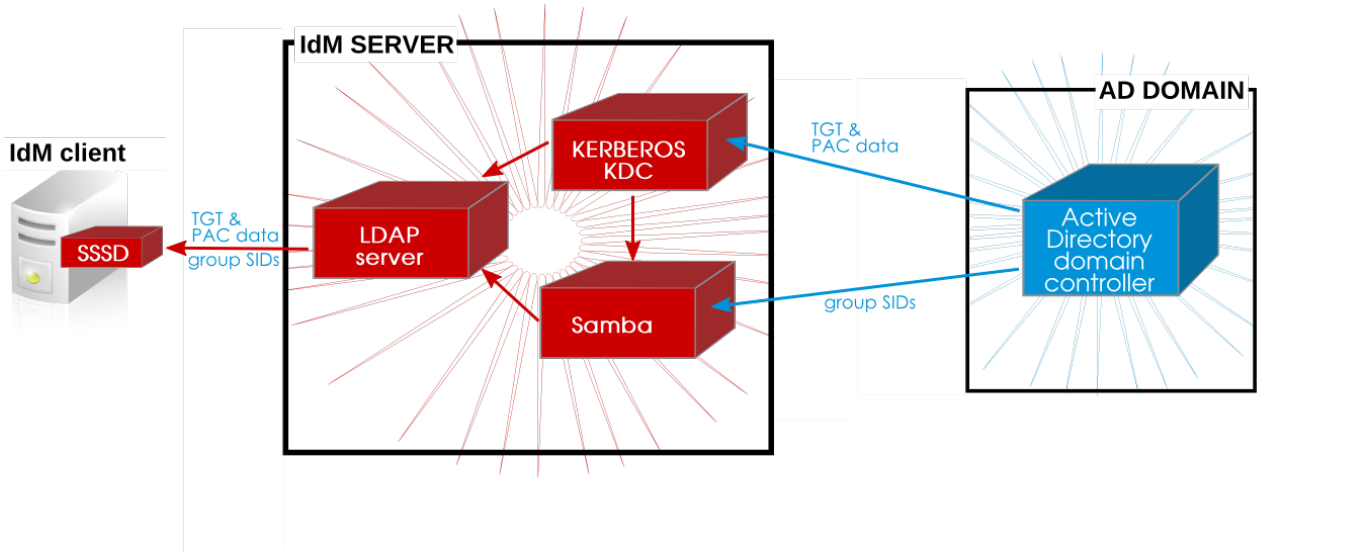


図5.4 信頼用のアプリケーションおよびサービス

5.1.3.1. Active Directory PAC および IdM チケット

Active Directory のグループ情報は、**privileged access certificates** または MS-PAC と呼ばれる特殊なデータセットで Active Directory ユーザーの各 Kerberos チケットの識別子の一覧に保存されます。PAC のグループ情報はまず Active Directory グループに、次に対応する IdM グループにマップされ、アクセスの判別が行われます。PAC は基本的にはアカウントユーザビリティーの拡張機能であり、Windows ドメイン内の他の Windows クライアントおよびサーバーに対してエンティティーを識別する手段として Kerberos チケットに組み込まれます。

IdM リソースの場合、Active Directory ユーザーがサービスのチケットを要求すると、IdM はその要求を Active Directory に転送してユーザー情報を取得します。Active Directory によって送り戻される PAC 情報は Kerberos チケットに組み込まれます。

IdM (IdM クライアントとして SSSD を経由) は、Active Directory グループの **セキュリティー識別子** (SID) を PAC から抽出し、PAC の Active Directory SID を、IdM グループのメンバーとして設定されるグループ SID と比較します。Active Directory グループが IdM グループのメンバーである場合、IdM グループ SID が PAC に追加され、Kerberos チケットは新規の PAC で更新されます。

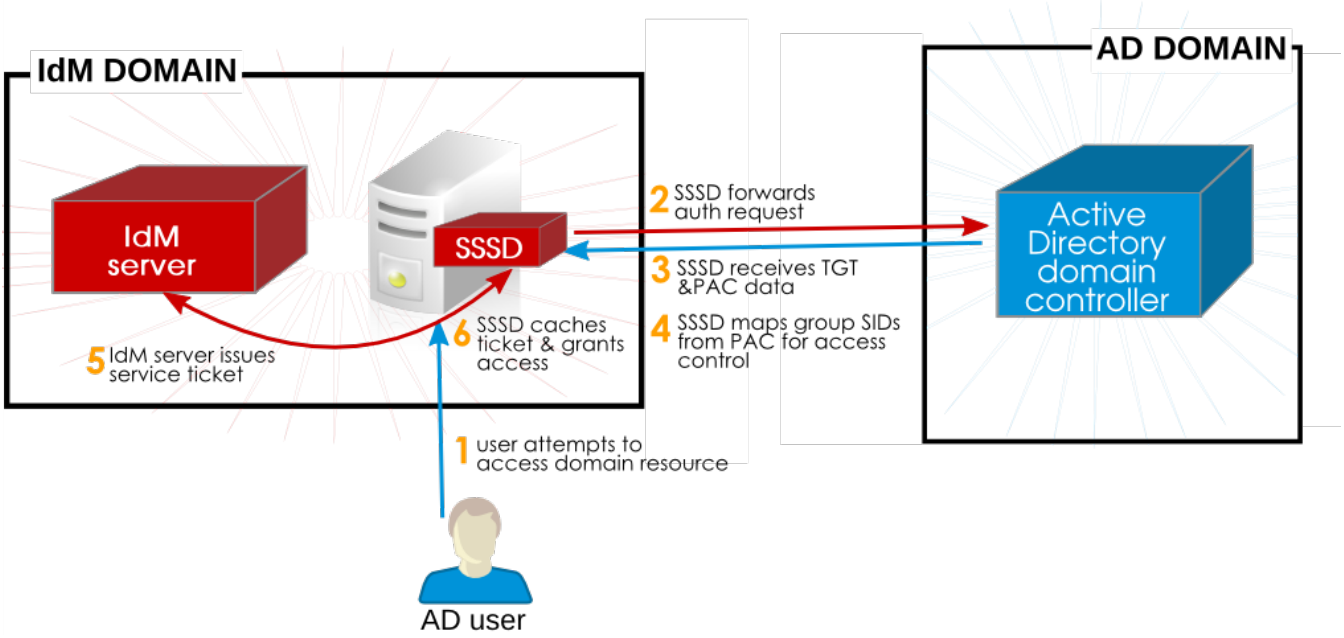


図5.5 IdM、SSSD および Active Directory

次に新規チケットは、ユーザーのサービスチケットを生成するために使用され、アクセスルールに従って IdM がホストするサービスへのアクセスがユーザーに付与されます。さらに、SSSD ユーザーキャッシュの IdM グループ情報は、 Active Directory ユーザーについてのマップされた IdM グループが含まれるように更新されます。

新規サービスがアクセスされると、SSSD は各ユーザーの複数の TGT およびチケットを保存します。

これを簡単に説明すると、Active Directory はグループの識別子に基づいて各ユーザーのグループ一覧を提供します。IdM は、その Active Directory グループの一覧を IdM グループのメンバーシップと比較します (ここで、各グループメンバーは名前または DN でなく SID で識別されます)。ユーザーが属する Active Directory グループが IdM ドメインに認識される場合に、そのユーザーも IdM ドメインによって認識されます。

5.1.3.2. Active Directory ユーザーおよび IdM グループ

重要: Active Directory ユーザーは、 Active Directory ユーザーエントリーではなく Active Directory グループメンバーシップによって IdM ドメインで認識されます。 つまり、 IdM ドメインが信頼するのは Active Directory **ユーザー** ではなく、Active Directory **グループ** です。

ただし、このメソッドによる Active Directory グループ SID の IdM グループメンバーへのマッピングで重要なポイントになるのは IdM のグループ構造です。Active Directory グループは Linux グループとは異なる属性を持ち、さらに IdM グループとは異なる属性を持ちます。また最も重要な点として、IdM グループは POSIX グループですが、Active Directory グループは POSIX グループではありません。

IdM は、仲介役として、POSIX 以外のグループタイプである **外部グループ** を使用します。これは IdM または Linux システムの外部にあるエンティティーがメンバーとして加わることを許可します。その外部グループは後に標準 IdM (POSIX) グループにメンバーとして追加できます。

Active Directory グループが IdM グループに追加されると、それらは SID または **DOMAIN\group_name** または **group_name@domain** 形式の名前で識別できます。次に IdM はグループ名を SID に対して解決し、提供されるユーザー PAC との比較に使用するためにその SID をグループメンバーエンティティーとして保存します。

Active Directory ユーザーのグループを実際に設定する方法は、「Active Directory ユーザー用の IdM グループの作成」 に説明されています。

5.1.3.3. ユーザー ID のマッピングおよび POSIX 属性の使用

Active Directory ユーザーエントリーは IdM サーバーにありません。それらは同期されたり、コピーされたりすることはありません。IdM リソースで要求されるすべての情報は Active Directory からプルされ、キャッシュされます。Active Directory 側の固有な識別子 (セキュリティー ID、セキュリティードメイン ID とユーザー ID の組み合わせ) はユーザー名に関連付けられます。ユーザー名が IdM リソースにアクセスするために使用される際、IdM (Samba 経由) はユーザー名を SID に対して解決してから、Active Directory ドメイン内の SID の情報を検索します。

Linux システムでは、すべてのユーザーがローカル UID 番号およびグループ ID 番号を持っている必要があります。ユーザーが IdM で作成されると、ユーザーにはデフォルトで UID/GID 番号が割り当てられます。信頼されるユーザーでも、Linux システムで UI/GID 番号が必要になります。その UID/GID 番号は IdM で生成できますが、Active Directory エントリーに UID/GID 番号がすでに割り当てられている場合、異なる番号を割り当てることにより競合が生じます。Active Directory で定義される POSIX 属性 (UID/GID 番号および優先されるログインシェルを含む) を使用することは可能です。

Active Directory は、フォレスト内のすべてのオブジェクトの情報のサブセットを **グローバルカタログ** に保存します。このグローバルカタログには、フォレスト内のすべてのドメインのすべてのエントリーが含まれます。

注記

Active Directory で定義された POSIX 属性を使用するには、それらの属性がグローバルカタログに公開されている必要があります。そうでない場合、POSIX 属性を持つ AD ユーザーは IdM リソースを利用することができません。

IdM は信頼の設定時に使用する ID 範囲のタイプを自動検出しますが、範囲タイプは、**`ipa trust-add`** コマンドを使用して手動で設定することができます。

```
ipa trust-add  –range-type=ipa-ad-trust-posix
```

表5.1 範囲のタイプ

範囲のオプション	説明
ipa-ad-trust	SID で設定された ID の場合: ユーザー名のマッピング
ipa-ad-trust-posix	Active Directory エントリーの POSIX 属性で定義される ID の場合

5.1.3.4. Active Directory ユーザー、IdM ポリシーおよび設定

SELinux、ホストベースのアクセス制御、sudo およびネットグループなどのいくつかの IdM ポリシー定義では、ポリシーの適用方法を識別する際にユーザーグループに依存します。

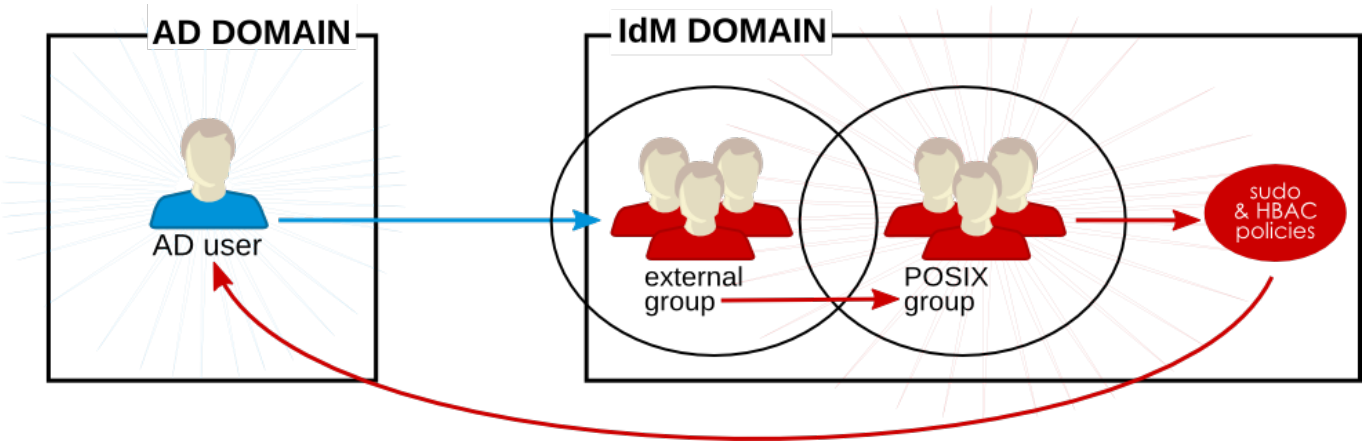


図5.6 Active Directory ユーザーおよび IdM グループおよびポリシー

Active Directory ユーザーは IdM ドメインに対して外部の位置付けになりますが、それらのグループは外部グループとして設定される限り、IdM グループのグループメンバーとして追加できます。IdM の外部グループは POSIX 以外のグループです。外部グループはその後に IdM グループ (POSIX グループ) のメンバーとして追加されます。

sudo、ホストベースのアクセス制御およびその他ポリシーは POSIX グループに適用され、最終的には IdM ドメインリソースにアクセスする際に Active Directory ユーザーに適用されます。

チケットの PAC にあるユーザー SID は Active Directory アイデンティティーに対して解決されます。これは、完全修飾ユーザー名または SID を使用して Active Directory ユーザーをグループメンバーとして追加できることを意味します。

注記

信頼されるユーザーグループの関連付けは動的に解決され、IdM ディレクトリーには保存されないため、**hbactest** などのテストツールは信頼されるユーザーに対しては機能しません。

5.1.4. 異なる DNS 信頼環境

Active Directory および Identity Management はどちらも DNS サービスを定義できます。それらの DNS ドメインは正常に相互に対話できる必要があります。以下の 2 つの DNS 設定が考えられます。

- DNS ドメインをそれぞれ独立させる。
- Identity Management を Active Directory のサブドメインとして設定する。

すべてのケースで、異なるドメインは必要に応じて要求を相互に転送し、それぞれの異なる DNS 名前空間を維持します。それらのドメインがクエリーを転送する際に相互をどのように認識するかを定義することがポイントになります。

重要

必須: 上位およびサブドメインの同一レコード

Identity Management 環境内で、**すべての** マシン名は完全に解決可能である必要があります。これには、信頼の下で信頼される Active Directory ドメイン内のマシンも含まれます。DNS 環境の設定に応じて、DNS サービスを設定する際に 2 つの異なる方法を使用できます。

IdM および Active Directory がより大きな共有名前空間内のサブドメインであるか、または IdM が Active Directory DNS 名前空間のサブドメインである場合、最も良い設定として **委任** を使用し、DNS ドメイン間の関係を作成することができます。委任 (NS) およびグルー (A または AAAA) レコードは、すべての上位ゾーン (**example.com** など) および下位ゾーン (**ipa.example.com** など) において同一である必要があります。つまり、上位ゾーンおよびサブドメインには同一の NS と A および AAAA レコードが含まれる必要があります。

Active Directory および IdM DNS ドメインが全く異なる名前空間にある場合、条件付きフォワーダーを使用します。この場合、転送ルールは両方の環境に置かれ、すべてのマシンを解決できるようにする必要があります。

設定オプション 1: 別々の DNS ドメイン

このケースでは、**ipaexample.com** および **adexample.com** などの 2 つの全く異なる名前空間があります。これらのドメインが通信できるようにするには、それらを相互のドメインの条件付きフォワーダーとして設定する必要があります。

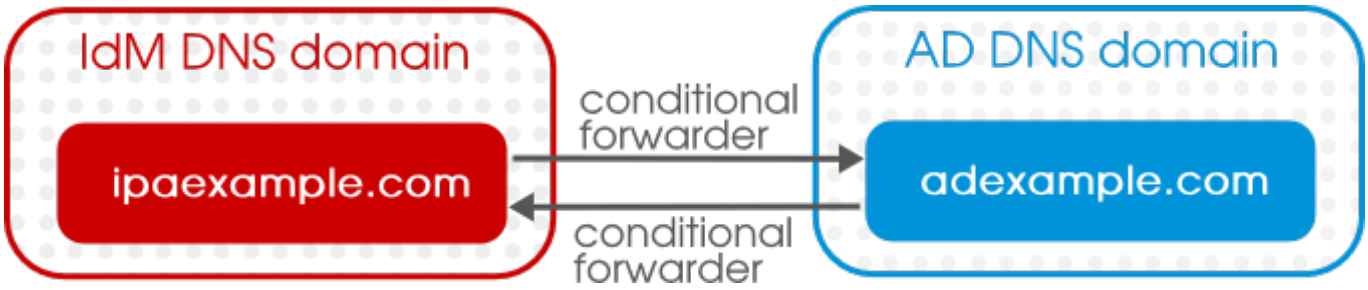


図5.7 別々の DNS ドメインにおける信頼

設定オプション 2: 別々の DNS サブドメイン

同様のシナリオは、Active Directory ドメインおよび IdM ドメインの両方がより大きな中央ドメインのサブドメインである場合に当てはまります。たとえば、Active Directory ドメインは **ad.example.com** であり、Identity Management ドメインは **ipa.example.com**、さらにこれら両方の上位のドメインは **example.com** であるとします。同等のサブドメインを使用する場合には**フォワーダーは使用しないでください**。その代わりに DNS 委任を使用してください。グローバル設定に設定されるフォワーダーは、ローカルサーバーが権限サーバーとして設定される場合でも委任を上書きします。代わりに、同一の委任ルールを上位ドメインおよびサブドメインに設定する必要があります。

設定オプション 3: Active Directory のサブドメインとしての Identity Management

このケースでは、Identity Management はより大きな Active Directory スペース内の名前空間になります (**linux.example.com** および **example.com** など)。IdM はすべての要求を Active Directory ドメインに送信するように設定 (forward-only ポリシー) することも、クエリーをまず Active Directory に送信してから自らでそれらの解決を試行 (forward-first ポリシー) することもできます。

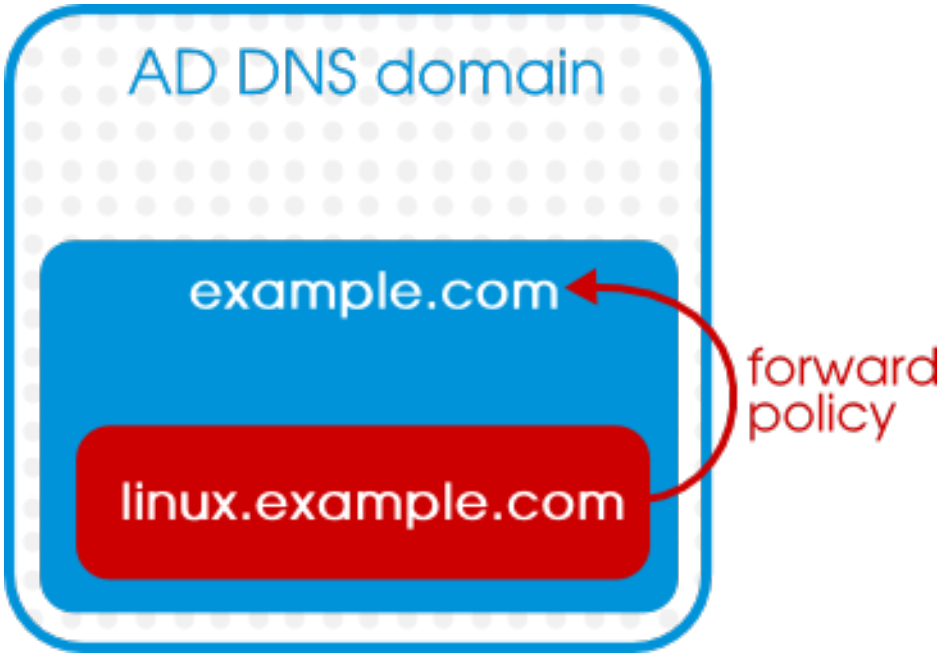


図5.8 Active Directory の DNS サブドメインとしての IdMの信頼

5.1.5. Identity Management の複数ドメインとの相互作用

IdM サーバーが信頼内のピアサーバーに接続する場合はいつでも、Active Directory DNS ドメインについての情報はドメインコントローラーから取得され、**cn=Realm Domains,cn=ipa,cn=etc,サフィックス** サブツリーで保存されます。

Active Directory ドメイン内に複数の DNS ドメインが定義されている場合、それらの DNS ドメインは設定サブツリーに個別に追加され、それらはクエリーのルートを適切なドメインに指定するために使用されます。

5.1.6. Active Directory 信頼の動作について起こり得る問題

5.1.6.1. Active Directory ユーザーおよび IdM 管理

信頼関係は一方向です。Active Directory ユーザーは Active Directory ドメイン内にのみ存在し、それらがアクセスできる IdM ドメイン内のリソースには制限があります。**Active Directory ユーザーは IdM 内に存在しないため、Active Directory ユーザーを IdM の管理者にすることはできません。**

さらに Active Directory ユーザーは、Web UI およびコマンドラインツールを含む IdM 管理ツールを使用することはできません。

5.1.6.2. 削除された Active Directory ユーザーの認証

デフォルトでは、すべての IdM クライアントは System Security Services Daemon (SSSD) を使用してユーザー ID および資格情報をキャッシュします。これにより、ローカルシステムでは、バックエンドプロバイダー(IdM または Active Directory) のいずれかが一時的に利用できない場合でも、すでに正常にログインしたことのあるユーザーの ID を参照することができます。

SSSD はユーザー一覧をローカルに維持するため、バックエンドに対して行われる変更がクライアントに即時に表示されない場合があります。

Active Directory ユーザーが IdM クライアントリソースに正常にログインした場合、そのユーザー ID はローカルクライアントと IdM サーバーの両方で SSSD にキャッシュされます。その Active Directory ユーザーが Active Directory で削除される場合、そのユーザーの ID は **依然として** IdM にキャッシュされます。つまり、そのユーザーは IdM リソースに正常にログインできることを意味します。

削除された Active Directory ユーザーは、任意のローカルクライアント **と** IdM サーバーの SSSD キャッシュの有効期限が切れるまで IdM リソースにログインできます。IdM サーバーが Active Directory から ID の取得を試行すると、このサーバーはユーザーが存在しないことを知らせる通知を受信し、ログインの試行は失敗します。

5.1.6.3. 資格情報キャッシュおよび Active Directory プリンシパルの選択

Kerberos 資格情報キャッシュは、サーバー名、ホスト名、次に (場合により) レルム名に基づいてクライアントプリンシパルをサーバープリンシパルに一致させるよう試行します。このクライアント/サーバー間のマッピングはホスト名およびレルム名を使用して実行されるので、Active Directory ユーザーのレルム名と IdM システムのレルム名の違いにより Active Directory ユーザーのバインディングで予期しない動作が生じる可能性があります。

つまり、実際には Active Directory ユーザーが **kinit** を実行してから SSH を使用して IdM リソースに接続する場合、そのプリンシパルはリソースチケット用に選択されないことになります。IdM プリンシパルがリソースのレルム名に一致するため、使用されるのは IdM プリンシパルになります。

たとえば、Active Directory ユーザーが **Administrator** で、ドメインが **ADEXAMPLE.ADREALM** の場合、プリンシパルは **Administrator@ADEXAMPLE.ADREALM** になります。

```
[root@server ~]# kinit Administrator@ADEXAMPLE.ADREALM
Password for Administrator@ADEXAMPLE.ADREALM:
[root@server ~]# klist
Ticket cache: KEYRING:persistent:0:0
Default principal: Administrator@ADEXAMPLE.ADREALM

Valid starting       Expires              Service principal
27.11.2013 11:25:23  27.11.2013 21:25:23
krbtgt/ADEXAMPLE.ADREALM@ADEXAMPLE.ADREALM
 renew until 28.11.2013 11:25:16
```

これは Active Directory チケットキャッシュのデフォルトプリンシパルとして設定されます。ただし、**任意の** IdM ユーザーが Kerberos チケット (**admin** など) も持つ場合、IdM デフォルトプリンシパルと共に別個の IdM 資格情報キャッシュも存在することになります。その IdM デフォルトプリンシパルは、Active Directory ユーザーが SSH を使用してリソースに接続する場合にホストチケット用に選択されます。

```
[root@vm-197 ~]# ssh -l Administrator@adexample.adrealm
ipaclient.example.com
Administrator@adexample.adrealm@ipaclient.example.com's password:

[root@vm-197 ~]# klist -A
Ticket cache: KEYRING:persistent:0:0
Default principal: Administrator@ADEXAMPLE.ADREALM

Valid starting       Expires              Service principal
27.11.2013 11:25:23  27.11.2013 21:25:23
krbtgt/ADEXAMPLE.ADREALM@ADEXAMPLE.ADREALM
 renew until 28.11.2013 11:25:16

Ticket cache: KEYRING:persistent:0:0
Default principal: admin@EXAMPLE.COM >>>>> IdM user
```

```
Valid starting       Expires              Service principal
27.11.2013 11:25:18  28.11.2013 11:25:16  krbtgt/EXAMPLE.COM@EXAMPLE.COM
27.11.2013 11:25:48 28.11.2013 11:25:16
host/ipaclient.example.com@EXAMPLE.COM >>>>> host principal
```

これは IdM プリンシパルのレルム名が IdM リソースのレルムに一致するために実行されます。

5.1.6.4. グループ SID の解決

Kerberos チケットの失効

net getlocalsid または **net getdomainsid** などの、Samba サービスから SID を取得するためのコマンドを実行すると、Kerberos キャッシュの既存の admin チケットが kill されます。

ユーザーのグループメンバーシップを確認できない

特定の信頼されるユーザーが特定の IdM グループ、外部または POSIX グループに関連付けられていることを確認する方法はありません。

Active Directory ユーザーの (リモート) Active Directory グループメンバーシップを表示できない

Linux システムユーザーの場合、ユーザーのローカルグループの関連付けは、**id** コマンドを使用して表示できます。ただし、Active Directory グループメンバーシップは、Samba ツールで設定されている場合でも Active Directory ユーザーの **id** では表示されません。

wbinfo コマンドを使用すると Active Directory ユーザーの SID を取得し、その後にその SID に関連付けられたグループを表示することができます。

```
[root@ipaserver ~]# wbinfo -n ADDOMAIN\\jsmith
S-1-5-21-1689615952-3716327440-3249090444-1104 SID_USER (1)

[root@ipaserver ~]# wbinfo --user-domgroups=S-1-5-21-1689615952-
3716327440-3249090444-1104
S-1-5-21-1689615952-3716327440-3249090444-513
S-1-5-21-1689615952-3716327440-3249090444-1106
```

id を使用する同じクエリーでは、Active Directory グループメンバーシップ情報ではなく、ユーザー情報のみを表示します。

```
[root@ipaserver ~]# id ADDOMAIN\\jsmith
uid=1921801104(jsmith@adexample.com) gid=1921801104(jsmith@adexample.com)
groups=1921801104(jsmith@adexample.com)
```

注記

この対策としては、所定の Active Directory ユーザーとして ssh を実行し、IdM クライアントマシンに接続します。初回の正常なログイン後に、Active Directory グループメンバーシップは **id** 検索で検出され、その情報が返されます。

```
[root@ipaserver ~]# id ADDOMAIN\jsmith
uid=1921801107(jsmith@adexample.com)
gid=1921801107(jsmith@adexample.com)
groups=1921801107(jsmith@adexample.com),129600004(ad_users),192180051
3(domain users@adexample.com)
```

5.2. 信頼をセットアップするための環境およびマシン要件

信頼契約を設定する **前に**、Active Directory および IdM サーバー、マシンおよび環境がこのセクションに記載する要件および設定条件を満たしていることを確認します。

5.2.1. サポートされる Windows プラットフォーム

信頼関係は、以下の Windows サーバーのバージョンを使用して設定することができます。

- Windows Server 2008 R2
- Windows Server 2012 R2

5.2.2. ドメインおよびレルム名

IdM の DNS ドメイン名および Kerberos レルム名は、Active Directory の DNS ドメイン名および Kerberos レルム名とは **異なる名前にする必要があります**。

5.2.3. NetBIOS 名

NetBIOS 名は、ドメイン名の左端のコンポーネントです。たとえば、ドメインが **linux.example.com** の場合、NetBIOS 名は **linux** になり、ドメイン名が単純に **example.com** の場合は **example** になります。NetBIOS 名は Active Directory ドメインを識別するために重要であり、IdM ドメインが Active Directory DNS のサブドメイン内にある場合に IdM ドメインおよびサービスを識別するために重要になります。

IdM ドメインおよび Active Directory ドメインにはそれぞれ異なる NetBIOS 名を設定する必要があります。

5.2.4. 統合 DNS

Active Directory サーバーおよび IdM サーバーはいずれも、独自の DNS サービスを実行するように設定する必要があります。

5.2.5. 統合された証明機関

Active Directory と Identity Management はどちらも統合された証明書サービスで設定される必要があります。

5.2.6. ファイアウォールおよびポート

信頼関係の下では、Active Directory サーバーおよび IdM サーバーで IdM サーバーのインストールに必要なシステムポートを開いておく必要があります。

IdM バックエンド LDAP サーバーは、 Active Directory ドメインコントローラーによって **アクセスできないようにする必要があります**。IdM サーバーホストの関連付けられたポート (389 および 636) が Active Directory ドメインコントローラーに対してシャットダウンされていることを確認します。389 および 636 ポートはドメインコントローラーに対してのみシャットダウンされている必要があることに注意してください。ドメインコントローラー以外の場合は、それらのポートを開いておく必要があります。

IdM で必要とされるポートの一覧については、『Linux Domain Identity, Authentication, and Policy Guide』の該当する章 を参照してください。

信頼関係が機能するために必要なポートを 表5.2「信頼に必要なポート」 に一覧表示します。

ポートを開くには、**firewalld** サービスが実行されている必要があります。システム起動時に **firewalld** が起動するように設定するには、以下を実行します。

```
[root@server ]# systemctl enable firewalld.service
```

firewalld 設定は、必要な IdM ポートへのアクセスを許可したり、各 Active Directory ホストが IdM LDAP ポートへのアクセスを拒否するために必要です。使用される **firewalld** ゾーンが **public** であることを仮定した場合に **firewalld** 設定を設定するには以下を実行します。

1. 各 Active Directory ホストについて LDAP ポートへのアクセスを制限するルールを追加します。

```
[root@server ~]# firewall-cmd --permanent --zone=public --add-rich-
rule='rule family="ipv4" source address="ad_ip_address" service
name="ldap" reject'
 [root@server ~]# firewall-cmd --permanent --zone=public --add-
rich-rule='rule family="ipv4" source address="ad_ip_address"
service name="ldaps" reject'
```

2. IdM で要求されるサービスへのポートを開きます。

```
[root@server ~]# firewall-cmd --permanent --zone=public --add-port=
{80/tcp,443/tcp,389/tcp,636/tcp,88/tcp,464/tcp,53/tcp,88/udp,464/u
dp,53/udp,123/udp}
```

3. 信頼関係に必要なサービスへのポートを開きます。

```
[root@server ~]# firewall-cmd --permanent --zone=public --add-port=
{138/tcp,139/tcp,445/tcp,138/udp,139/udp,389/udp,445/udp}
```

4. **firewalld** 設定を再読み込みし、変更が即時に適用されるようにします。

```
[root@server ~]# firewall-cmd --reload
```

表5.2 信頼に必要なポート

サービス	ポート	タイプ
NetBIOS-DGM	138	TCP および UDP
NetBIOS-SSN	139	TCP および UDP

サービス	ポート	タイプ
LDAP	389	UDP
Microsoft-DS	445	TCP および UDP

5.2.7. IPv6 設定

IdMシステムでは、IPv6 がカーネルで **有効にされている必要があります**。IPv6 が無効にされている場合、IdM サービスで使用される CLDAP プラグインは初期化に失敗します。

5.2.8. 時計の設定

Active Directory サーバーと IdM サーバーの両方でそれらの時計が同期されている必要があります。

5.2.9. サポートされているユーザー名の形式

ユーザー名のマッピングはローカル SSSD クライアントで実行されます。Python 正規表現は、ユーザー名とそれが属するドメインを識別するために SSSD で使用されます。

SSSD のデフォルトで、ユーザー名の形式は **name@domain** 形式で定義されます。これは正規表現を使用します。

```
re_expression = (?P<name>[^@]+)@?(?P<domain>[^@]*$)
```

ただし、Active Directory は名前形式のいくつかの異なるタイプをサポートします。そのため、IdM バックエンドまたは Active Directory バックエンド用の SSSD 設定ファイルの ***re_expression*** パラメーターはより複雑な式を使用します。

```
re_expression = (((?P<domain>[^\\]+)\\(?P<name>.+$))|((?P<name>[^@]+)@(?
P<domain>.+$))|(^(?P<name>[^@\\]+)$))
```

複数の形式のユーザー名がサポートされます。

- **username**
- **username@domain.name**
- **DOMAIN\username**

注記

追加の SSSD パラメーター ***default_domain_suffix*** を使用して、ユーザー名のデフォルトドメイン値を指定できます。たとえば、すべてのユーザーが **adexample.com** の信頼される Active Directory ドメインにあり、アイデンティティーバックエンドが **ipa.example.com** の IdM ドメインである場合、***default_domain_suffix*** パラメーターは値 **adexample.com** を使用して設定できます。ドメイン値がユーザー名と共に明示的に設定されない限り、すべてのユーザーがこのユーザードメインに属することが自動的に想定されます。

5.3. Active Directory ユーザー用の IdM グループの作成

ユーザーグループは、アクセス権限、ホストベースのアクセス制御、sudo ルールおよび IdM ユーザーの他の制御を設定するために必要です。これらのグループは、アクセスを制限するだけでなく、IdM ドメインリソースへのアクセスを付与する際のベースになります。

「Active Directory セキュリティーオブジェクトおよび信頼」で説明されているように、Active Directory ユーザーは、一種のデイジーチェーンの形で IdM ドメインに追加されます。それらは IdM 外部グループ (つまり POSIX 以外のグループ) に追加され、次にその外部グループはローカル POSIX グループにメンバーとして追加されます。その後に IdM POSIX グループは Active Directory ユーザーのユーザー/ロール管理に使用されます。

注記

Active Directory ユーザーグループを IdM 外部グループのメンバーとして追加することもできます。これにより、ユーザーおよびグループ管理を単一のレルム (Active Directory) 内で維持し、Windows ユーザーのポリシーを定義することがより容易になる場合があります。

1. **オプション**: IdM レルムで Active Directory ユーザーを管理するために使用する Active Directory ドメインのグループを作成または選択します。(複数のグループを使用し、IdM 側の異なるグループに追加することができます。)

2. Active Directory ユーザー用に IdM ドメインの **外部** グループを作成します。**--external** 引数が使用されていることは、このグループに IdM ドメイン外からのメンバーが含まれることを示します。以下が例になります。

```
[root@ipaserver ~]# ipa group-add --desc='AD users external map'
ad_users_external --external
--------------------------------
Added group "ad_users_external"
--------------------------------
  Group name: ad_users_external
  Description: AD users external map
```

3. 実際に IdM ポリシーを管理するために **POSIX** グループを作成します。

```
[root@ipaserver ~]# ipa group-add --desc='AD users' ad_users
----------------------
Added group "ad_users"
----------------------
  Group name: ad_users
  Description: AD users
  GID: 129600004
```

4. Active Directory ユーザーまたはグループを外部メンバーとして IdM 外部グループに追加します。Active Directory メンバーは、**DOMAIN\group_name** または **DOMAIN\username** などのその完全修飾名で識別されます。次に Active Directory のアイデンティティーはユーザーまたはグループの Active Directory SID にマップされます。

   例: Active Directory グループの場合

```
[root@ipaserver ~]# ipa group-add-member ad_users_external --
external "AD\Domain Users"
 [member user]:
 [member group]:
  Group name: ad_users_external
```

```
  Description: AD users external map
  External member: S-1-5-21-3655990580-1375374850-1633065477-513
SID_DOM_GROUP (2)
-------------------------
Number of members added 1
-------------------------
```

5. 外部 IdM グループを POSIX IdM グループにメンバーとして追加します。以下が例になります。

```
[root@ipaserver ~]# ipa group-add-member ad_users --groups
ad_users_external
  Group name: ad_users
  Description: AD users
  GID: 129600004
  Member groups: ad_users_external
-------------------------
Number of members added 1
-------------------------
```

5.4. 信頼の維持

信頼設定にはいくつかの層があります。IdM とそのピア Active Directory 間には直接の信頼契約があります。また、 IdM には多数のバックエンド設定が行われます。IdMが信頼をサポートするように設定される場合、数多くの異なる種類の設定領域が作成されます。

- Windows ドメイン内で IdM を識別するために使用されるグローバル信頼設定 (SID など)
- Active Directory で識別される DNS ドメイン。これらは IdM DNS ゾーン設定にプルされます (レルムドメイン)
- Kerberos 信頼設定 (信頼ドメインの個別の信頼契約)
- Windows ユーザーの IdM ドメインへの参加時に Windows ユーザーに UID および GID 番号を割り当てるために使用する、IdM サーバーごとに割り当てられる利用可能な ID 範囲。

5.4.1. グローバル信頼設定の編集

`ipa-adtrust-install` が実行されると、Active Directory ドメインとの信頼を作成するために必要な IdM ドメインのバックグランド情報が自動的に設定されます。外部信頼の場合でも、Active Directory ドメインは、その信頼されるピアにセキュリティー ID やドメイン ID などの特定の設定属性があると仮定します。それらの属性は、Active Directory に準拠するように IdM サーバー用に作成されます。

グローバル信頼設定には 5 つの属性が含まれます。

- Windows スタイルのセキュリティー ID
- ドメイン GUID
- Kerberos ドメイン名
- Windows ユーザーを追加するデフォルトグループ

上記の属性は、その一部 (NetBIOS 名およびデフォルトグループ) のみを編集できます。GUID および SID は自動生成され、Kerberos レルム名は IdM 設定から取られます。

信頼設定は **`cn=domain,cn=ad,cn=etc,dc=example,dc=com`** サブツリーに保存されます。

5.4.1.1. NetBIOS 名の変更

NetBIOS 名はドメイン名の左端のコンポーネントです。これは、ドメインコントローラーのホストシステムの主な識別子です。IdM が信頼を設定するために有効にされると、NetBIOS 名は Active Directory トポロジー内の互換性維持のために IdMサーバーに設定されます。これは **ipa-adtrust-install** コマンドで設定されます。設定を変更するには、**ipa-adtrust-install** コマンドを返します。

```
[root@ipaserver ]# ipa-adtrust-install --netbios-name=NEWBIOSNAME -a
secret
```

-a オプションにより、IdM の管理者パスワードが指定されます。

5.4.1.2. Windows ユーザーのデフォルトグループの変更

IdM には **自動メンバーシップ** 機能があります。これは、新規ユーザーを特定グループに自動的に追加します。Windows ユーザーを **デフォルト SMB グループ** (IdM 信頼設定の一部として作成されるグループ) に自動的に追加するデフォルトの自動メンバーシップルールがあります。これは、他の自動メンバーシップルールが Windows ユーザーに適用されない場合に使用されるフォールバックグループです。

デフォルトグループは変更することができます。これは Windows ユーザーに複数の異なる外部グループが追加される場合にとくに役立ちます (「Active Directory ユーザー用の IdM グループの作成」)。**このグループは、すべての Windoes ユーザーにグローバルに使用されるフォールバックまたはデフォルトグループです。デフォルトを使用せずに特定グループを異なる Windows ユーザーに適用するために他のルールを設定することもできます。**

デフォルトグループは、**trustconfig-mod** コマンドを使用して設定できます。

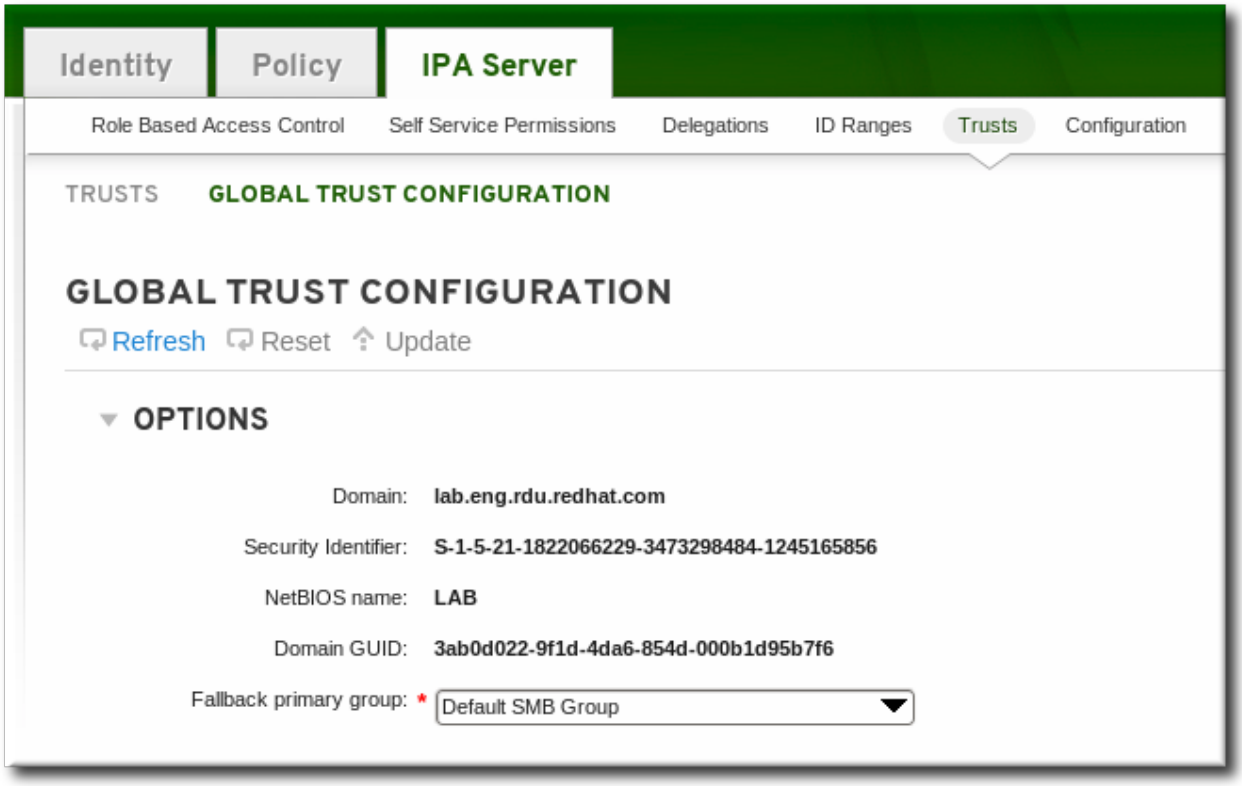
```
[root@server ~]# kinit admin
[root@server ~]# ipa trustconfig-mod --fallback-primary-group="Example
Windows Group"
```

これは IdM web UI を使用して変更することもできます。

1. IdM web UI を開きます。

   ```
   https://ipaserver.example.com
   ```

2. **IPA Server** メインタブを開いてから、**Trusts** サブタブを選択します。
3. **Global Configuration** サブタブで、**Fallback primary group** に一覧表示されるすべての IdM グループから新規グループを選択します。



4. **Update** リンクをクリックして、新規設定を保存します。

5.4.2. 信頼ドメインの検出、有効化、および無効化

推移する信頼とは、信頼パスがドメインのチェーンに従って形成されることを意味します。ドメイン A がドメイン B を信頼し、ドメイン B がドメイン C を信頼する場合、ドメイン A はドメイン C を暗黙的に信頼します。ドメイン A のドメイン B との間の信頼は推移し、ドメイン C への信頼パスが形成されます。

IdM にはフォレストのルートドメインとの間に信頼があり、そのサブドメインおよび信頼されるドメインにすべてその信頼に暗黙的に組み込まれます。IdM は、Windows ユーザーがフォレストの任意の場所から IdM リソースへのアクセスを試行する際にそのトポロジーに従います。各ドメインおよびサブドメインは IdM 信頼設定の **信頼ドメイン** です。各ドメインは、信頼サブツリーのそれぞれのエントリー **cn=subdomain,cn=trust_name,cn=ad,cn=trusts,dc=example,dc=com** に保存されます。

IdM は、信頼が最初に設定される際に完全な Active Directory トポロジーの検出およびそのマッピングを試行します。ただし、場合によってはそのトポロジーを手動で取得することが必要であるか、またはそれが望ましい場合があります。これは **trust-fetch-domains** コマンドで実行されます。

```
[root@ipaserver ~]# kinit admin
[root@ipaserver ~]# ipa trust-fetch-domains adexample.com
-----------------------------------------
List of trust domains successfully refreshed
-----------------------------------------
  Realm name: test.adexample.com
  Domain NetBIOS name: TEST
  Domain Security Identifier: S-1-5-21-87535643-5658642561-5780864324

  Realm name: users.adexample.com
  Domain NetBIOS name: USERS
  Domain Security Identifier: S-1-5-21-91314187-2404433721-1858927112

  Realm name: prod.adexample.com
  Domain NetBIOS name: PROD
```

```
  Domain Security Identifier: S-1-5-21-46580863-3346886432-4578854233
----------------------------
Number of entries returned 3
----------------------------
```

注記

共有秘密値で信頼を追加する際には、AD フォレストのトポロジーを手動で取得する必要があります。"ipa trust-add ad.domain --trust-secret" コマンドを実行した後に「AD Domains and Trusts (Active Directory ドメインと信頼関係)」ツールでフォレスト信頼プロパティーを使用し、AD 側で入力方向の信頼を検証します。次に "ipa trust-fetch-domains ad.domain" コマンドを実行します。IdM は使用可能になる信頼についての情報を受信します。

トポロジーが取得されると (自動検出または手動検出)、そのトポロジーの個別のドメインおよびサブドメインを有効にしたり、無効にしたり、または IdM 信頼設定内で完全に削除したりできます。

たとえば、特定サブドメインのユーザーが IdM リソースを使用できないようにするには、その信頼ドメインを無効にします。

```
[root@ipaserver ~]# kinit admin
[root@ipaserver ~]# ipa trustdomain-disable test.adexample.com
----------------------------------------
Disabled trust domain "test.adexample.com"
----------------------------------------
```

その信頼ドメインは、**trustdomain-enable** コマンドを使用して再度有効にできます。

ドメインがトポロジーから永久的に削除される必要がある場合、IdM 信頼設定からこれを削除することができます。

```
[root@ipaserver ~]# kinit admin
[root@ipaserver ~]# ipa trustdomain-del prod.adexample.com
------------------------------------------------------------------------
Removed information about the trusted domain " "prod.adexample.com"
------------------------------------------------------------------------
```

5.4.3. DNS レルムの表示および管理

信頼が設定される際に、Active Directory DNS 設定は IdM DNS 設定に追加され、それぞれのレルムは特別な **レルムドメイン** として追加されます。各ドメインは、IdM ディレクトリーの**cn=Realm Domains,cn=ipa,cn=etc,dc=example,dc=com** サブツリーに保存されます。

これらのレルムドメインは自動的に追加されるため、通常 DNS ゾーンを追加したり、変更したりする必要はありません。設定されたレルムドメインの一覧は、**realmdomains-show** コマンドを使用して表示することができます (IdM で設定されるすべての DNS ゾーンを一覧表示するのではない)。

```
[root@ipaserver ~]# kinit admin
[root@ipaserver ~]# ipa realmdomains-show
 Domain: ipa.example.org, ipa.example.com, example.com
```

単一レルムドメインを設定に追加する必要がある場合は、**--add-domain** オプションを使用して実行できます。

```
[root@ipaserver ~]# kinit admin
[root@ipaserver ~]# ipa realmdomains-mod --add-domain=adexample.com
 Domain: ipa.example.org, ipa.example.com, example.com, adexample.com
```

単一ドメインは **--del-domain** オプションを使用して削除することができます。

ドメインの一覧に対して複数の変更が行われる場合、**--domain** オプションを使用して一覧自体を変更し、置き換えることができます。

```
[root@ipaserver ~]# ipa realmdomains-mod --domain=
{ipa.example.org,adexample.com}
```

5.4.4. 推移的な信頼における UID/GID 番号範囲の追加

Windows システムは、Linux システムとは異なる方法で ID 番号を処理します。ユーザーが Linux で作成されると、そのユーザーにはユーザー ID 番号が割り当てられ、そのユーザーのプライベートグループが作成されます。プライベートグループの UI 番号は ID 番号と同じです。Linux ではこれに関する競合は生じません。ただし Windows では、セキュリティー ID はドメイン内のすべてのオブジェクトについて一意であるため、ユーザーとグループの ID が同じである場合は競合が生じます。

POSIX 属性 (***uidNumber*** および ***gidNumber*** を含む) が Active Directory のグローバルカタログからプルされる場合、ID 番号は Active Directory 環境内で固有であるため、固有な番号になります。これは、信頼の作成時に **trust-add** コマンドで設定される**ipa-ad-trust-posix** 範囲タイプです。基本的に ID 検証または範囲 は不要です。

ただし SID/ユーザー名のマッピングを使用して SID が IdM で生成される場合、Windows アイデンティティーと IdMユーザーおよびグループの両方の ID 範囲には、固有の重複しない有効な範囲がなければなりません。これが **ipa-ad-trust** 範囲タイプです。

Active Directory ドメインが信頼に基づいて IdM に追加される際に、固有の ID 範囲が各 Active Directory ドメインに自動的に作成されます。ただし、Active Directory および IdM は **推移する** 信頼で機能できます。推移する信頼では、レルム A がレルム B を信頼し、レルム B がレルム C を信頼する場合、レルム A もレルム C も信頼するというデイジーチェーンが展開されます。信頼の設定時に、範囲は信頼契約で指定されるドメインについてのみ追加されます。推移的に信頼されるドメインの範囲は手動で追加する必要があります。

ID 範囲を追加するには、POSIX 範囲のベース ID (開始番号)、RID の開始番号 (SID の右端にある番号)、範囲のサイズおよびドメイン SID を設定します (信頼について複数のドメインが設定される可能性があるため)。

```
[root@server ~]# kinit admin
[root@server ~]# ipa idrange-add --base-id=1200000 --range-size=200000 --
rid-base=0 --dom-sid=S-1-5-21-123-456-789 trusted_dom_range
```

ベース ID は POSIX ID の開始番号です。RID は、競合を避けるためにベース ID に追加する範囲です。ベース ID が 1200000 で RID が 1000 の場合、結果として生成される ID 番号は 1201000 になります。

5.5. IdM マシンに解決可能な名前があるかどうかの確認

「異なる DNS 信頼環境」で説明されているように、Identity Management および Active Directory DNS ドメイン内のすべてのホスト名は、DNS 設定にかかわらず、信頼されるサービスが確実に機能できるよう完全に解決可能である必要があります。

信頼を設定した後に、Identity Management サーバーが IdM と Active Directory レルムの両方で解決可能であることを確認します。

最初に IdM でホストされるサービスが IdM ドメインで解決可能であることを確認します。

1. UDP 経由で Kerberos レコードの DNS クエリーを実行します。

```
[root@ipaserver ~]# dig +short -t SRV @10.1.1.1
_kerberos._udp.ipa.example.com.
0 100 88 ipamaster1.ipa.example.com.
```

2. TCP 経由で LDAP レコードの DNS クエリーを実行します。

```
[root@ipaserver ~]# dig +short -t SRV @10.1.1.1
_ldap._tcp.ipa.example.com.
0 100 389 ipamaster1.ipa.example.com.
```

3. Kerberos レルム名を使用して TXT レコードの DNS クエリーを実行します。これは Identity Management サーバーの Kerberos レルムと一致している必要があります。

```
[root@ipaserver ~]# dig +short -t TXT @10.1.1.1
_kerberos.ipa.example.com.
```

Active Directory サーバーで、IdM でホストされるサーバーおよびサービスのすべてが解決可能であることを確認します。

Active Directory には、DNS 設定を照会する **nslookup.exe** というユーティリティーがあります。

1. **nslookup.exe** ユーティリティーを設定して、サービスレコードを検索します。

```
C:\>nslookup.exe
> set type=SRV
```

2. サービスの名前および (オプションで) IdM ネームサーバーの IP アドレスを入力します。

```
> _ldap._tcp.ipa.example.com 10.1.1.1
Server:   [10.1.1.1]
Address:   10.1.1.1

_ldap._tcp.ipa.example.com       SRV service location:
          priority                = 0
          weight                  = 100
          port                    = 389
          svr hostname   = ipaserver.ipa.example.com
ipaserver.ipa.example.com      internet address = 10.1.1.1
```

3. IdM Kerberos レルム設定を検査するためにサービスタイプを TXT に変更します [5]

```
> set type=TXT
```

4. Kerberos レコードを照会します。

```
> _kerberos.ipa.example.com. 10.1.1.1
```

Active Directory は DNS 参照の結果をキャッシュします。現在のキャッシュは **ipconfig /displaydns** を実行して表示でき、キャッシュは **ipconfig /flushdns** を実行して削除できます。

5.6. サービスの PAC タイプの設定

IdMリソースについては、Active Directory ユーザーがサービスのチケットを要求する場合に IdM はその要求を Active Directory に転送して、ユーザー情報を取得します。ユーザーの Active Directory グループ割り当てに関連付けられたアクセスデータが Active Directory によって送り戻され、Kerberos チケットに組み込まれます。

Active Directory のグループ情報は、**privileged access certificates** または MS-PAC と呼ばれる特殊なデータセットとして Active Directory ユーザーの各 Kerberos チケットの識別子の一覧に保存されます。PAC のグループ情報は Active Directory グループにマップされてから、対応する IdM グループにマップされ、アクセスの判別が行われます。

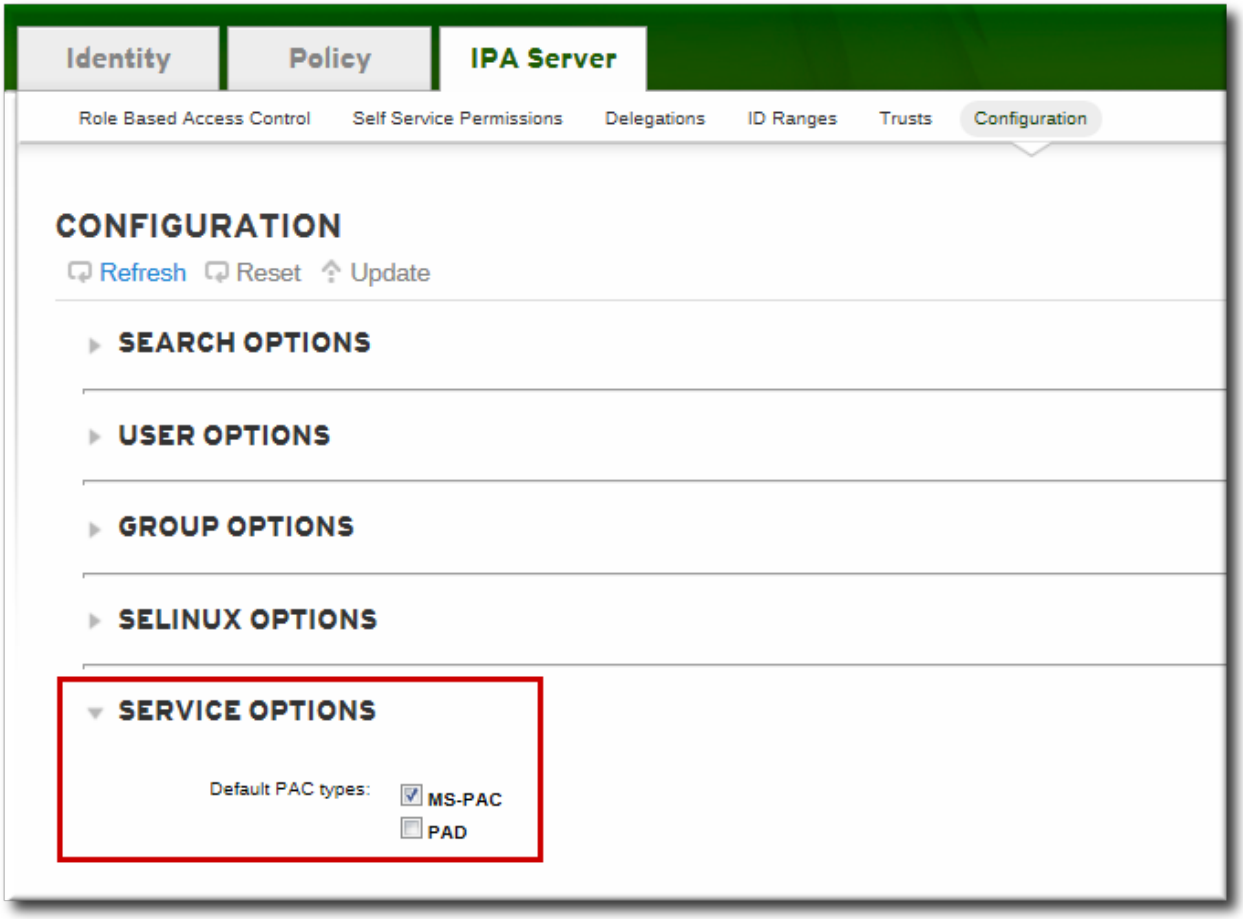
POSIX システムには、**POSIX authorization data (POSIX 認可データ)** 要素という同様のデータセットがあります。PAD には、 PAC と同様にユーザーのグループベースの認証データが含まれます。アクセスデータは初回の認証要求への返信として返されるため、グループデータを取得するために追加のクロスレルム通信は必要ありません。

IdM サービスは、ドメインサービスに対するユーザー認証の初回試行時の認証要求用に PAC、PAD、またはその両方を生成するように設定できます。

5.6.1. デフォルト PAC タイプの設定

IdM サーバー設定は、サービスについてデフォルトで生成される PAC タイプを定義します。グローバル設定は、特定サービスのローカル設定を変更して上書きできます。

1. **IPA Server** タブを開きます。
2. **Configuration** サブタブを選択します。
3. **Service Options** 領域にスクロールします。

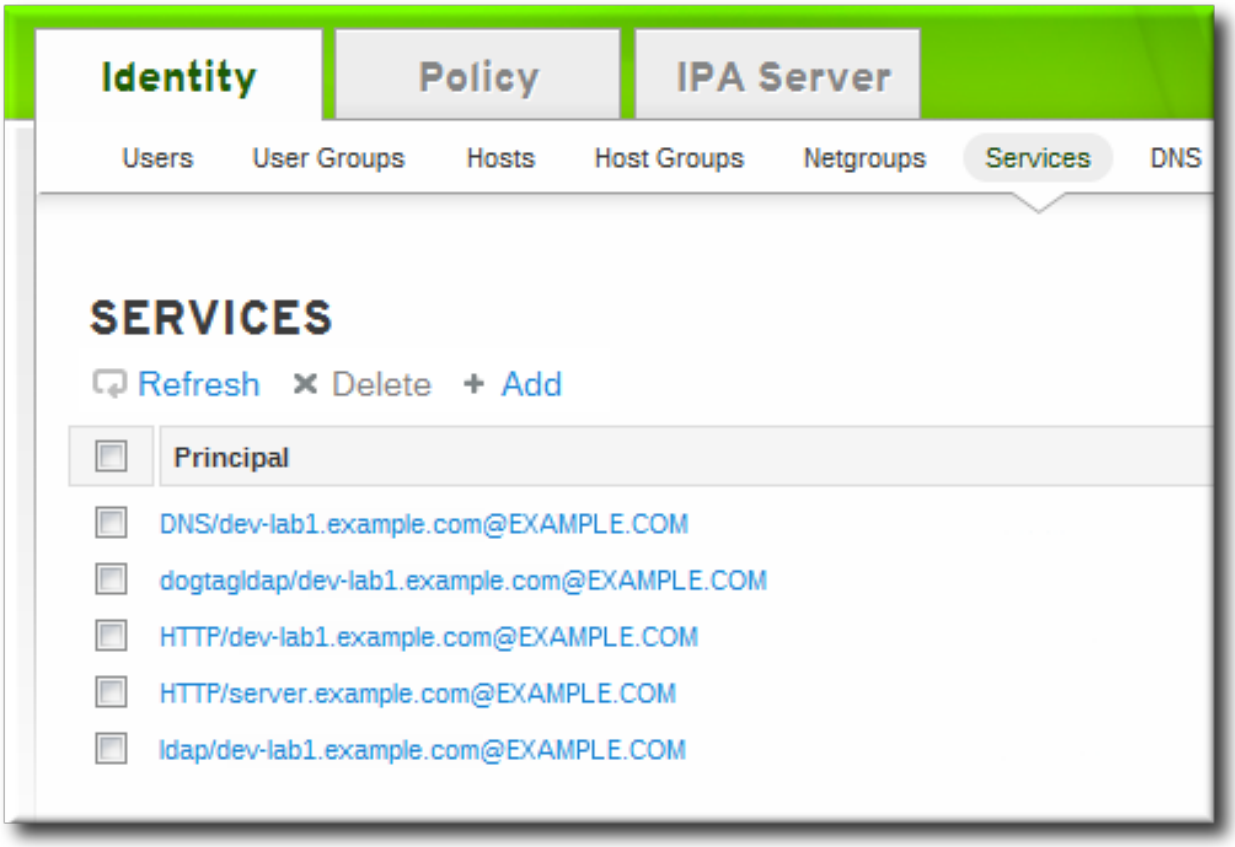


4. 使用する PAC タイプのチェックボックスを選択します。どちらの PAC タイプも選択されている場合はいずれも Kerberos チケットに追加されます。
   - **MS-PAC** では、Active Directory サービスで使用できる証明書を追加します。
   - **PAD** では POSIX (Windows 以外の) システムで使用できる証明書を追加します。
   - チェックボックスが選択されていない場合、PAC は Kerberos チケットに追加されません。
5. 変更を保存するには、ページの上部にある **Update** リンクをクリックします。

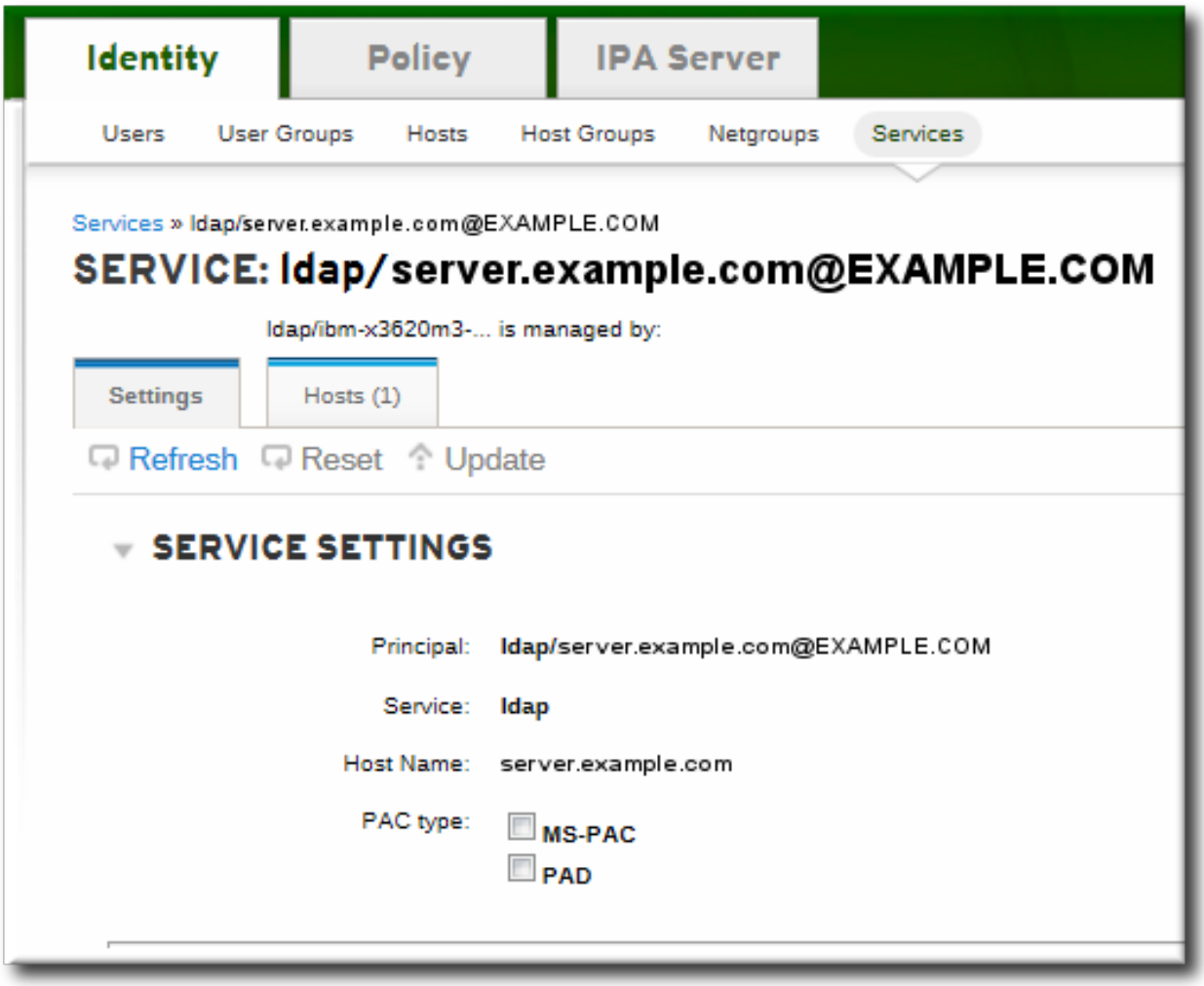
5.6.2. サービスの PAC タイプの設定

グローバルポリシーは、サービスに明示的な設定がない場合にサービスに使用する PAC タイプを設定します。ただし、グローバル設定はローカルサービス設定で上書きされる可能性があります。

1. **Identity** タブを開き、**Services** サブタブを選択します。
2. 編集するサービスの名前をクリックします。



3. **`Service Settings`** 領域では、使用する PAC タイプのチェックボックスを選択します。どちらの PAC タイプも選択されている場合はいずれも Kerberos チケットに追加されます。



- **MS-PAC** では、Active Directory サービスで使用できる証明書を追加します。
- **PAD** では POSIX (Windows 以外の) システムで使用できる証明書を追加します。
- チェックボックスが選択されていない場合、PAC は Kerberos チケットに追加されません。

4. 変更を保存するには、ページの上部にある **Update** リンクをクリックします。

5.7. IdM リソースのために Active Directory マシンから SSH を使用

信頼が設定されると、Active Directory ユーザーは SSH およびそれらの Active Directory 資格情報を使用して、IdM ホスト上のマシン、サービスおよびファイルにアクセスすることができます。

注記

Windows マシンで PuTTY を使用する際は、GSS-API 資格情報の委任が有効にされていることを確認します。

5.7.1. SSH におけるユーザー名の要件

SSH の使用時の 1 つの重要なポイントになるのはユーザー名です。ユーザー名は以下のいくつかの基準を満たしている必要があります。

- ユーザー名には **ad_user@ad_domain** 形式を使用する必要があります。
- ドメイン名自体は小文字にする必要があります。これは Kerberos プリンシパルのマッピングに必要です。
- ユーザー名の大文字/小文字の区別は、Active Directory のユーザー名の大文字/小文字の区別に完全に一致している必要があります。 **jsmith** と **JSmith** は、大文字/小文字の使用が異なるために異なるユーザーと見なされます。

5.7.2. パスワードなしの SSH の使用

適切な Kerberos チケットが取得されている場合でも、SSH を使用することで、依然として Active Directory ドメインユーザーのユーザーパスワードを求めるプロンプトが出されます。SSH はユーザー名を **-l** で指定しますが、Kerberos チケットには、ユーザー名ではなく Kerberos プリンシパルが含まれます。システムは、ユーザーがチケットを持つかどうかを調べるために、提供されるローカルユーザー名とプリンシパル名を比較する方法を要求します。 **.k5login** ファイルは、ローカルユーザーを Kerberos プリンシパルにマップする簡単な方法を提供します。このファイルはローカルユーザーのホームディレクトリーにあり (ユーザーは SSH の **-l** オプションで識別される)、そのユーザーの Kerberos プリンシパルを一覧表示します。認証しているユーザーが既存 Kerberos チケットのプリンシパルと一致する場合、ユーザーはパスワードを求められることなく、認証用にチケットを使用してログインできます。

Kerberos 認証に切り替える (つまりパスワードが不要の SSH 認証を使用する) には、各 Active Directory ユーザーには Linux ホームディレクトリーに **.k5login** ファイルがなければなりません。このファイルには、ユーザーが使用する Kerberos プリンシパルの一覧のみが含まれます。プリンシパルには **user@REALM.COM**、 **AD.domain\user**、または **AD\user** などの「サポートされているユーザー名の形式」にあるいずれかの形式を使用することができます。

たとえば **ENGINEERING.ADREALM.COM** という名前の Active Directory レルムのユーザー**jsmith** の場合、以下のように **.k5login** がホームディレクトリーに置かれます。

```
/home/engineering.adrealm/jsmith/.k5login
```

ファイルの内容には、以下のような 2 つの異なるプリンシパル名が含まれます。

```
jsmith@ENGINEERING.ADREALM
ENGINEERING.ADREALM.COM\jsmith
```

.k5login ファイルでは大文字/小文字が区別されるため、複数のプリンシパルを異なる形式で、かつ大文字/小文字を区別して一覧表示します。

2 種類のプリンシパルがあるため、チケットを取得するにはいずれかの文字列を **kinit** と共に使用することができます。

.k5login man ページには詳細が記載されています。

5.8. Kerberos 対応 Web アプリケーションでの信頼の使用

いずれの既存の web アプリケーションも、信頼される Active Directory および IdM Kerberos レルムを参照する Kerberos 認証を使用できるように設定できます。詳細の Kerberos 設定ディレクティブは mod_auth_kerb モジュールの man ページに記載されています。

たとえば Apache サーバーの場合、Apache サーバーが IdM Kerberos レルムに接続する方法を定義するいくつかのパラメーターがあります。

- ***KrbAuthRealms*** ディレクティブはアプリケーションの場所を IdM ドメインの名前に指定します。これは必須です。
- ***Krb5Keytab*** は IdM サーバーキータブの場所を指定します。これは必須です。
- ***KrbServiceName*** はキータブに使用される Kerberos サービス名を設定します (HTTP)。このパラメーターの使用は推奨されています。
- Kerberos メソッドのディレクティブ (***KrbMethodNegotiate*** および ***KrbMethodK5Passwd***) は、有効なユーザーのパスワードベースの認証を有効にします。多くのユーザーを扱う場合の使いやすさのため、このパラメーターの使用をお勧めします。
- ***KrbLocalUserMapping*** ディレクティブは、通常の web ログイン (通常はアカウントの UID または共通名) を完全修飾ユーザー名 (**user@REALM.COM** 形式) にマップできるようにします。

  上記のパラメーターの使用を強くお勧めします。ドメイン名/ログイン名のマッピングがないと、web ログインにはドメインユーザーとは異なるユーザーアカウントが表示され、ユーザーには予測しないデータが表示されてしまいます。

  「サポートされているユーザー名の形式」では、複数の異なるサポートされているユーザー名の形式について説明しています。

例5.1 Apache Web アプリケーションの Kerberos 設定

```
<Location "/mywebapp">
   AuthType Kerberos
   AuthName "IPA Kerberos authentication"
   KrbMethodNegotiate on
   KrbMethodK5Passwd on
   KrbServiceName HTTP
   KrbAuthRealms IDM_DOMAIN
   Krb5Keytab /etc/httpd/conf/ipa.keytab
```

```
    KrbLocalUserMapping on
    KrbSaveCredentials off
    Require valid-user
</Location>
```

注記

Apache アプリケーション設定を変更した後に、Apache サービスを再起動します。

```
[root@ipaserver ~]# systemctl restart httpd.service
```

[5] 通常 Active Directory ドメインの TXT レコードはありません。

第6章 Kerberos クロスレルム認証のセットアップ

Kerberos v5 は複数のクライアントから構成されるレルムを作成します。レルムには信頼を設定でき、他の Kerberos ドメインと同様に Active Directory ドメインと統合できます。Kerberos 自体はシステムに依存しないため、数多くの異なる環境、システムおよびアプリケーションで機能します。

数多くの Linux 環境 (および混在環境) には、シングルサインオン、アプリケーションの認証およびユーザー管理用に Kerberos レルムが事前に実装されます。これにより、Linux 環境が Identity Management などのより構造化されたドメイン設定を使用しない場合などは Kerberos が Windows と linux の混在環境用の共通の統合パスとなる可能性があります。

6.1. 信頼関係

信頼 とは、あるレルム内のユーザーが別のドメインのリソースにアクセスできるように信頼されることを意味します。**それらのユーザーは別のレルムに実際に属しているかの様に機能します**。これは、両方のドメインで保持される単一プリンシパルの共有キーを作成することによって実行されます。

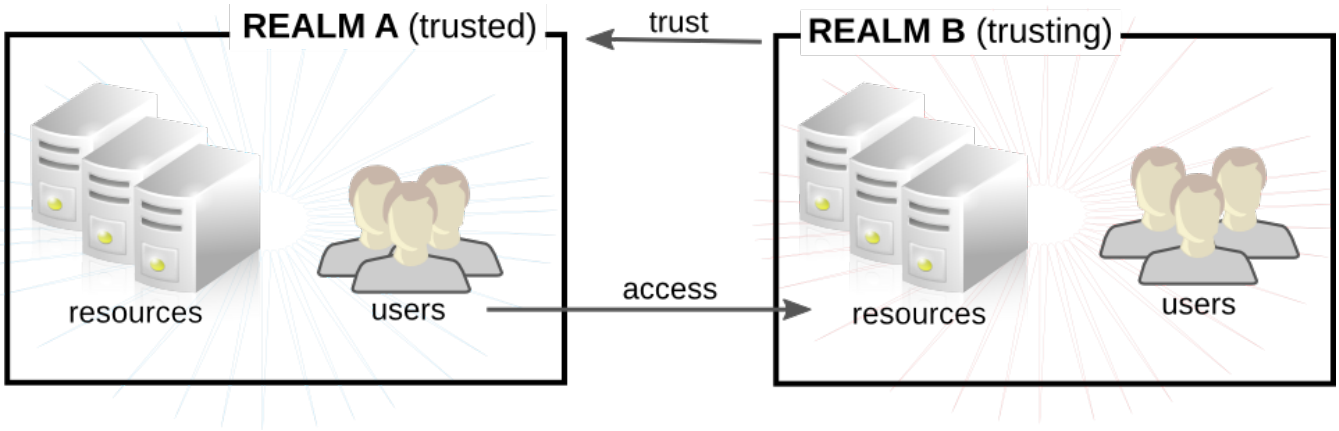


図6.1 基本的な信頼

図6.1「基本的な信頼」では、共有されるプリンシパルは Domain B (**krbtgt/B.EXAMPLE.COM@A.EXAMPLE.COM**) に属します。そのプリンシパルが Domain A にも追加されると、Domain A のクライアントは Domain B のリソースにアクセスできます。設定されるプリンシパルは両方のレルムに存在します。この共有されるプリンシパルには 以下の 3 つの特徴があります。

- このプリンシパルは両方のレルムに存在する。
- キーの作成時には、同じパスワードが両方のレルムで使用される。
- キーのキーバージョン番号は同一である (kvno)。

デフォルトでは **クロスレルム信頼は一方向** です。この信頼は、**B.EXAMPLE.COM** レルムが **A.EXAMPLE.COM** レルムのサービスに対して認証されるように自動的に双方向になる訳ではありません。他の方向で信頼を設定するには、両方のレルムが **krbtgt/A.EXAMPLE.COM@B.EXAMPLE.COM** サービスのキー (直前の例とは逆方向でマップされているエントリー) を共有する必要があります。

レルムには、信頼するレルムと信頼されるレルムの両方の信頼を複数含めることができます。Kerberos 信頼では、信頼はチェーンで推移します。Realm A が Realm B を信頼し、Realm B が Realm C を信頼する場合、Realm A も Realm C を暗黙的に信頼します。信頼は複数のレルムに推移します。これを **推移する** 信頼と言います。

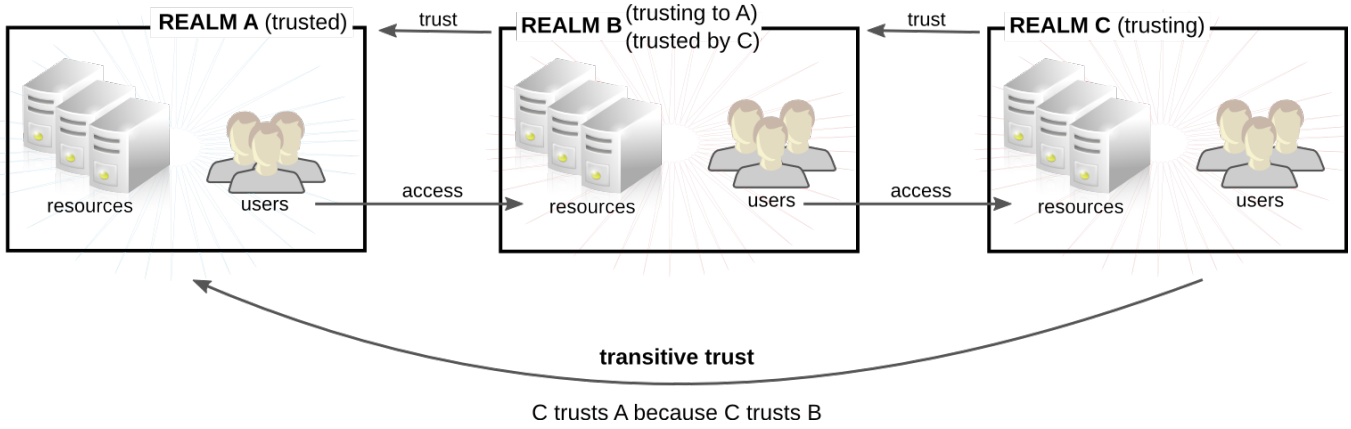


図6.2 推移的な信頼

注記

Kerberos 信頼はデフォルトで推移しますが、これは Windows ドメインの Kerberos 信頼については必ずしも推移するとは限りません。Windows では、他の Windows ドメインとの信頼は推移しますが、外部 (Windows 以外の) レルムとの信頼はデフォルトでは推移しません。ただし、これらを推移するように設定することはできます。

推移的な信頼の方向は **信頼フロー** と呼ばれています。信頼フローは、まずサービスが属するレルムを認識し、次にそのサービスにアクセスするためにクライアントが接続する必要のあるレルムを識別して定義する必要があります。

Kerberos プリンシパル名は **service/hostname@REALM** 形式で構成されます。**service** は通常、LDAP、IMAP、HTTP、またはホストなどのプロトコルです。**hostname** はホストシステムの完全修飾ドメイン名であり、**REALM** はそれが属する Kerberos レルムです。通常クライアントは、ホスト名または DNS ドメイン名をレルムにマップします。次にレルムは、DNS ドメイン名に何らかの方法で関連付けられます (レルムが **`/etc/krb5.conf`** の **`domain_realm`** セクションで明示的に定義されていない場合に限ります)。

信頼関係をスキャンする際に、Kerberos は各レルムがルートドメインとサブドメインからなる階層的な DNS ドメインのように構成されていると仮定します。つまり、信頼は共有ルートまで移動します。**ホップ** と呼ばれる各ステップには共有キーがあります。図6.3「同一ドメイン内の信頼」では、A は EXAMPLE.COM とキーを共有し、EXAMPLE.COM は B とキーを共有しています。

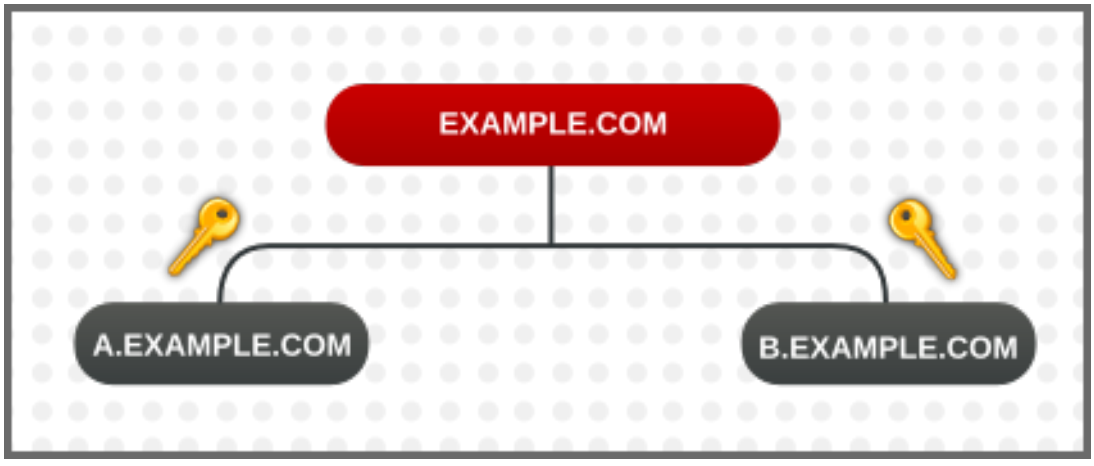


図6.3 同一ドメイン内の信頼

クライアントはレルム名を DNS 名として処理し、ルート名にたどり着くまで自らのレルム名から要素を取り除くことで信頼パスを判別します。次にクライアントは、サービスのレルムにたどり着くまで名前を先頭に追加していきます。

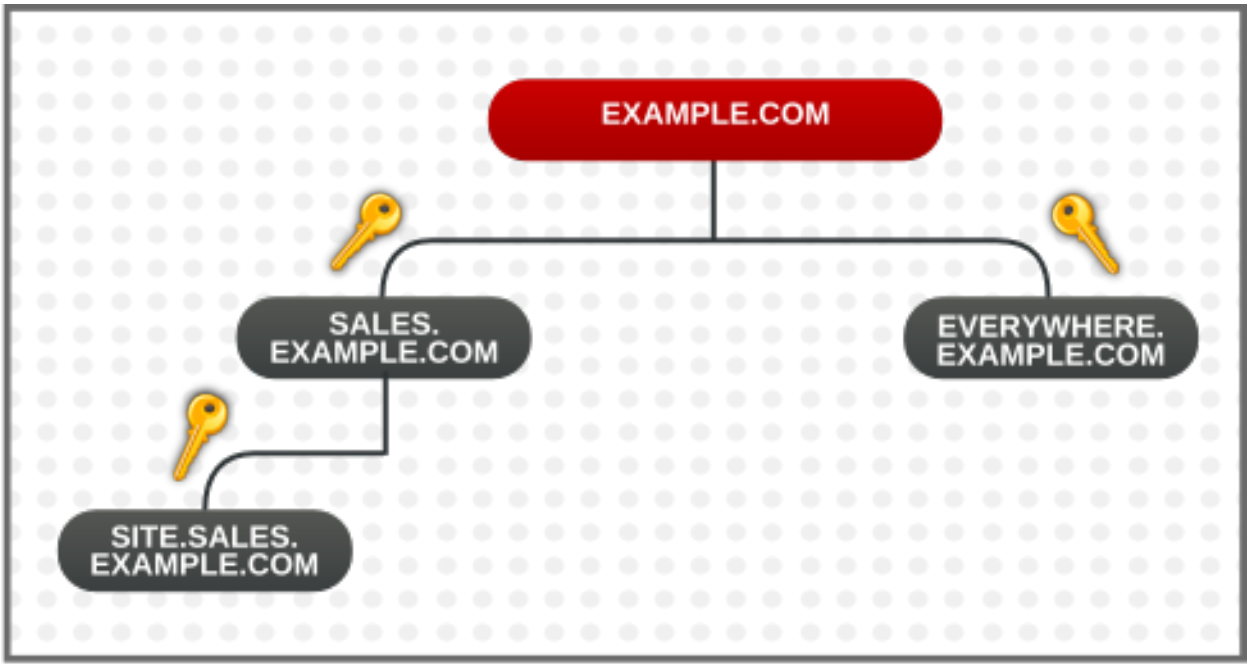


図6.4 同一ドメイン内の子/親の信頼

これは、信頼の推移的な性質を示しています。SITE.SALES.EXAMPLE.COM には SALES.EXAMPLE.COM との共有キーが 1 つだけあります。しかし一連の小規模な信頼により、SITE.SALES.EXAMPLE.COM から EVERYWHERE.EXAMPLE.COM への信頼の移動を可能にする大規模な信頼フローが形成されます。

その信頼フローは、共有キーをドメインレベルで作成することで、サイト間でサフィックスが共有されない場合でも全く異なるドメイン間で移動させることができます。

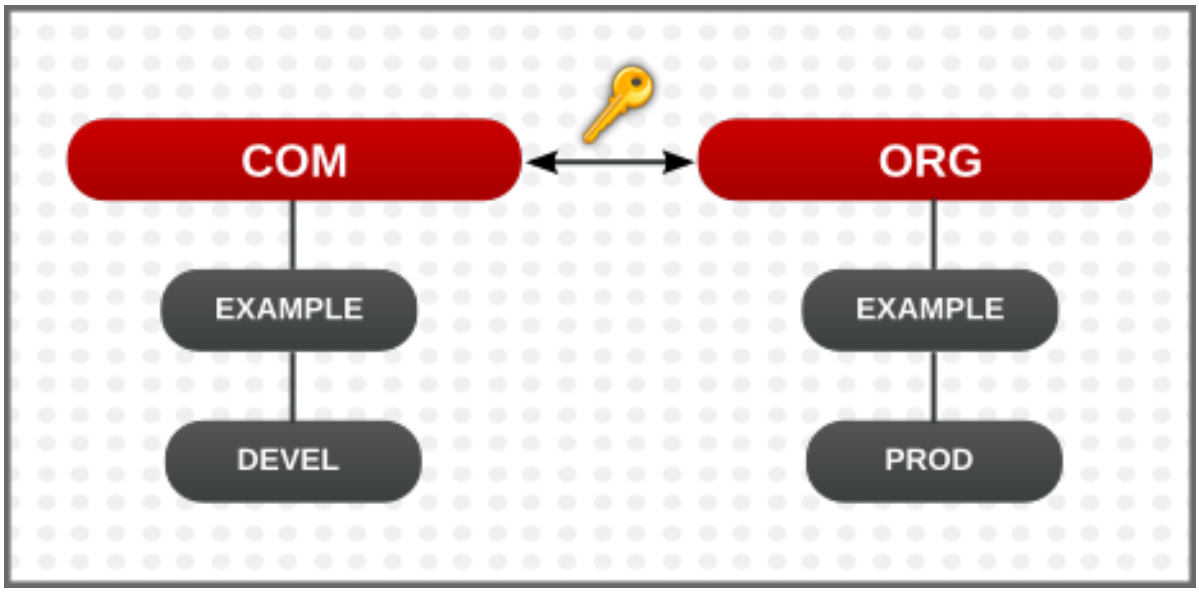


図6.5 複数の異なるドメインにおける信頼

フローを明示的に定義することで、ホップ数を減らして非常に複雑な信頼を表示することもできます。`/etc/krb5.conf` ファイルの `capaths` セクションでは異なるレルム間の信頼フローを定義します。

capaths セクションの形式は比較的単純です。クライアントがプリンシパルを持つ各レルムのメインエントリーがあり、各レルムセクションには、クライアントの資格情報の取得元となる中間レルムの一覧があります。

たとえば **A.EXAMPLE.COM** のレルム、および A から D までのホップのセットがあるとします。レルム A のクライアントはレルム B から資格情報を取得する必要があります (**.** は、クライアントが中間的なホップなしに資格情報を直接取得でき、これがない場合は、クライアントが階層を参照して資格情報へのアクセスを試行することを意味します)。次に、クライアントは C からの資格情報を取得するために B の資格情報を使用し、次に D の資格情報を取得するために C の資格情報を使用する必要があります。

```
[capaths]
A.EXAMPLE.COM = {
B.EXAMPLE.COM = .
C.EXAMPLE.COM = B.EXAMPLE.COM
D.EXAMPLE.COM = C.EXAMPLE.COM
}
```

6.2. レルム信頼のセットアップ

以下の例では、Kerberos レルムは **KRB.EXAMPLE.COM** であり、Active Directory レルムは **AD.EXAMPLE.COM** です。

1. **kadmin** を使用して、Kerberos の共有プリンシパルのエントリーを作成します。

   ```
   [root@server ~]# kadmin -r KRB.EXAMPLE.COM
   kadmin: add_principal krbtgt/AD.EXAMPLE.COM@KRB.EXAMPLE.COM
   Enter password for principal
   "krbtgt/AD.EXAMPLE.COM@KRB.EXAMPLE.COM":
   Re-enter password for principal
   "krbtgt/AD.EXAMPLE.COM@KRB.EXAMPLE.COM":
   Principal "krbtgt/AD.EXAMPLE.COM@KRB.EXAMPLE.COM" created.
   quit
   ```

2. レルム信頼は **Active Directory Domains and Trusts (Active Directory ドメインと信頼関係)** コンソールで設定されます。適切なドメインを選択し、新規のトラストを作成します。以下が使用する設定です。

   - **Trust Type** は **Realm** にします。
   - **Transitivity of Trust** は推移または非推移のいずれかにします。
   - **Direction of Trust** は **One-way: incoming** にします。これにより、Kerberos レルムの Active Directory ユーザーが信頼されます。

     上記により、一方向の信頼が作成されます。ここで、Active Directory ユーザーは Kerberos レルムで信頼されます。双方向の信頼を作成するには、信頼の方向を双方向に設定します。これについては、Microsoft TechNet 文書 で説明されています。
   - **Sides of Trust** は **This domain only** にします。
   - **Trust Password** には任意の値を使用できます。これは Kerberos の信頼を設定する際に使用する必要があります。

第7章 Active Directory および Identity Management ユーザーの同期

Red Hat Enterprise Linux Identity Management は、アクティブな **同期** を使用して Active Directory ドメインに保存されるユーザーデータと IdM ドメインに保存されるユーザーデータを組み合わせます。パスワードを含む重要なユーザー属性はサービス間でコピーされ、同期されます。

エントリーの同期は、Windows サーバーのディレクトリーデータに接続およびそれを取得するためにフックを使用するレプリケーションと同様のプロセスで実行されます。この機能は、Active Directory ドメインに追加設定を行うことなく、Identity Management ですぐに使用できます。

パスワードの同期は、Windows サーバーにインストールされ、Identity Management サーバーと通信する Windows サービスで実行されます。

7.1. サポートされる Windows プラットフォーム

同期は以下の Windows サーバーでサポートされます。

- Windows Server 2008 R2
- Windows Server 2012 R2

Windows で使用できるパスワード同期サービスのバージョンは 1.1.5 です。このバージョンは Red Hat Network の Red Hat Directory Server のダウンロードで利用できます。

7.2. Active Directory および Identity Management について

IdM ドメイン内では、情報はデータマスター (サーバーとレプリカ) 間で信頼性と予測性のある方法でコピーされ、複数のサーバーとレプリカ間で共有されます。このプロセスを **レプリケーション** といいます。

同様のプロセスは、IdM ドメインと Microsoft Active Directory ドメイン間でデータを共有するために使用できます。これが **同期** です。

同期とは Active Directory と Identity Management 間でユーザーデータのコピーを双方向に行うプロセスです。

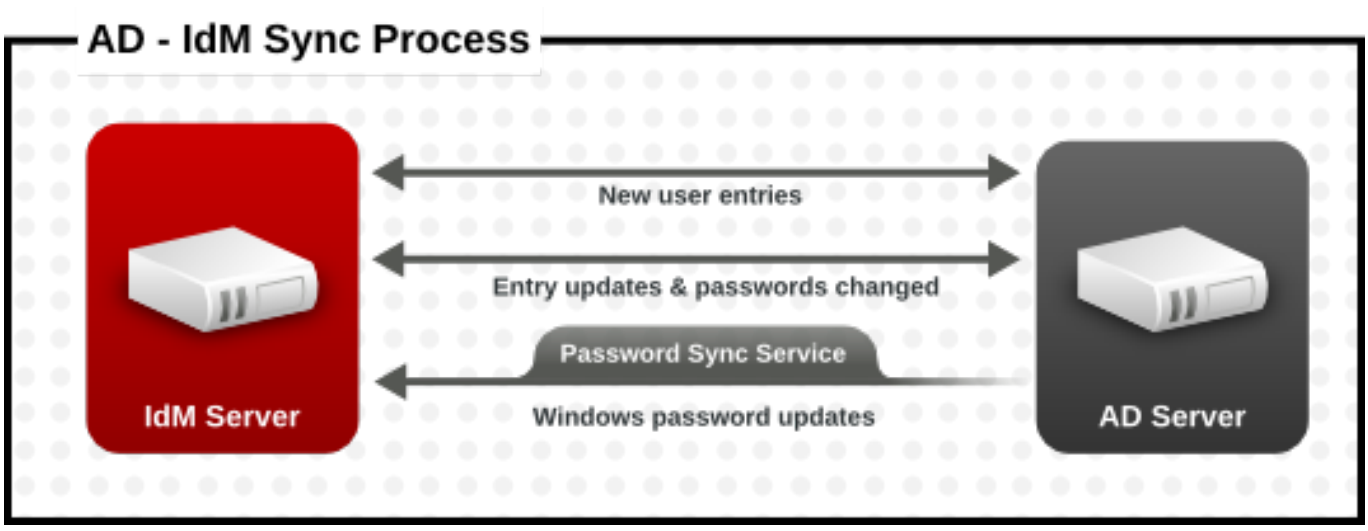


図7.1 Active Directory および IdM の同期

同期は IdM サーバーと Active Directory ドメインコントローラー間の **契約** で定義されます。同期契約は、アカウント属性が処理される方法と同様に、同期可能なユーザーエントリー (同期するサブツリーおよびユーザーエントリー内の必須のオブジェクトクラスなど) を識別するために必要なすべての情報を定義します。同期契約は、特定ドメインのニーズに合わせて調整可能なデフォルト値で作成されます。2 つのサーバーが同期を行う場合、それらは **ピア** と呼ばれます。

表7.1 同期契約の情報

Windows 情報	IdM 情報
ユーザーサブツリー (**cn=Users,$SUFFIX**) 接続情報 Active Directory 管理者ユーザー名およびパスワード パスワード同期サービスのパスワード CA 証明書	ユーザーサブツリー (**ou=People,$SUFFIX**)

同期はほとんどの場合 **双方向** で行われます。情報は、IdM ドメインと Windows ドメイン間で、IdM サーバーとレプリカが情報を共有する方法によく似たプロセスで共有されます。同期は、1 方向のみで行われるように設定することもできます。これは **一方向の** 同期と呼ばれます。

データ競合のリスクを避けるには、1 つのディレクトリーのみからユーザーエントリーを発生させるか、または削除する必要があります。このディレクトリーは通常、IT 環境の主要な ID ストアである Windows ディレクトリーであり、新規のアカウントまたはアカウント削除は Identity Management ピアに対して同期します。いずれのディレクトリーもエントリーを変更できます。

次に同期は 1 つの Identity Management サーバーと 1 つの Active Directory ドメインコントローラー間で設定されます。Identity Management サーバーはスループットを IdM ドメイン全体に伝播し、ドメインコントローラーは変更を Windows ドメイン全体に伝播します。

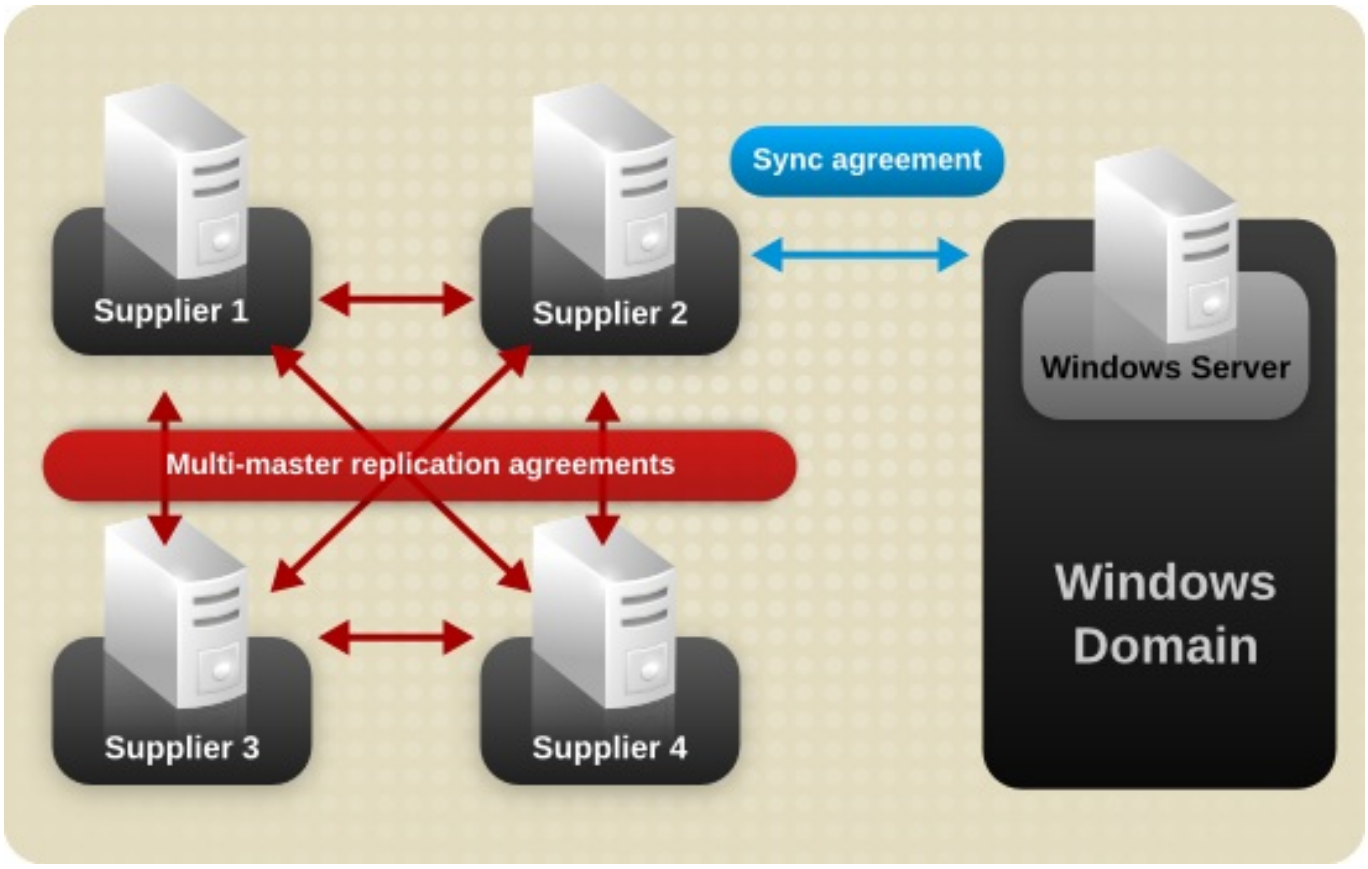


図7.2 同期トポロジー

IdM 同期には、以下のようないくつかの主要な機能があります。

- 同期操作は 5 分ごとに実行されます。
- 同期は 1 つの Active Directory ドメインでのみ設定できます。
- 同期は **1 つの** Active Directory ドメインコントローラーでのみ設定できます。
- ユーザー情報のみが同期されます。
- ユーザー属性とパスワードの両方を同期することができます。
- 変更は双方向 (Active Directory から IdM および IdM から Active Directory の両方向) で行われますが、アカウントの作成は、Active Directory から Identity Management へと一方向でのみ行われます。Active Directory で作成される新規アカウントは IdM に対して自動的に同期されます。ただし IdM で作成されるユーザーアカウントも、同期前に Active Directory で作成されている必要があります。
- アカウントロック情報はデフォルトで同期され、1 つのドメインで無効にされているユーザーアカウントは他方のドメインでも無効にされます。
- パスワードの変更は即時に有効になります。ユーザーパスワードが 1 つのピアで追加または変更される場合、その変更は他のピアサーバーに即時に伝播します。

  パスワード同期クライアントは新規パスワードまたはパスワード更新を同期します。

  IdM と Active Directory の両方でハッシュ化された形式で保存されている既存のパスワードについては、パスワード同期クライアントがインストールされている場合も暗号化を解除したり、同期したりすることができないため、既存のパスワードは同期されません。ピアサーバー間の同期を開始するにはユーザーパスワードを変更する必要があります。

Active Directory ユーザーが IdM に対して同期される場合、特定の属性 (Kerberos および POSIX 属性を含む) では IPA 属性がユーザーエントリーの自動的に追加されます。これらの属性は、IdM ドメイン内で IdM によって使用されます。それらは対応する Active Directory ユーザーエントリーに対して同期し直されることはありません。

同期対象の一部のデータの変更は、同期プロセスの一環として実行できます。たとえば、特定の属性は、それらが IdM ドメインに対して同期される場合に Active Directory ユーザーアカウントに同時に追加できます。それらの属性の変更は同期契約の一部として定義されます。これについては、「ユーザーアカウント属性を同期する動作の変更」 で説明されています。

7.3. 同期された属性について

Identity Management は IdM と Active Directory ユーザーエントリー間のユーザー属性のサブセットを同期します。Identity Management または Active Directory のいずれかにあるエントリーのその他の属性は同期時に無視されます。

注記

ほとんどの POSIX 属性は同期されません。

Active Directory LDAP スキーマと Identity Management で使用される 389 Directory Server LDAP スキーマ間には大きなスキーマの相違点がありますが、同じ属性も多数あります。これらの属性は、Active Directory と IdM ユーザーエントリー間で単純に同期され、属性名や値の形式には変更が加えられません。

Identity Management および Windows サーバーで同一のユーザースキーマ

- cn [6]
- physicalDeliveryOfficeName
- 説明
- postOfficeBox
- destinationIndicator
- postalAddress
- facsimileTelephoneNumber
- postalCode
- givenname
- registeredAddress
- homePhone
- sn
- homePostalAddress
- st
- initials
- street
- l
- telephoneNumber
- mail
- teletexTerminalIdentifier
- mobile
- telexNumber
- o
- title
- ou
- userCertificate
- pager
- x121Address

一部の属性には異なる名前が使用されていますが、IdM (389 Directory Server を使用) と Active Directory 間には直接的な対応関係があります。それらの属性は同期プロセスで **マップされます**。

表7.2 Identity Management と Active Directory 間でマップされるユーザースキーマ

Identity Management	Active Directory
cn [a]	name
nsAccountLock	userAccountControl
ntUserDomainId	sAMAccountName
ntUserHomeDir	homeDirectory
ntUserScriptPath	scriptPath
ntUserLastLogon	lastLogon
ntUserLastLogoff	lastLogoff
ntUserAcctExpires	accountExpires
ntUserCodePage	codePage
ntUserLogonHours	logonHours
ntUserMaxStorage	maxStorage
ntUserProfile	profilePath
ntUserParms	userParameters
ntUserWorkstations	userWorkstations

[a] ***cn*** は Identity Management から Active Directory に同期される際に直接マップされます (***cn*** から ***cn***)。Active Directory から同期される際に、***cn*** は Active Directory の ***name*** 属性から Identity Management の ***cn*** 属性にマップされます。

7.3.1. Identity Management と Active Directory 間のユーザースキーマの相違点

属性は Active Directory と IdM 間で正常に同期される場合でも、Active Directory および Identity Management が基礎となる X.500 オブジェクトクラスを定義する方法には依然として違いがあります。この定義方法の違いは、複数の LDAP サービスでのデータの処理方法の違いにつながる可能性があります。

このセクションでは、Active Directory および Identity Management がそれら 2 つのドメイン間で同期できる一部の属性を処理する方法の相違点について説明します。

7.3.1.1. cn 属性の値

389 Directory Server では、***cn*** 属性に複数の値を設定できますが、Active Directory ではこの属性には単一の値のみを設定できます。Identity Management の ***cn*** 属性が同期されると、単一の値のみが Active Directory ピアに送信されます。

これを同期との関連で見ると、***cn*** 値が Active Directory エントリーに追加され、その値が Identity Management の ***cn*** の値のいずれでもない場合、Identity Management のすべての***cn*** 値は単一の Active Directory 値で上書きされます。

もう 1 つの重要な相違点として、Active Directory では ***cn*** 属性をその命名属性として使用するのに対し、Identity Management は ***uid*** を使用する点があります。これは、***cn*** 属性が Identity Management で編集される場合、エントリーの名前を完全に (および間違って) 変更してしまう可能性があることを意味しています。その ***cn*** の変更が Active Directory エントリーに対して書き込まれる場合、エントリーの名前が変更され、新たに名前の付けられたエントリーは Identity Management に書き戻されます。

7.3.1.2. street および streetAddress の値

Active Directory はユーザーのユーザーの住所に ***streetAddress*** 属性を使用します。これは 389 Directory Server が ***street*** 属性を使用する方法に相当します。Active Directory および Identity Management が ***streetAddress*** および ***street*** 属性を使用する方法には 2 つの重要な相違点があります。

- 389 Directory Server では、***streetAddress*** は ***street*** の別名です。Active Directory には ***street*** 属性もありますが、これは***streetAddress*** の別名ではなく、独立した値を保持できる別個の属性です。
- Active Directory は ***streetAddress*** と ***street*** の両方を単一値の属性として定義しますが、389 Directory Server は RFC 4519 で指定されるように ***street*** を複数値の属性として定義します。

389 Directory Server および Active Directory が ***streetAddress*** および ***street*** 属性を処理する方法が異なるため、Active Directory および Identity Management で address 属性を設定する際には以下のような 2 つのルールに従う必要があります。

- 同期プロセスでは、Active Directory エントリーの ***streetAddress*** を Identity Management の ***street*** にマップします。競合を避けるために、***street*** 属性を Active Directory で使用しないでください。
- 単一の Identity Management ***street*** 属性値のみが Active Directory に同期されます。***streetAddress*** 属性が Active Directory で変更され、新規の値が Identity Management に存在しない場合には、Identity Management のすべての ***street*** 属性値は、新規の単一の Active Directory 値に置き換わります。

7.3.1.3. initials 属性についての制約

initials 属性の場合、Active Directory は最大の長さとして 6 文字の制限を課しますが、389 Directory Server には長さ制限がありません。7 文字以上の ***initials*** 属性が Identity Management に追加される場合、値は Active Directory エントリーと同期される際にトリミングされます。

7.3.1.4. surname (sn) 属性の要求

Active Directory は surname 属性なしに **person** エントリーを作成することを許可します。ただし RFC 4519 は **person** オブジェクトクラスを surname 属性を必要するものとして定義し、これは Directory Server で使用される定義になります。

Active Directory **person** エントリーが surname 属性なしで作成される場合、そのエントリーはオブジェクトクラスの違反で失敗するため、IdM に対して同期されません。

7.3.2. Active Directory エントリーおよび POSIX 属性

Windows は固有なランダムの **セキュリティー ID (SID)** を使用してユーザーを識別します。これらの SID はブロックまたは範囲を指定して割り当てられ、Windows ドメイン内の異なるシステムユーザータイプを識別します。ユーザーが Identity Management と Active Directory 間で同期される場合、ユーザーの Windows SID は Identity Management エントリーによって使用される Unix UID にマップされます。つまり Windows SID は、対応する Unix エントリーの ID として使用される Windows エントリー内の唯一の ID であり、これがマッピングに使用されます。

Active Directory ドメインが Unix スタイルのアプリケーションまたはドメインと対話する際に、Active Directory ドメインは Unix で Unix スタイルの ***uidNumber*** および ***gidNumber*** 属性を有効にするために Unix または Unix 用 IdM のサービスを使用することができます。これにより、Windows ユーザーエントリーは RFC 2307 の属性の仕様を満たすことができます。

ただし ***uidNumber*** および ***gidNumber*** 属性は、Identity Management エントリーの***uidNumber*** および ***gidNumber*** 属性としては実際に使用されません。Identity Management の***uidNumber*** および ***gidNumber*** 属性は、Windows ユーザーが同期される場合に生成されます。

注記

Identity Management で定義され、使用される ***uidNumber*** および ***gidNumber*** 属性は Active Directory エントリーで定義され、使用されるものと同じ ***uidNumber*** および ***gidNumber*** 属性ではなく、これらの番号に関連性はありません。

7.4. 同期用の Active Directory のセットアップ

ユーザーアカウントのみの同期は IdM 内で有効にされるので、同期契約をセットアップ (「同期契約の作成」) するタスクのみが必要になります。ただし Active Directory は、Identity Management サーバーから接続できるように設定する必要があります。

7.4.1. 同期用の Active Directory ユーザーの作成

Windows サーバーでは、IdM サーバーが Active Directory ドメインに接続するために使用するユーザーを作成する必要があります。

Active Directory でユーザーを作成するプロセスは Windows サーバーの文書 (http://technet.microsoft.com/en-us/library/cc732336.aspx) で説明されています。新規のユーザーアカウントには適切な権限を設定する必要があります。

- 同期ユーザーアカウントに、同期される Active Directory サブツリーに対する **ディレクトリー変更の複製** 権限を付与します。レプリケーター権限は同期ユーザーが同期操作を実行するために必要です。

  レプリケーター権限については、http://support.microsoft.com/kb/303972 に説明されています。

- 同期ユーザーを **Account Operator** および **Enterprise Read-Only Domain controller** グループのメンバーとして追加します。このユーザーを完全な **Domain Admin** グループに属させる必要はありません。

7.4.2. Active Directory 証明機関のセットアップ

Identity Management サーバーは、セキュアな接続を使用して Active Directory サーバーに接続します。この接続には Active Directory サーバーで利用可能な CA 証明書または CA 証明書チェーンがあることが条件となり、Windows サーバーを信頼されるピアにするためにこれらの証明書を Identity Management セキュリティーデータベースにインポートすることができます。

これは技術的には (Active Directory に対して) 外部の CA を使って実行できますが、ほとんどのデプロイては Active Directory で利用可能な証明書サービスを使用する必要があります。

Active Directory で証明書サービスをセットアップし、これを設定する手順は、Microsoft 文書 (http://technet.microsoft.com/en-us/library/cc772393(v=WS.10).aspx) で説明されています。

7.5. 同期契約の管理

7.5.1. Active Directory および IdM CA 証明書の信頼

Active Directory と Identity Management はどちらもサーバー認証に証明書を使用します。Active Directory および IdM SSL サーバー証明書が相互に信頼されるようにするには、両方のサーバーがそれらの証明書を発行する CA の CA 証明書を信頼する必要があります。つまり、Active Directory CA 証明書は IdM データベースにインポートされ、IdM CA 証明書は Active Directory データベースにインポートされる必要があることを意味します。

1. Active Directory サーバーで、IdM サーバーの CA 証明書を **`http://ipa.example.com/ipa/config/ca.crt`** からダウンロードします。

2. IdM CA 証明書を Active Directory 証明書データベースにインストールします。これは Microsoft 管理コンソールまたは certutil ユーティリティー を使用して実行できます。

   実行可能プログラムを右クリックし、**`Run as administrator`** を選択してから **`certutil`** を **`-installcert`** オプションを指定して実行します。以下は例になります。

   ```
   C:\Windows\system32\certutil -installcert -v -config
   "ipaserver.example.com\Example Domain CA" c:\path\to\ca.crt
   ```

   このコマンドは管理アカウントで実行する必要があります。管理アカウントでないと、証明書データベースにアクセスできずに失敗します。

   証明書のインストールについての詳細は、Active Directory 文書を参照してください。

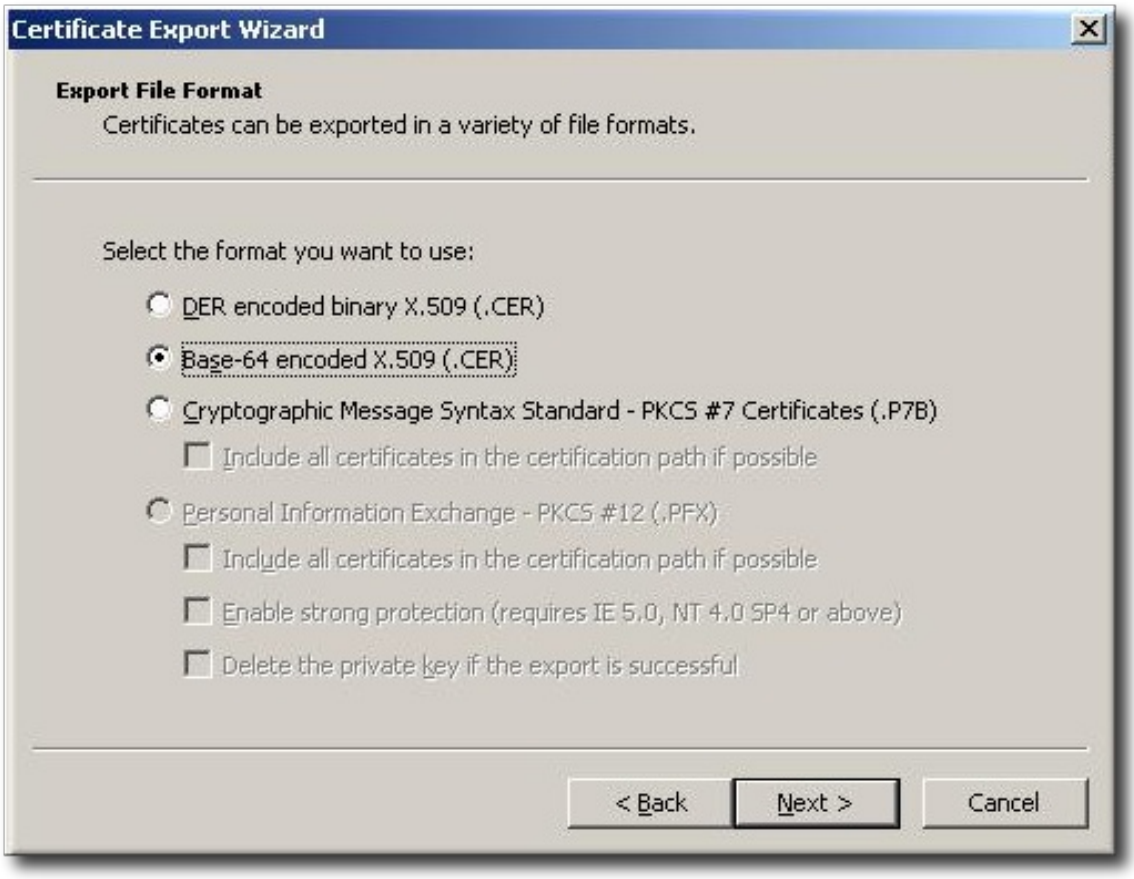
3. Active Directory CA 証明書をエクスポートします。

   a. **`My Network Places`** で、CA 配布ポイントを開きます。

   b. セキュリティー証明書ファイル (**`.crt`** ファイル) をダブルクリックして **`Certificate`** ダイアログボックスを表示します。

   c. **`Details`** タブで、**`Copy to File`** をクリックし、**Certificate Export Wizard** を開始します。

   d. **`Next`** をクリックしてから、**`Base-64 encoded X.509 (.CER)`** を選択します。



   e. エクスポートされたファイルに適切なディレクトリーおよびファイル名を指定します。**`Next`** をクリックして証明書をエクスポートしてから**`Finish`** をクリックします。

4. Active Directory 証明書を IdM サーバーマシンにコピーします。
5. IdM サーバーの CA 証明書を **http://ipa.example.com/ipa/config/ca.crt** からダウンロードします。
6. Active Directory CA 証明書および IdM CA 証明書の両方を **/etc/openldap/cacerts/** ディレクトリーにコピーします。
7. 証明書の hash symlink を更新します。

```
cacertdir_rehash /etc/openldap/cacerts/
```

8. **/etc/openldap/ldap.conf** ファイルを編集し、**/etc/openldap/cacerts/** ディレクトリーの証明書を参照し、これを使用するための情報を追加します。

```
TLS_CACERTDIR /etc/openldap/cacerts/
TLS_REQCERT allow
```

7.5.2. 同期契約の作成

同期契約は Active Directory ドメインへの **接続** を作成するので、**ipa-replica-manage connect** コマンドを使用して IdM サーバー上に作成されます。同期契約を作成するためのオプションは 表7.3「同期契約のオプション」 に一覧表示されています。

1. 「Active Directory および IdM CA 証明書の信頼」にあるように Active Directory および IdM サーバーが相互の CA 証明書を信頼していることを確認します。
2. IdM サーバー上の既存の Kerberos 資格情報を削除します。

```
$ kdestroy
```

3. **ipa-replica-manage** コマンドを使用して Windows 同期契約を作成します。これには**--winsync** オプションが必要です。パスワードがユーザーアカウントと同様に同期される場合、**--passsync** オプションも使用して、パスワードの同期に使用するパスワードを設定します。

   --binddn および**--bindpwd** オプションは、IdM が Active Directory サーバーに接続するために使用するシステムアカウントのユーザー名とパスワードを Active Directory サーバー上で指定します。

```
$ ipa-replica-manage connect --winsync
 --binddn cn=administrator,cn=users,dc=example,dc=com
 --bindpw Windows-secret
 --passsync secretpwd
 --cacert /etc/openldap/cacerts/windows.cer
 adserver.example.com -v
```

4. プロンプトが出されたら、Directory Manager のパスワードを入力します。
5. **オプション:** 「パスワード同期のセットアップ」に説明されているようにパスワードの同期を設定します。パスワード同期クライアントがないと、ユーザー属性はピアサーバー間で同期されますが、パスワードは同期されません。

注記

パスワード同期クライアントはパスワードの変更を取り込み、 Active Directory と IdM 間でこれらの変更を同期します。つまり、そのクライアントは新規パスワードまたはパスワード更新を同期します。

IdM と Active Directory の両方でハッシュ化された形式で保存されている既存のパスワードについては、パスワード同期クライアントがインストールされている場合も暗号化を解除したり、同期したりすることができないため、既存のパスワードは同期されません。ピアサーバー間の同期を開始するにはユーザーパスワードを変更する必要があります。

表7.3 同期契約のオプション

オプション	説明
--winsync	同期契約として識別します。
--binddn	同期 ID の完全ユーザー DN を指定します。これは、IdM LDAP サーバーが Active Directory にバインドするために使用するユーザー DN です。このユーザーは Active Directory ドメインに存在し、Active Directory サブツリー上にレプリケーター、読み取り、検索および書き込み権限が設定されている必要があります。
--bindpw	同期ユーザーのパスワードを指定します。
--passsync	同期に関係する Windows ユーザーアカウントのパスワードを指定します。
--cacert	Active Directory CA 証明書の完全パスおよびファイル名を指定します。この証明書は「Active Directory および IdM CA 証明書の信頼」にあるようにエクスポートされます。
--win-subtree	同期するユーザーが含まれる Windows サブツリーの DN を指定します。デフォルト値は **cn=Users,$SUFFIX** です。
AD_server_name	Active Directory ドメインコントローラーのホスト名を指定します。

7.5.3. ユーザーアカウント属性を同期する動作の変更

同期契約が作成されると、同期中の同期プロセスで、ユーザーアカウント属性を処理する方法についての特定のデフォルト動作が定義されます。動作のタイプには、ロックアウト属性の処理方法や異なる DN 形式の処理方法などが含まれます。この動作は、同期契約を編集して変更できます。属性に関連したパラメーターの一覧は 表7.4「同期される属性設定」 にあります。

同期契約は LDAP サーバーの特殊なプラグインエントリーとして存在し、それぞれの属性動作は LDAP 属性から設定されます。同期の動作を変更するには、**ldapmodify** コマンドを使用して LDAP サーバーエントリーを直接変更します。

たとえば、アカウントロックアウト属性はデフォルトでは IdM と Active Directory 間で同期されますが、これは、***ipaWinSyncAcctDisable*** 属性を編集して無効にできます。(この変更により、アカウントは Active Directory で無効にされている場合にも IdM ではアクティブな状態になり、その逆の場合も同じになります。)

```
[jsmith@ipaserver ~]$ ldapmodify -x -D "cn=directory manager" -w password
```

```
dn: cn=ipa-winsync,cn=plugins,cn=config
changetype: modify
replace: ipaWinSyncAcctDisable
ipaWinSyncAcctDisable: none

modifying entry "cn=ipa-winsync,cn=plugins,cn=config"
```

表7.4 同期される属性設定

パラメーター	説明	使用可能な値
一般的なユーザーアカウントパラメーター		
ipaWinSyncNewEntryFilter	新規ユーザーエントリーに追加するオブジェクトクラスの一覧が含まれるエントリーの検索に使用する検索フィルターを設定します。	デフォルト: **(cn=ipaConfig)**
ipaWinSyncNewUserOCAttr	新規ユーザーエントリーに追加するオブジェクトクラスの一覧が実際に含まれる設定エントリーの属性を設定します。	デフォルト: **ipauserobjectclasses**
ipaWinSyncHomeDirAttr	POSIX ホームディレクトリーのデフォルトの場所を含むエントリー内の属性を識別します。	デフォルト: **ipaHomesRootDir**
ipaWinSyncUserAttr	Active Directory ユーザーが Active Directory ドメインから同期される場合に Active Directory ユーザーに追加する特定の値で追加の属性を設定します。この属性が複数値の属性の場合は、これを複数回設定でき、同期プロセスは値のすべてをエントリーに追加します。 **注記** これにより、エントリーに属性が存在しない場合に属性値のみが設定されます。属性が存在する場合はエントリーの値は Active Directory エントリーの同期時に使用されます。	ipaWinSyncUserAttr: **attributeName attributeValue**

パラメーター	説明	使用可能な値
ipaWinSyncForceSync	既存の Active Directory ユーザーに一致する既存の IdM ユーザーを同期対象とするかどうかを設定します。true に設定すると、この IdM ユーザーは同期されるように自動的に編集されます。IdM ユーザーに、既存の Active Directory ユーザーの ***sAMAccountName*** と同一の ***uid*** パラメーターがある場合、そのアカウントはデフォルトでは同期 **されません**。この属性は、同期サービスに対して、***ntUser*** および ***ntUserDomainId*** を IdM ユーザーエントリーに自動的に追加し、それらが同期されるように指示します。	true | false
ユーザーアカウントロックパラメーター		
ipaWinSyncAcctDisable	アカウントロックアウト属性を同期する方法を設定します。有効にするアカウントロックアウト設定を制御することができます。たとえば **to_ad** は、アカウントロックアウト属性が IdM に設定される場合に、その値が Active Directory に対して同期され、ローカルの Active Directory 値を上書きすることを意味します。デフォルトでは、アカウントロックアウト属性は両方のドメインから同期されます。	- both (デフォルト) - to_ad - to_ds - none
ipaWinSyncInactivatedFilter	非アクティブ化された (無効にされた) ユーザーを保持するために使用されるグループの DN の検索に使用する検索フィルターを設定します。これは、ほとんどの実装では変更される必要はありません。	デフォルト: **(&(cn=inactivated)(objectclass=groupOfNames))**
ipaWinSyncActivatedFilter	アクティブユーザーを保持するために使用するグループの DN の検索に使用する検索フィルターを設定します。これは、ほとんどの実装では変更される必要はありません。	デフォルト: **(&(cn=activated)(objectclass=groupOfNames))**
グループパラメーター		
ipaWinSyncDefaultGroupAttr	ユーザーのデフォルトグループを確認するために参照する新規ユーザーアカウントの属性を設定します。その後、エントリーのグループ名がユーザーアカウントの ***gidNumber*** の検索に使用されます。	デフォルト: **ipaDefaultPrimaryGroup**

パラメーター	説明	使用可能な値
ipaWinSyncDefaultGroupFilter	グループ名を POSIX ***gidNumber*** にマップするために検索フィルターを設定します。	デフォルト: **(&(gidNumber=*)(objectclass=posixGroup)(cn=groupAttr_value))**
レルムパラメーター		
ipaWinSyncRealmAttr	レルムエントリーにレルム名を含む属性を設定します。	デフォルト: ***cn***
ipaWinSyncRealmFilter	IdM レルム名を含むエントリーの検索に使用する検索フィルターを設定します。	デフォルト: **(objectclass=krbRealmContainer)**

7.5.4. 同期された Windows サブツリーの変更

同期契約を作成すると、同期されたユーザーデータベースとして使用する 2 つのサブツリーが自動的に設定されます。IdM の場合、デフォルトは **cn=users,cn=accounts,$SUFFIX** となり、Active Directory の場合、デフォルトは **CN=Users,$SUFFIX** となります。

Active Directory サブツリーの値は、**--win-subtree** オプションを使用して同期契約が作成される場合はデフォルト以外の値に設定できます。この契約の作成後に、**ldapmodify** コマンドを使用し、同期契約エントリー内の ***nsds7WindowsReplicaSubtree*** 値を編集して Active Directory サブツリーを変更できます。

1. **ldapsearch** を使用して同期契約の名前を取得します。この検索により、エントリー全体ではなく、***dn*** および ***nsds7WindowsReplicaSubtree*** 属性の値のみが返されます。

```
[jsmith@ipaserver ~]$ ldapsearch -xLLL -D "cn=directory manager" -w
password -p 389 -h ipaserver.example.com -b cn=config
objectclass=nsdswindowsreplicationagreement dn
nsds7WindowsReplicaSubtree

dn:
cn=meToWindowsBox.example.com,cn=replica,cn=dc\3Dexample\2Cdc\3Dco
m,cn=mapping tree,cn=config
nsds7WindowsReplicaSubtree: cn=users,dc=example,dc=com

... 8< ...
```

2. 同期契約を変更します。

```
[jsmith@ipaserver ~]$ ldapmodify -x -D "cn=directory manager" -W -p
389 -h ipaserver.example.com <<EOF
 dn:
cn=meToWindowsBox.example.com,cn=replica,cn=dc\3Dexample\2Cdc\3Dco
m,cn=mapping tree,cn=config
 changetype: modify
 replace: nsds7WindowsReplicaSubtree
 nsds7WindowsReplicaSubtree: cn=alternateusers,dc=example,dc=com
 EOF

 modifying entry
"cn=meToWindowsBox.example.com,cn=replica,cn=dc\3Dexample\2Cdc\3Dc
om,cn=mapping tree,cn=config"
```

新規のサブツリー設定は即時に有効になります。同期操作が実行中の場合は、現在の操作が完了するとすぐに有効になります。

7.5.5. 一方向同期の設定

デフォルトでは、すべての変更および削除は双方向に行われます。Active Directory の変更は Identity Management に対して行われ、Identity Management のエントリーの変更は Active Directory に対して同期されます。これは本質的に平等な複数マスター関係であり、Active Directory と Identity Management はどちらも同期におけるピアであり、どちらもデータマスターになります。

ただし一部のデータ構造または IT 設計では、一方のドメインのみをデータマスターとし、他方のドメインでは更新を受け入れられるようにする必要があります。この場合、複数マスターの関係 (ピアサーバーが平等) からマスター対コンシュマーの関係に同期関係が変更されます。

これは、同期契約に ***oneWaySync*** パラメーターを設定することによって実行されます。使用できる値は ***fromWindows*** (Active Directory から Identity Management への同期) および ***toWindows*** (Identity Management から Active Directory への同期) です。

たとえば、Active Directory から Identity Management に変更を同期するには、以下を実行します。

```
[jsmith@ipaserver ~]$ ldapmodify -x -D "cn=directory manager" -w password
-p 389 -h ipaserver.example.com

dn:
cn=windows.example.com,cn=replica,cn=dc\3Dexample\2Cdc\3Dcom,cn=mapping
tree,cn=config
changetype: modify
add: oneWaySync
oneWaySync: fromWindows
```

重要

一方向の同期を有効にしても、一方向で同期されたサーバー上の変更を自動的に**回避する訳ではないため**、これにより同期更新における同期ピア間の不整合が生じる可能性があります。たとえば一方向の同期は、Active Directory から Identity Management の方向に設定されるため、Active Directory は (基本的には) データマスターになります。エントリーが Identity Management で変更されるか、または削除される場合、異なる情報が生じますが、その情報および変更は Active Directory に移行されることはありません。次の同期更新時に、編集内容は Directory Server で上書きされ、エントリーを削除していても再び追加されます。

7.5.6. 同期契約の削除

同期は、IdM および Active Directory サーバーの**接続を解除する**ために同期契約を削除することによって停止することができます。同期契約を作成する場合とは逆に、同期契約の削除では **`ipa-replica-manage disconnect`** コマンドおよび Active Directory サーバーのホスト名が使用されます。

1. 同期契約を削除します。

```
# ipa-replica-manage disconnect adserver.example.com
```

2. IdM サーバーデータベースから Active Directory CA 証明書を削除します。

```
# certutil -D -d /etc/dirsrv/slapd-EXAMPLE.COM/ -n "Imported CA"
```

7.5.7. Winsync 契約のエラー

Active Directory サーバーに接続できないので、同期契約の作成に失敗する。

最も一般的な同期契約のエラーの 1 つは、IdM サーバーが Active Directory サーバーに接続できない場合に生じます。

```
"Update failed! Status: [81  - LDAP error: Can't contact LDAP server]
```

これは、契約の作成時に正しくない Active Directory CA 証明書が指定される場合に生じる可能性があります。これにより、IdM LDAP データベース (**/etc/dirsrv/slapd-DOMAIN/** ディレクトリー内) に **Imported CA** という名前で重複した証明書が作成されます。これは、**certutil** を使用して確認できます。

```
$ certutil -L -d /etc/dirsrv/slapd-DOMAIN/

Certificate Nickname                                         Trust
Attributes
SSL,S/MIME,JAR/XPI

CA certificate                                               CTu,u,Cu
Imported CA                                                  CT,,C
Server-Cert                                                  u,u,u
Imported CA                                                  CT,,C
```

この問題を解決するには、証明書データベースをクリアします。

```
# certutil -d /etc/dirsrv/slapd-DOMAIN-NAME -D -n "Imported CA"
```

これにより、LDAP データベースから CA 証明書が削除されます。

エントリーが存在するためパスワードが同期されないというエラーが出される。

ユーザーデータベースの一部のエントリーについて、エントリーがすでに存在するためにパスワードはリセットされないという情報のエラーメッセージが表示される可能性があります。

```
"Windows PassSync entry exists, not resetting password"
```

これはエラーではありません。このメッセージは、適用除外ユーザー、パスワード同期ユーザーが変更されていない場合に生じます。パスワード同期ユーザーは、IdM でパスワードを変更するためにサービスで使用される操作上のユーザーです。

7.6. パスワード同期の管理

ユーザーエントリーの同期は、同期契約で設定されます。ただし、Active Directory と Identity Management の両方にあるパスワードは通常のユーザー同期プロセスの一部として組み込まれてはいません。ユーザーアカウントの作成またはパスワードの変更時にパスワードを取り込み、同期更新でそのパスワード情報を転送できるようにするには、別個のクライアントが Active Directory サーバー上にインストールされる必要があります。

注記

パスワード同期クライアントはパスワードの変更を取り込み、 Active Directory と IdM 間でこれらの変更を同期します。つまり、そのクライアントは新規パスワードまたはパスワード更新を同期します。

IdM と Active Directory の両方でハッシュ化された形式で保存されている既存のパスワードについては、パスワード同期クライアントがインストールされている場合も暗号化を解除したり、同期したりすることができないため、既存のパスワードは同期されません。ピアサーバー間の同期を開始するにはユーザーパスワードを変更する必要があります。

7.6.1. パスワード同期のための Windows Server のセットアップ

パスワードの同期には、以下の 2 点が必要になります。

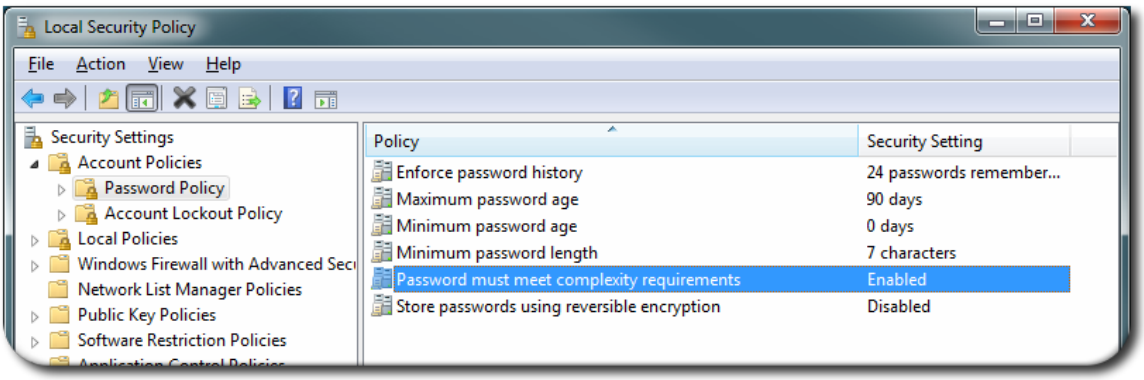
- Active Directory は SSL で実行されている必要がある。
- パスワード同期サービスは **各** Active Directory ドメインコントローラーにインストールされている必要がある。

パスワード同期サービスはパスワードの変更を記録し、それらをセキュアな接続により IdM エントリーに対して同期します。

注記

Enterprise Root Mode で Microsoft 証明書システムをインストールします。次に Active Directory はその SSL サーバー証明書を取得するために自動的に登録されます。

1. Active Directory のパスワードの複雑性に関するポリシーは有効にされ、パスワード同期サービスが実行されます。
    a. コマンドラインから **`secpol.msc`** を実行します。
    b. **Security Settings** を選択します。
    c. **Account Policies** を開いてから、**Password Policy** を開きます。
    d. **`Password must meet complexity requirements`** オプションを有効にし、保存します。



2. SSL がすでに有効にされていない場合、Active Directory サーバーに SSL をセットアップします。LDAPS のセットアップの詳細は、Microsoft ナレッジベース (http://support.microsoft.com/kb/321051) で説明されています。

    a. 証明機関を、**Add/Remove Programs** の **Windows Components** セクションにインストールします。

    b. **Enterprise Root CA** オプションを選択します。

    c. Active Directory サーバーを再起動します。IIS web サービスが実行中である場合、CA 証明書は **http://servername/certsrv** を開いてアクセスできます。

    d. Active Directory サーバーをセットアップして SSL サーバー証明書を使用します。

        a. Active Directory の完全修飾ドメイン名を証明書のサブジェクトに使用し、証明書要求 **.inf** を作成します。以下が例になります。

```
;----------------- request.inf -----------------

[Version]

Signature="$Windows NT$

[NewRequest]

Subject = "CN=ad.server.example.com, O=Engineering,
L=Raleigh, S=North Carolina, C=US"
KeySpec = 1
KeyLength = 2048
Exportable = TRUE
MachineKeySet = TRUE
SMIME = False
PrivateKeyArchive = FALSE
UserProtected = FALSE
UseExistingKeySet = FALSE
ProviderName = "Microsoft RSA SChannel Cryptographic
Provider"
ProviderType = 12
RequestType = PKCS10
KeyUsage = 0xa0

[EnhancedKeyUsageExtension]

OID=1.3.6.1.5.5.7.3.1

;-------------------------------------------------
```

        .inf 要求ファイルの詳細は、http://technet.microsoft.com/en-us/library/cc783835.aspx などの Microsoft 文書を参照してください。

        b. 証明書要求を生成します。

```
certreq -new request.inf request.req
```

        c. 要求を Active Directory CA に送信します。以下が例になります。

```
certreq -submit request.req certnew.cer
```

注記

コマンドラインがエラーメッセージを返す場合、Web ブラウザーを使用して CA にアクセスし、証明書要求を送信します。IIS が実行中の場合、CA URL は **http://servername/certsrv** です。

d. 証明書要求を受け入れます。以下が例になります。

```
certreq -accept certnew.cer
```

e. サーバー証明書が Active Directory サーバー上に置かれるようにします。

File メニューで、**Add/Remove** をクリックしてから **Certificates** および **Personal>Certificates** をクリックします。

f. CA 証明書を Directory Server から Active Directory にインポートします。**Trusted Root CA** をクリックしてから **Import** をクリックし、Directory Server CA 証明書を参照します。

e. ドメインコントローラーを再起動します。

7.6.2. パスワード同期のセットアップ

Windows パスワードを同期するために、Active Directory ドメインのすべてのドメインコントローラーにパスワード同期サービスをインストールします。

1. **PassSync.msi** ファイルを Active Directory マシンにダウンロードします。

a. カスタマーポータルにログインします。

b. **ダウンロード** タブをクリックします。

c. ページの中央にある **Red Hat Enterprise Linux** のダウンロードをクリックします。

d. **Directory Server** などの検索キーワードを使用してダウンロードをフィルターし、Red Hat Enterprise Linux バージョンのいずれかを拡張します。

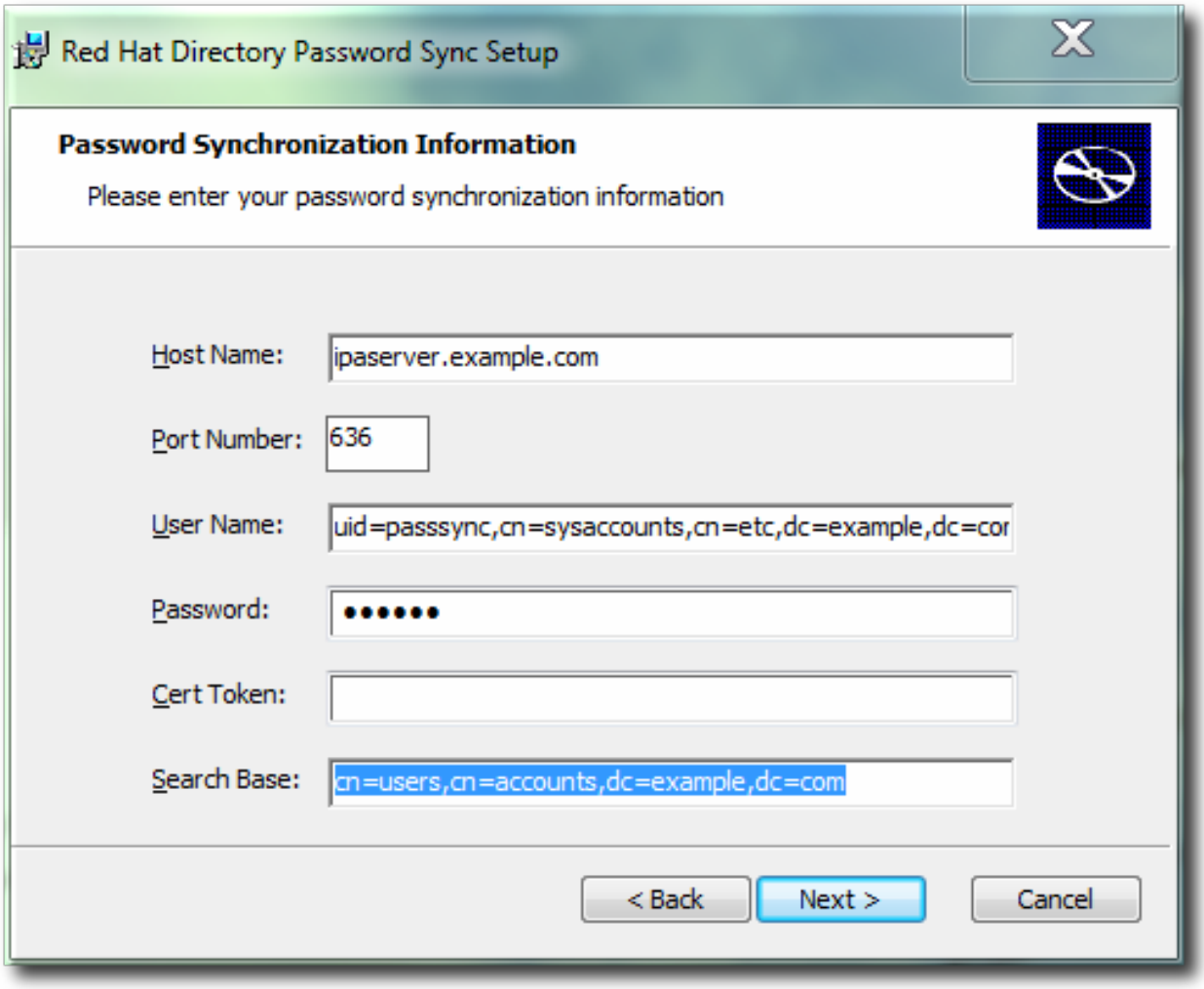
e. Directory Server リンクをクリックします。

f. Directory Server ページで、WinSync Installer の適切なバージョンをダウンロードします。これは Password Sync MSI ファイル (**RedHat-PassSync-1.1.5-*arch*.msi**) です。

注記

Red Hat Enterprise Linux アーキテクチャーの種類を問わず、利用できる 2 つの PassSync パッケージがあります。1 つは 32-bit Windows サーバー用で、もう 1 つは 64-bit 用です。お使いの Windodws プラットフォームに適したパッケージを選択するようにしてください。

2. Password Sync MSI ファイルをダブルクリックして、これをインストールします。
3. **Password Sync Setup** 画面が表示されます。**Next** を押して、インストールを開始します。
4. 以下の情報を入力し、IdM サーバーへの接続を設定します。
   - ホスト名およびセキュアなポート番号を含む IdM サーバー接続情報。
   - IdM マシンに接続するために Active Directory が使用するシステムユーザーのユーザー名。このアカウントは、同期が IdM サーバー上に設定される場合に自動的に設定されます。デフォルトのアカウントは **uid=passsync,cn=sysaccounts,cn=etc,dc=example,dc=com** です。
   - 同期契約の作成時に **--passsync** オプションに設定されるパスワード。
   - IdM サーバー上の People サブツリーの検索ベース。Active Directory サーバーは、**ldapsearch** またはレプリケーション操作の場合と同様に IdM サーバーに接続します。そのため、IdM サブツリーのどこでユーザーアカウントを検索できるかを認識している必要があります。ユーザーサブツリーは **cn=users,cn=accounts,dc=example,dc=com** です。
   - 証明書トークンはこの時点では使用されないため、このフィールドは空白にする必要があります。



Next を押してから **Finish** を押し、パスワード同期をインストールします。

5. IdM サーバーの CA 証明書を Active Directory 証明書ストアにインポートします。
   a. IdM サーバーの CA 証明書を **http://ipa.example.com/ipa/config/ca.crt** からダウンロードします。

b. IdM CA 証明書を Active Directory サーバーにコピーします。

c. IdM CA 証明書をパスワード同期データベースにインストールします。以下が例になります。

```
cd "C:\Program Files\Red Hat Directory Password
Synchronization"

certutil.exe -d . -A -n "IPASERVER.EXAMPLE.COM IPA CA" -t
CT,, -a -i ipaca.crt
```

6. Windows マシンを再起動して、パスワード同期を開始します。

注記

Windows マシンは再起動されている必要があります。再起動しないと **`PasswordHook.dll`** は有効にされず、パスワードの同期は機能しません。

7. 既存のアカウントのパスワードを同期する必要がある場合、ユーザーパスワードをリセットします。

注記

パスワード同期クライアントはパスワードの変更を取り込み、 Active Directory と IdM 間でこれらの変更を同期します。つまり、そのクライアントは新規パスワードまたはパスワード更新を同期します。

IdM と Active Directory の両方でハッシュ化された形式で保存されている既存のパスワードについては、パスワード同期クライアントがインストールされている場合も暗号化を解除したり、同期したりすることができないため、既存のパスワードは同期されません。ピアサーバー間の同期を開始するにはユーザーパスワードを変更する必要があります。

パスワード同期アプリケーションのインストール時におけるパスワード同期の初回の試行は、Directory Server と Active Directory 同期ピアー間の SSL 接続により常に失敗します。証明書およびキーデータベースを作成するためのツールは **`.msi`** でインストールされます。

7.6.3. ユーザーが他のユーザーのパスワードを正常に変更することを許可

デフォルトでは、管理者がユーザーパスワードを変更するたびに、ユーザーは次回ログイン時にパスワードをリセットする必要があります。ただしこの動作については、管理者が即時のパスワードリセットを**要さずに** パスワードをリセットできるように変更することができます。

`passSyncManagersDNs` 属性は、パスワード変更操作が許可され、**かつ**パスワードのリセットが要求されない管理者アカウントを一覧表示します。

重要

上記はパスワードの同期において必要となります。この動作がないと、パスワードの同期が行われるたびに IdM サーバーがこれをパスワード変更操作として解釈し、次回のログイン時にパスワード変更を要求することになってしまうためです。

パスワード同期エントリー **cn=ipa_pwd_extop,cn=plugins,cn=config** を編集し、***passSyncManagersDNs*** 属性をユーザーの名前と共に追加します。この属性は複数値の属性になります。以下が例になります。

```
$ ldapmodify -x -D "cn=Directory Manager" -w secret -h ldap.example.com
-p 389

dn: cn=ipa_pwd_extop,cn=plugins,cn=config
changetype: modify
add: passSyncManagersDNs
passSyncManagersDNs: uid=admin,cn=users,cn=accounts,dc=example,dc=com
```

警告

一覧表示された DN を、ユーザーパスワードの設定機能が必要な管理者アカウントにのみ制限するように注意してください。ここに一覧表示されているすべてのユーザーはすべてのユーザーパスワードにアクセスできる強力な権限が付与されます。

---

[6] ***cn*** は他の同期される属性とは異なる方法で処理されます。これは、Identity Management から Active Directory に同期される際に直接マップされます (***cn*** から ***cn***)。ただし、Active Directory から Identity Management に同期される際に、***cn*** は Windows 上の ***name*** 属性から Identity Management の ***cn*** 属性にマップされます。

第8章 ID ビューおよび既存環境の信頼への移行

Red Hat Identity Management の一部である *ID ビュー* メカニズムにより、ユーザーまたはグループの POSIX 属性を指定できます。新規の ID ビューを作成する際に、上書きする必要のあるユーザーまたはグループ属性を定義することができます。次に、新たに定義された属性がユーザーまたはグループに適用されます。これを許可することにより、ID ビューは他のアイデンティティー管理またはシステム統合ソリューションからの移行時に既存の環境を保持するソリューションを提供します。

ipa-adtrust-install コマンドを実行した後に、デフォルト信頼ビューが作成されます。デフォルト信頼ビューは常に Active Directory ユーザーおよびグループに適用されます。これにより、AD 自体による定義方法にかかわらず、AD ユーザーおよびグループの POSIX 属性を定義できます。AD ユーザーまたはグループを上書きするホスト固有の ID ビューを追加する場合、ホスト固有 ID ビューの属性はデフォルトの信頼ビューの上部で適用されます。新規 ID ビューがデフォルト信頼ビューを上書きする間は、デフォルトビュー自体を削除することはできません。特定の ID ビューがクライアントに適用されない場合、デフォルトの信頼ビューが常に適用されます。

注記

ipa-adtrust-install を実行しない場合、ID ビューおよび IdM ユーザーの上書きを管理するために純粋な IdM 環境で ID ビュー機能を使用することができます。

同期ベースの AD 統合が設定されたセットアップで、すべてのユーザーは、ログイン名、UID、GID またはシェルなどの生成された POSIX 属性と共に IdM サーバーにコピーされます。「間接的な統合」 で説明されているように、同期ベースのアプローチは一般的に推奨されず、代わりに信頼ベースのアプローチを使用することが推奨されます。管理者が AD が AD ユーザーに事前に生成した POSIX 属性を変更できるようにすることにより、ID ビュー機能は、信頼ベースの AD 統合に既存の環境を移行するソリューションを提供します。

ID ビューのユースケースには以下が含まれます。

AD ユーザーの POSIX 属性および SSH データの保管

AD ユーザーの POSIX 属性または SSH キーおよび SSH ログイン情報を定義し、AD ユーザーが ID ビューサポートを使用して SSSD を実行中のクライアントに対して認証される際または AD ユーザーが *コンパクトな LDAP ツリー*を使って認証される際に、それらの定義が適用されるようにします。*コンパクトな LDAP ツリー*は、レガシークライアントのユーザーおよびグループデータと共に単純化した LDAP ツリーを提供します。

この機能は、同期ベースのソリューションからの移行や、Linux 管理者が AD ユーザーの POSIX 属性を手動で定義することを希望するにもかかわらず AD ポリシーではそれが許可されない状況などで役立ちます。

同期ベースから信頼ベースの統合への移行

以前に使用した UID または他のツールを指定して ID ビューの上書きを作成することで、同期ベースの環境にあるユーザーの POSIX 属性を設定します。次にユーザーを AD に戻します。

IdM ユーザーの POSIX 属性についてホストごとのグループ上書きを実行

IdM と AD 間の統合に移行中の NIS ベースのインフラストラクチャーでは、元の POSIX データを一部の NIS ドメイン上で変更されない状態にするか、または会社ポリシーにより AD の元の POSIX データが直接設定されないようにすることが必要になる場合がよくあります。このような場面では、ID ビューを使用してアイデンティティー管理サーバーで POSIX データを直接設定することができます。

複数の異なる環境に異なる POSIX 属性または SSH データを設定

対応するホストグループに応じて、開発、テスト、または本番などの異なる実稼働環境用に異なる POSIX 属性または異なるユーザー SSH 公開キーを設定します。

8.1. ユーザー上書きおよびグループ上書き

すべての *ID ビュー* は、指定されたホストに適用される*ユーザー上書き* および *グループ上書き* のコレクションです。上書きにより、以前の内容を上書きする新規ユーザーまたはグループ属性が提供されます。これにより、以前に生成された属性を新しい属性に置き換えることができます。すべての上書きは AD または IdM ユーザーまたはグループに関連します。

注記

IdM 以外の統合システムでは、IdM で使用されるアルゴリズムとは異なるアルゴリズムを使用して UID および GID 属性を生成できます。Id M システムに準拠させるように生成済みの属性を上書きすることによって、別の統合システムのメンバーであったクライアントを Id M に完全に統合できます。

以下のユーザー属性は ID ビューで上書きできます。

- **`uid`**: ユーザーログイン名
- **`uidNumber`**: ユーザー UID 番号
- **`gidNumber`**: ユーザー GID 番号
- **`loginShell`**: ユーザーログインシェル
- **`gecos`**: ユーザー GECOS エントリー
- **`homeDirectory`**: ユーザーホームディレクトリー
- **`ipaSshPubkey`**: ユーザー SSH 公開キー (単数または複数)

以下のグループ属性は ID ビューで上書きできます。

- **`cn`**: グループ名
- **`gidNumber`**: グループ GID 番号

注記

IdM は *ID 範囲* を使用して異なるドメインとの POSIX ID の競合を防ぎます。IdM は他の種類の ID 範囲との重複を許可するので ID ビューの POSIX ID は特別な範囲タイプを使用しません。たとえば同期で使用された AD ユーザーには、IdM ユーザーと同じ ID 範囲の POSIX ID があります。競合が生じても、POSIX ID は IdM 側の ID ビューで手動で管理されるため、競合する ID を変更することによって簡単に競合を解決できます。

8.2. ID ビューの管理

ID ビューは、追加し、変更し、または削除することができます。ID ビューが上書きする必要のある ID 属性、およびこれを適用する必要のあるクライアントホストを定義することができます。

AD ユーザーの場合、デフォルト信頼ビューからの上書きは常に適用されます。ホストに割り当てられる ID ビューがデフォルト信頼ビューにある値か、または一部の属性についての AD の元の値を上書きする場合、これらの上書きされる値はホストに表示されます。ID ビューがデフォルト信頼ビューにある値を上書きしない場合、別の ID ビューに割り当てられるすべてのクライアントはデフォルト信頼ビューの値を表示します。

デフォルト信頼ビューは AD ユーザーの上書きのみを受け入れます。デフォルト信頼ビューには IdM ユーザーまたはグループの上書きを追加することはできません。IdM ユーザーの場合、デフォルトビューは対応する IdM ユーザーレコードで定義される値によって表示されます。

IdM サーバーおよびレプリカは、ID ビューの上書きなしにデフォルト信頼ビューを常に適用します。異なる ID ビューをそれらに割り当てることはできません。さらに、デフォルトビューは常に AD ユーザーまたはグループに適用されます。

8.2.1. ID ビューおよび SSSD

管理者がクライアントの別の ID ビューを適用する場合、この ID ビューを適用するクライアントおよびその他すべてのクライアントは SSSD サービスを再起動します。さらに、新規の ID ビューが UID または GID を変更する場合、この ID ビューを適用するクライアントおよびその他すべてのクライアントは SSSD キャッシュをクリアする必要があります。

注記

ID ビューを適用すると、特定の最適化および ID ビューが同時に実行されなくなるので、SSSD パフォーマンスに負の影響が及ぶ可能性があります。

たとえば ID ビューは、SSSD がサーバー上のグループを検索するプロセスを最適化することを防ぎます。ID ビューでは、グループ名が上書きされる場合にグループメンバー名の返された一覧で SSSD がすべてのメンバーを検査する必要があります。ID ビューがないと、SSSD はグループオブジェクトのメンバー属性からユーザー名のみを収集します。SSSD キャッシュが空になるか、またはキャッシュをクリアした後にすべてのエントリーが無効になる場合などに負の影響が出ることがあります。

ID ビューはクライアント側に適用されます。これは、IdM の以前のバージョンを実行するクライアントはデフォルト信頼ビューのみを表示することを意味します。クライアントが別の ID ビューを必要とする場合は、クライアント上の SSSD をID ビューサポートのあるバージョンに更新するか、またはクライアントにコンパクトな LDAP ツリーを使用させるようにします。

8.2.2. Web UI からの ID ビューの管理

IdM Web UI から ID ビューを管理するには、**IPA Server** メインタブを開いてから**ID Views** サブタブを選択します。

新規 ID ビューを追加するには、以下を実行します。

1. すべての ID ビューの一覧の上で **Add** をクリックします。

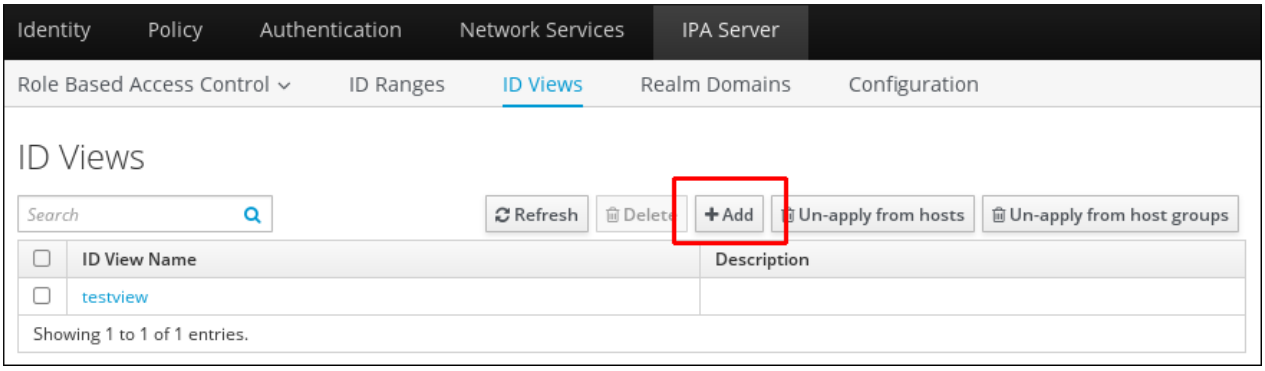


図8.1 新規 ID ビューの追加

2. 表示されるフォームに、新規 ID ビューについての情報を記載します。

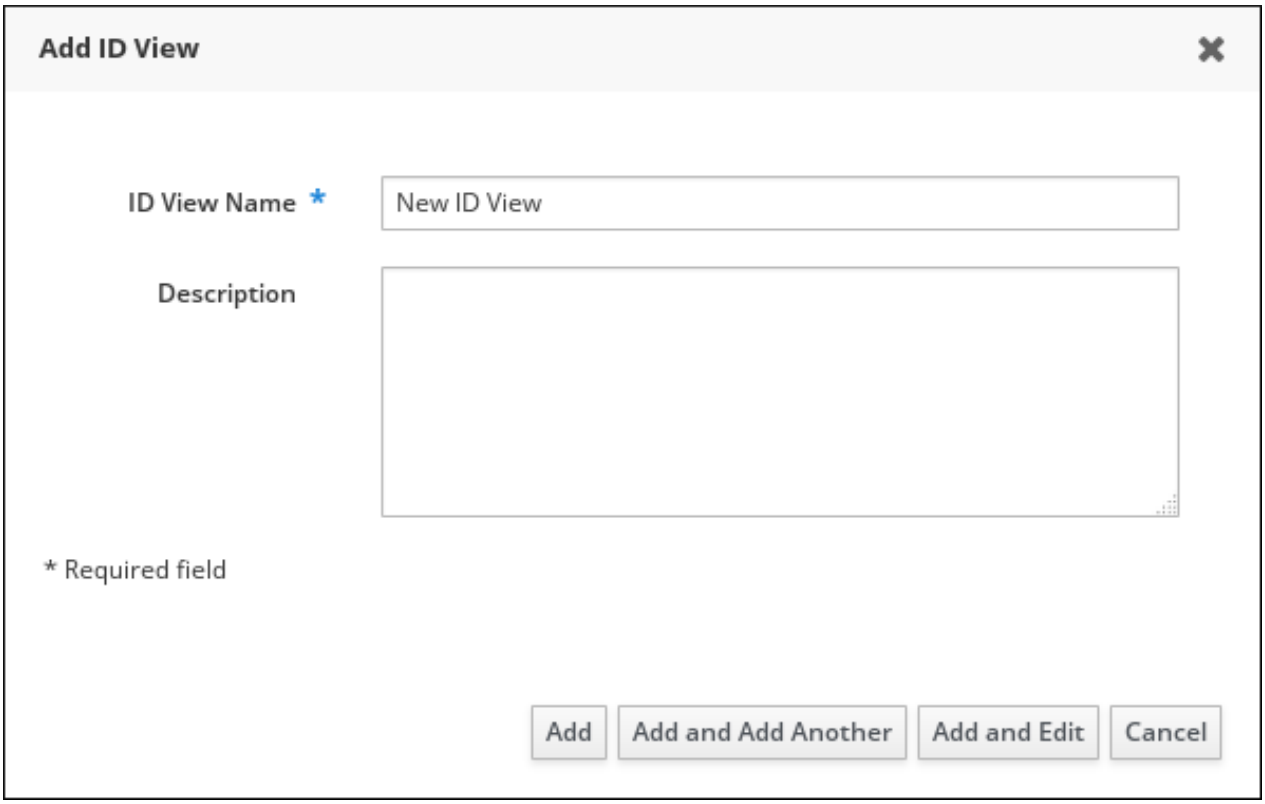


図8.2 新規 ID ビューを追加するためのフォーム

3. フォームの下にある **Add** ボタンをクリックします。

ID ビューのプロパティーを定義するには、以下を実行します。

1. ID ビューの一覧にある ID ビューの名前をクリックしてから、適切なタブを選択します。

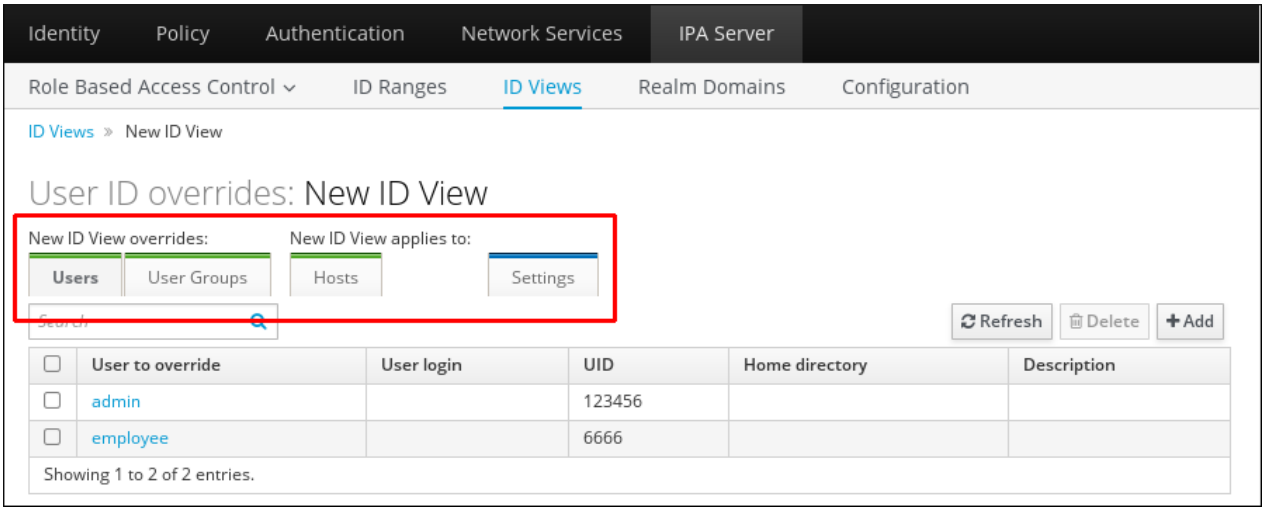


図8.3 ID ビュータブ

2. **Users** は ID ビューが上書きするユーザー属性のユーザー一覧を表示します。

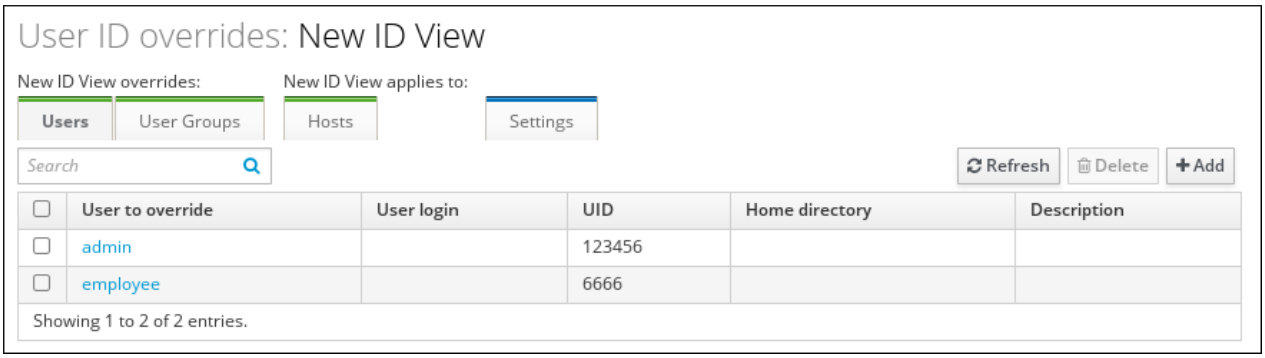


図8.4 ユーザー上書きの追加

新規ユーザーの上書きを作成するために **Add** をクリックします。ユーザー属性の新規の値を入力するように求められます。

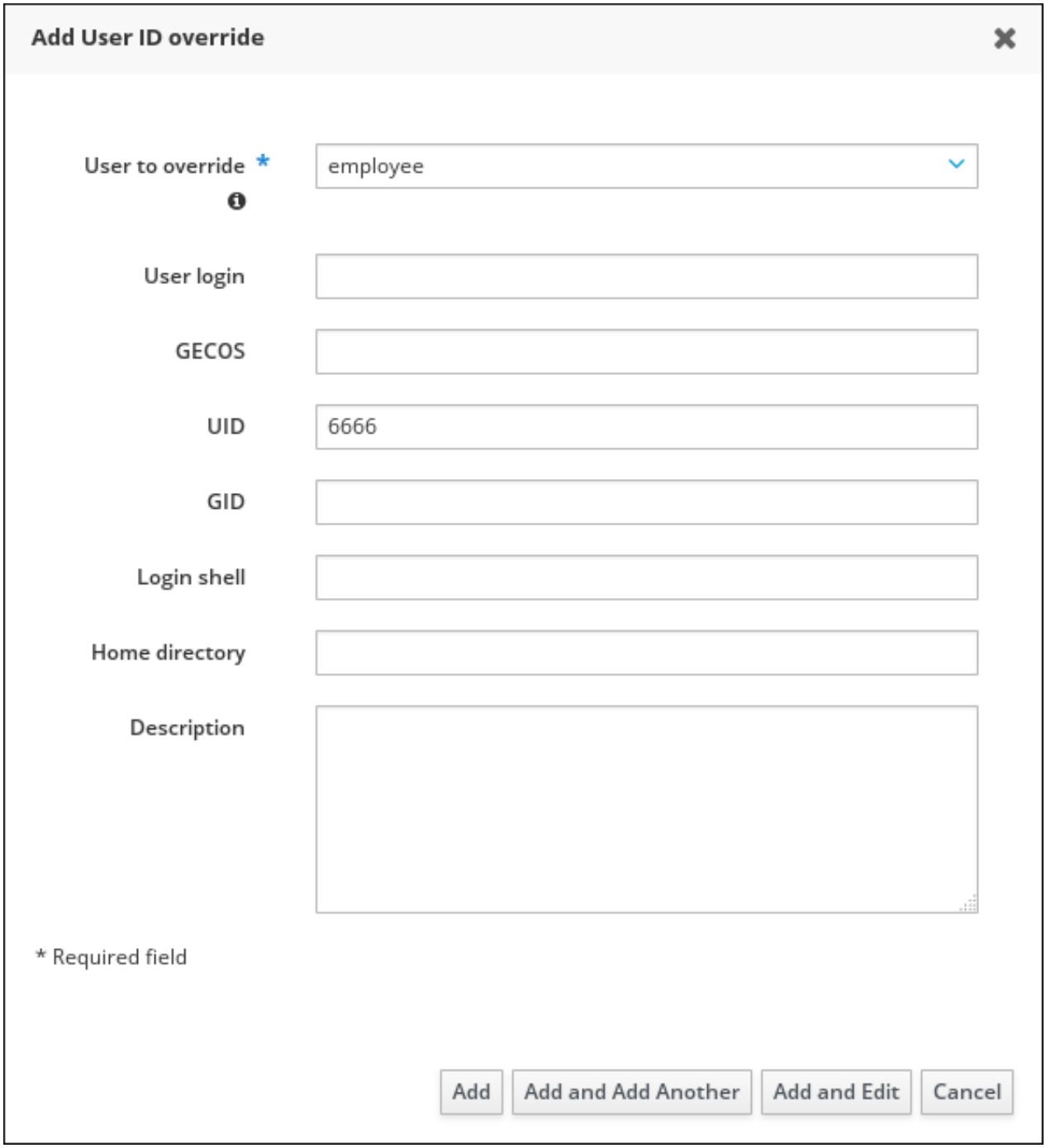


図8.5 ユーザー上書きの追加

選択されたユーザー上書きを削除するために、**Delete** をクリックします。

3. **User Groups** は、ID ビューが上書きするグループのユーザーグループ一覧を表示します。

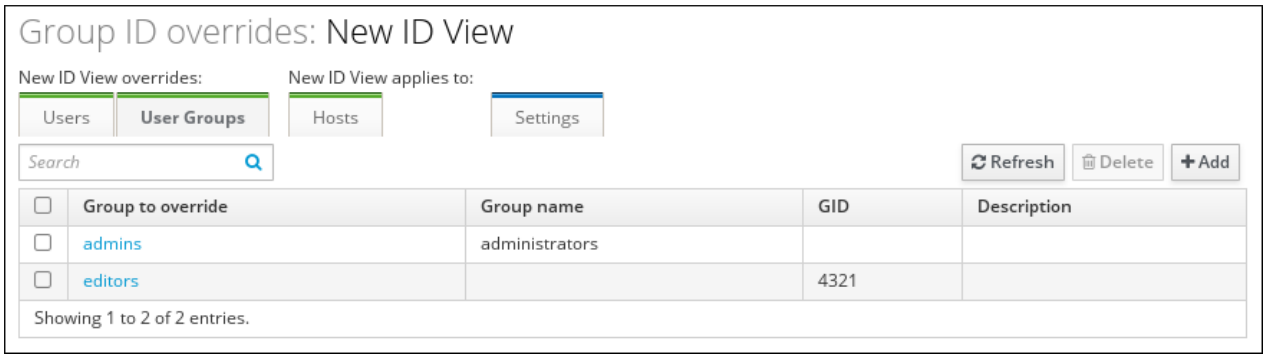


図8.6 ユーザーグループタブ

新規グループの上書きを作成するために **Add** をクリックします。グループ属性の新規の値を入力するように求められます。

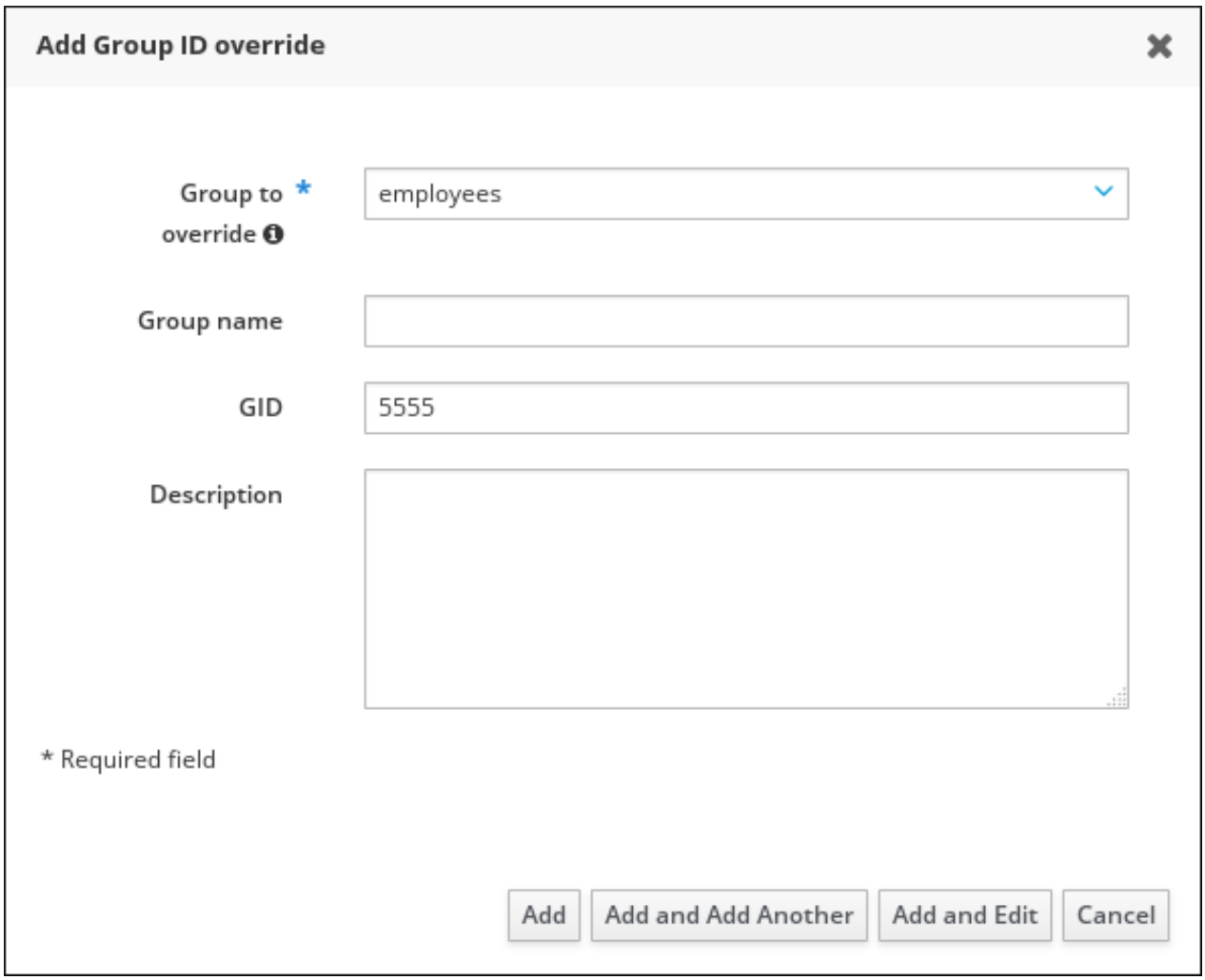


図8.7 グループ上書きの追加

選択されたグループ上書きを削除するために、**Delete** をクリックします。

4. **Hosts** は、ID ビューが適用されるホストまたはホストグループの一覧を表示します。

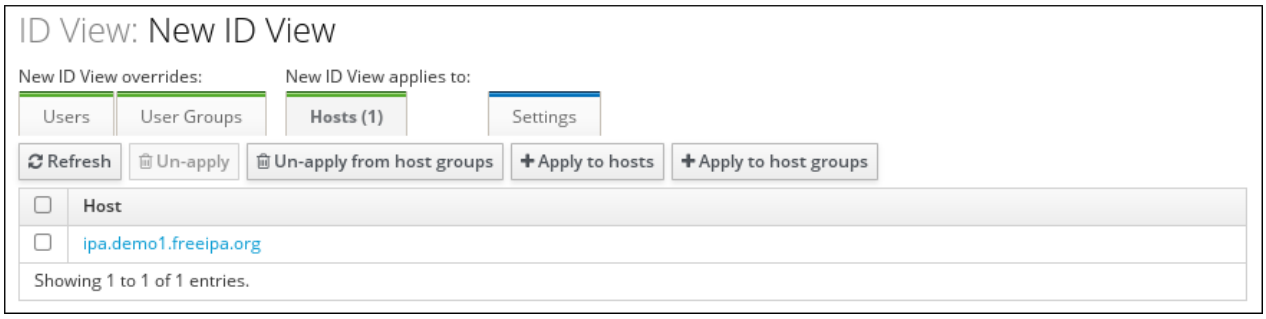


図8.8 Hosts タブ

新規ホストを追加するか、ホストグループに属するホストを追加するために **Apply to hosts** または **Apply to host groups** をクリックします。表示されるフォームで、必要なホストまたはホストグループを **Available** から **Prospective** 列に移動し、**Apply** をクリックします。

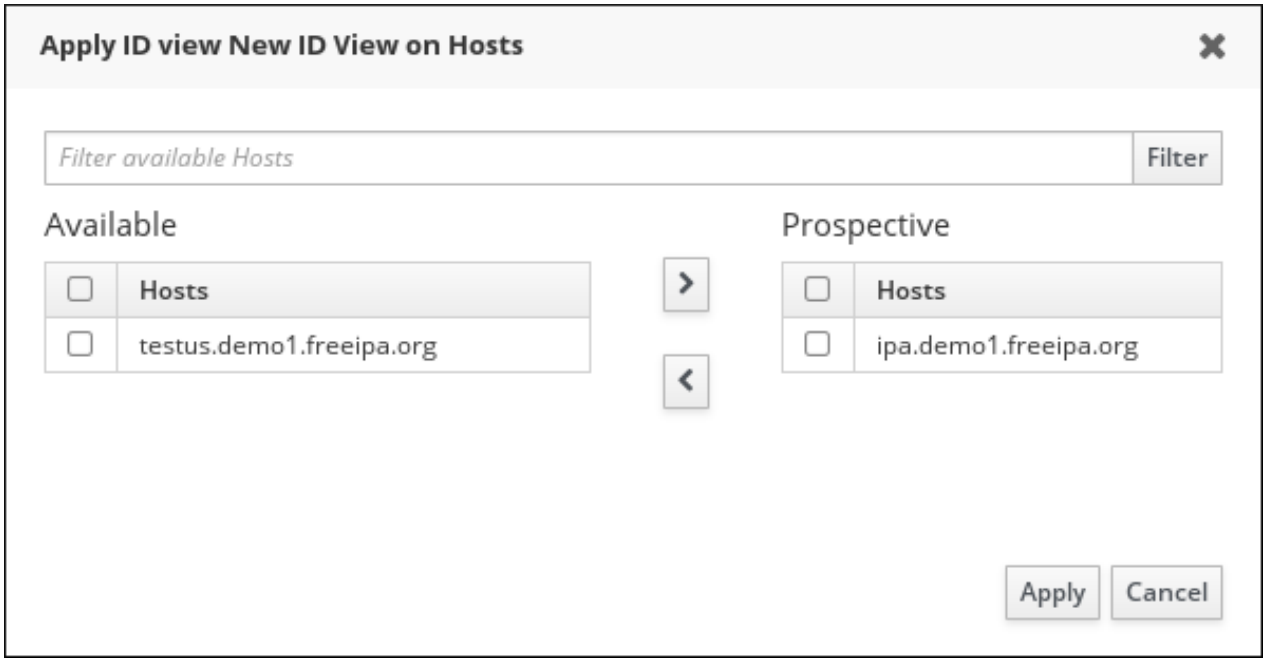


図8.9 ID ビューのホストへの適用

Un-apply は、ID ビューを指定したホストから削除します。**Un-apply from host groups** では ID ビューを指定したホストグループから削除することができます。

5. **Settings** では ID ビューの説明を変更することができます。

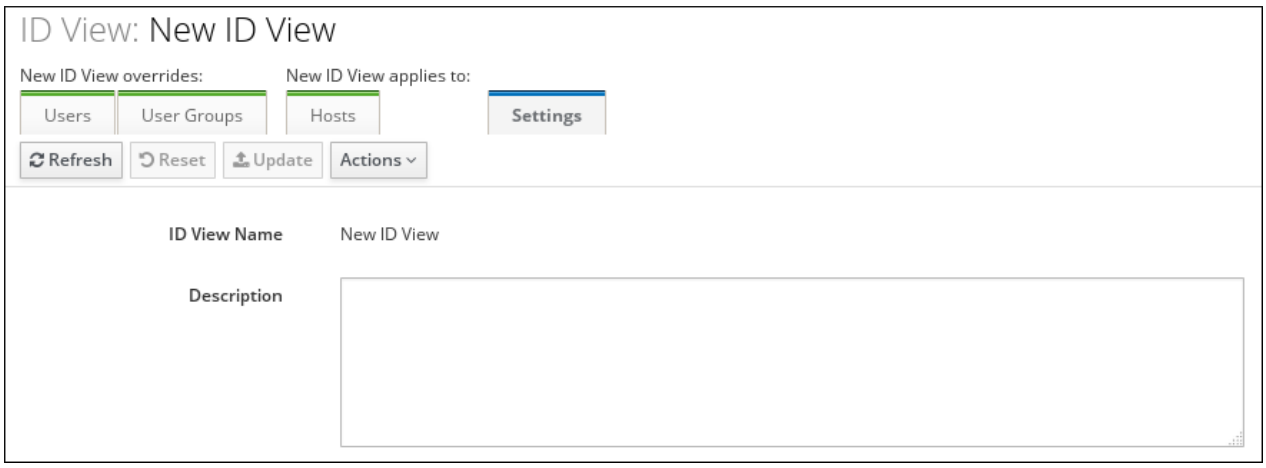


図8.10 Settings タブ

8.2.3. コマンドラインからの ID ビューの管理

コマンドラインで ID ビューを管理するには、以下のコマンドを使用します。

- **ipa idview-add** は新規 ID ビューを追加します。
- **ipa idview-apply** は ID ビューを指定されたホストまたはホストグループに適用します。それよりも前に適用された ID ビューは上書きされます。

- **ipa idview-del** は ID ビューを削除します。
- **ipa idview-find** は指定された ID ビューを検索します。
- **ipa idview-mod** は ID ビューを変更します。
- **ipa idview-show** は ID ビューについての情報を表示します。
- **ipa idview-unapply** は指定されたホストまたはホストグループから ID ビューを削除します。

グループおよびユーザー ID 上書きを管理するには、以下のコマンドを使用します。

- **ipa idoverridegroup-add** は、新規のグループ ID 上書きを追加します。

  ipa idoverrideuser-add は、新規のユーザー ID 上書きを追加します。
- **ipa idoverridegroup-del** は、グループ ID 上書きを削除します。

  ipa idoverrideuser-del は、ユーザー ID 上書きを削除します。
- **ipa idoverridegroup-find** は、指定されたグループ ID 上書きを検索します。

  ipa idoverrideuser-find は、指定されたユーザー ID 上書きを検索します。
- **ipa idoverridegroup-mod** は、グループ ID 上書きを変更します。

  ipa idoverrideuser-mod は、ユーザー ID 上書きを変更します。
- **ipa idoverridegroup-show** は、グループ ID 上書きについての情報を表示します。

  ipa idoverrideuser-show は、ユーザー ID 上書きについての情報を表示します。

上記のコマンドに渡すことのできるオプションについての詳細は、対応する man ページを参照するか、または **--help** オプションを追加してこれらの内のいずれかのコマンドを実行します。

例8.1 ホスト固有の ID ビューを使用した AD ユーザーの POSIX 属性および SSH キーの保存

testuser ユーザーの UID を 6666 に変更するには、以下を実行します。

1. **ipa idview-add** を使用して新規のホスト固有 ID を追加し、必要な値を指定します。

```
[user@client ~]$ ipa idview-add testview --desc "Our new host-
specific view"
----------------------------------------------
Added ID View "testview"
----------------------------------------------
  ID View Name: testview
  Description: Our new host-specific view
```

2. **ipa idoverrideuser-add** を実行し、必要な値を指定することにより ID 上書きを ID ビューに追加します。

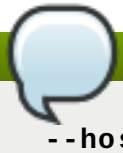
```
[user@client ~]$ ipa idoverrideuser-add testview
testuser@example.com --uid 6666
----------------------------------------------
Added User ID override "testuser@example.com"
```

```
-----------------------------------------
  Anchor to override: testuser@example.com
  UID: 6666
```

3. **ipa idview-apply** を実行し、**--hosts** オプションを使用してホストを指定することにより、ID ビューを特定のホストに適用します。

```
[user@client ~]$ ipa idview-apply testview --hosts
examplehost.com
-----------------------------------------
Applied ID View "testview"
-----------------------------------------
  hosts: examplehost.com
-----------------------------------------
Number of hosts the ID View was applied to: 1
```

同様の手順を使用して、GID および他の属性を上書きできます。詳細については、**ipa idoverrideuser-add --help** コマンドを実行してください。



注記

--hostgroups オプションは、ID ビューを指定されたホストグループに属するホストに適用し、**--hosts** オプションと同じ方法で使用できます。**--hostgroups** オプションは、ID ビューをホストグループ自体に関連付けません。これは指定されたホストグループのメンバーを拡張し、メンバーのすべてに対して **--hosts** を個別に適用します。

8.3. 同期ベースのソリューションから信頼ベースのソリューションへの移行

同期ベースの統合を使用する環境では、以下のステップを実行して、信頼ベースの統合に移行することができます。

1. 同期されたドメインで信頼を作成します。信頼を作成する方法についての詳細は、*5章Active Directory および Identity Management によるクロスレルム信頼の作成* を参照してください。
2. 同期されたすべてのユーザーまたはグループについては、IdM で生成される UID および GID を保持するためにホスト固有のビューまたはデフォルト信頼ビューで ID 上書きを個別に作成します。これを実行する方法については、例8.1「ホスト固有の ID ビューを使用した AD ユーザーの POSIX 属性および SSH キーの保存」 を参照してください。
3. 元の同期したユーザーまたはグループエントリーのバックアップコピーを作成します。
4. 元の同期したユーザーまたはグループエントリーをすべて削除します。

索引

シンボル

スキーマ

- Identity Management と Active Directory 間の相違点, Identity Management と Active Directory 間のユーザースキーマの相違点
  - cn, cn 属性の値

付録A 改訂履歴

改訂 7.0-13.2 **Sun Nov 29 2015** **Aiko Sasaki**
作者による一部内容変更の反映

改訂 7.0-13.1 **Wed Nov 18 2015** **Aiko Sasaki**
翻訳ファイルを XML ソースバージョン 7.0-13 と同期

改訂 7.0-13 **Wed Feb 25 2015** **Tomáš Čapek**
7.1 GA リリース用バージョン。

改訂 7.0-11 **Fri Dec 05 2014** **Tomáš Čapek**
スプラッシュページでの分類順序を更新して再構築。

改訂 7.0-7 **Mon Sep 15 2014** **Tomáš Čapek**
セクション 5.3 信頼の作成をコンテンツの更新のために一時的に削除。

改訂 7.0-5 **June 27, 2014** **Ella Deon Ballard**
Samba+Kerberos+Winbind の各章を改善。

改訂 7.0-4 **June 13, 2014** **Ella Deon Ballard**
Kerberos レルムの章を追加。

改訂 7.0-3 **June 11, 2014** **Ella Deon Ballard**
初期リリース。