Red Hat Enterprise Linux 8

RHEL 8 での Identity Management への移行

RHEL 7 IdM 環境の RHEL 8 へのアップグレードおよび FreeIPA または外部 LDAP ソリューションの IdM への移行

Last Updated: 2024-06-27

Red Hat Enterprise Linux 8 RHEL 8 での Identity Management への移行

RHEL 7 IdM 環境の RHEL 8 へのアップグレードおよび FreeIPA または外部 LDAP ソリューションの IdM への移行

法律上の通知

概要

Red Hat は、Red Hat Enterprise Linux (RHEL) 上の Identity Management (IdM) のみをサポートします。RHEL 7 上の IdM、他の Linux ディストリビューション上の FreeIPA、または LDAP ディレクトリーを実行している場合は、これらのソリューションを RHEL 8 上の IdM に移行できます。

目次

多様性を受け入れるオープンソースの強化

Red Hat では、コード、ドキュメント、Web プロパティーにおける配慮に欠ける用語の置き換えに取り組んでいます。まずは、マスター (master)、スレーブ (slave)、ブラックリスト (blacklist)、ホワイトリスト (whitelist) の 4 つの用語の置き換えから始めます。この取り組みは膨大な作業を要するため、今後の複数のリリースで段階的に用語の置き換えを実施してまいります。詳細は、Red Hat CTO である Chris Wright のメッセージ を参照してください。

Identity Management では、次のような用語の置き換えが計画されています。

- **ブラックリスト** から **ブロックリスト**
- **ホワイトリスト** から **許可リスト**
- **スレーブ** から **セカンダリー**
- **マスター** という言葉は、文脈に応じて、より正確な言葉に置き換えられています。
  - **IdM マスター** から **IdM サーバー**
  - **CA 更新マスター** から **CA 更新サーバー**
  - **CRL マスター** から **CRL パブリッシャーサーバー**
  - **マルチマスター** から **マルチサプライヤー**

RED HAT ドキュメントへのフィードバック (英語のみ)

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

Jira からのフィードバック送信 (アカウントが必要)

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

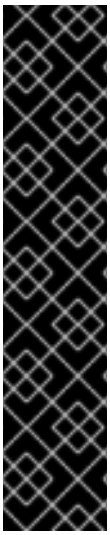
パート I. RHEL 7 から RHEL 8 への IDM の移行

第1章 RHEL 7 サーバーから RHEL 8 サーバーへの IDM 環境の移行

RHEL 7 IdM 環境を RHEL 8 にアップグレードするには、最初に新しい RHEL 8 IdM レプリカを RHEL 7 IdM 環境に追加し、RHEL 7 サーバーをリタイアさせる必要があります。

警告

- RHEL 8 への RHEL 7 IdM サーバーのインプレースアップグレードはサポートしていません。
- RHEL 6 以前のバージョンから RHEL 8 への直接移行はサポートされていません。IdM データを適切に更新するには、増分移行を実行する必要があります。
  たとえば、RHEL 6 IdM 環境を RHEL 8 に移行するには、次のコマンドを実行します。
  a. RHEL 6 サーバーから RHEL 7 サーバーに移行します。Red Hat Enterprise Linux 6 からバージョン 7 への Identity Management の移行 を参照してください。
  b. 本セクションで説明されているように、RHEL 7 サーバーから RHEL 8 サーバーに移行します。



重要

RHEL 8 は SPAKE と IdP 事前認証をサポートしていますが、RHEL 7 はサポートしていません。RHEL 7 IdM デプロイメントで SPAKE または IdP が有効になっている RHEL 8 サーバーを使用すると、ユーザーがログインできないなどの問題が発生する可能性があります。

Red Hat は、IdM デプロイメント内のすべてのサーバーをできるだけ早く移行し、古いシステムを長期間稼働させたままにしないことを強く推奨します。

詳細は以下を参照してください。

- https://access.redhat.com/solutions/7053377
- https://access.redhat.com/solutions/3529911

この手順では、すべての Identity Management (IdM) のデータおよび設定を、RHEL (Red Hat Enterprise Linux) 7 サーバーから RHEL 8 サーバーへ **移行** する方法を説明します。この手順を使用して、RHEL 以外の Linux ディストリビューション上の FreeIPA サーバーから RHEL 8 サーバー上の IdM に移行することもできます。

移行手順には、以下が含まれます。

1. RHEL 8 IdM サーバーを設定し、現在の RHEL 7 IdM 環境にレプリカとして追加します。詳細は、RHEL 8 レプリカのインストール を参照してください。
2. RHEL 8 サーバーを認証局 (CA) 更新サーバーにする。詳細は、RHEL 8 IdM サーバーへの CA 更新サーバーロールの割り当て を参照してください。

3. RHEL 7 サーバーで証明書失効リスト (CRL) の生成を停止し、CRL 要求を RHEL 8 にリダイレクトする。詳細は RHEL 7 IdM CA サーバーでの CRL 生成の停止 を参照してください。

4. RHEL 8 サーバーで CRL の生成を開始する。詳細は 新しい RHEL 8 IdM CA サーバーでの CRL 生成の開始 を参照してください。

5. 元の RHEL 7 CA 更新サーバーを停止して使用を中止する。詳細は RHEL 7 サーバーの停止および使用停止 を参照してください。

手順では、以下を前提としています。

- **rhel8.example.com** は、新しい CA 更新サーバーとなる RHEL 8 システムです。
- **rhel7.example.com** は、元の RHEL 7 CA 更新サーバーです。マスター CA 更新サーバーである Red Hat Enterprise Linux 7 サーバーを特定するには、任意の IdM サーバーで次のコマンドを実行します。

```
[root@rhel7 ~]# ipa config-show | grep "CA renewal"
IPA CA renewal master: rhel7.example.com
```

  IdM デプロイメントで認証局 (CA) を使用しない場合、RHEL 7 で実行している IdM サーバーは **rhel7.example.com** にすることができます。

注記

IdM デプロイメントが組み込み CA を使用する場合に **のみ**、以下のセクションの手順を完了します。

- RHEL 8 IdM サーバーへの CA 更新サーバーロールの割り当て
- RHEL 7 IdM CA サーバーでの CRL 生成の停止
- 新しい RHEL 8 IdM CA サーバーでの CRL 生成の開始

1.1. RHEL 7 から 8 への IDM の移行の前提条件

rhel7.example.com で、以下を行います。

1. システムを最新の RHEL 7 バージョンへアップグレードしている。
2. ドメインのドメインレベルが 1 に設定されていることを確認します。詳細は、RHEL 7 の **Linux ドメイン ID、認証、およびポリシーガイド** の ドメインレベルの表示と引き上げ を参照してください。
3. **ipa-*** パッケージを最新バージョンへ更新している。

```
[root@rhel7 ~]# yum update ipa-*
```

警告

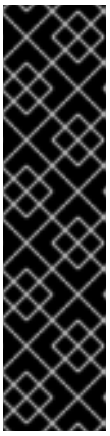
複数の Identity Management (IdM) サーバーをアップグレードする場合は、各アップグレードの間隔は少なくとも 10 分あけてください。

複数のサーバーで同時または間隔をあまりあけないでアップグレードを行うと、トポロジー全体でアップグレード後のデータ変更を複製する時間が足りず、複製イベントが競合する可能性があります。

rhel8.example.com で、以下を行います。

1. 最新バージョンの Red Hat Enterprise Linux がシステムにインストールされている。詳細は 標準的な RHEL 8 インストールの実行 を参照してください。
2. 時刻サーバー **rhel7.example.com** が以下と同期していることを確認します。

```
[root@rhel7 ~]# ntpstat
synchronised to NTP server (ntp.example.com) at stratum 3
   time correct to within 42 ms
   polling server every 1024 s
```

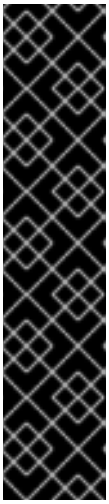


重要

RHEL 8 では IdM は独自の時刻サーバーを提供しません。つまり、**rhel8.example.com** に IdM をインストールしても、ホストに NTP サーバーはインストールされません。したがって、**ntp.example.com** などの別の NTP サーバーを使用する必要があります。詳細は chrony への移行 および IdM のタイムサービス要件 を参照してください。

rhel7.example.com は NTP サーバーロールで使用できますが、移行プロセスの一環としてサーバーの使用を停止します。したがって、**rhel8.example.com** は、代わりに **ntp.example.com** と直接同期する必要があります。これは、クライアントのインストールプロセス中に指定できます。

3. システムが、**rhel7.example.com** IdM サーバーが権威のあるドメインに登録されている IdM クライアントであることを確認します。詳細は、IdM クライアントのインストール を参照してください。



重要

クライアントをインストールするときに、**--ntp-server** オプションを使用して、前の手順のタイムサーバーを指定します。NTP サーバーのプールを使用している場合は、**--ntp-pool** オプションを使用します。

NTP サーバーを手動で指定しない場合は、DNS レコードから自動的に設定されます。これにより、**rhel8.example.com** が **rhel7.example.com** と同期することになります。これにより、RHEL 7 サーバーが使用できなくなると問題が発生します。

RHEL8 システムがすでに NTP クライアントとして適切に設定されている場合は、IdM クライアントのインストールを実行するときに **--no-ntp** オプションを使用できます。

4. システムが IdM サーバーのインストール要件を満たしていることを確認します。Preparing the system for IdM server installation を参照してください。
5. システムで IdM レプリカのインストールが許可されていることを確認します。Authorizing the installation of a replica on an IdM client を参照してください。
6. **ipa-*** パッケージを最新バージョンへ更新している。

```
[root@rhel7 ~]# yum update ipa-*
```

関連情報

- CA サービスの計画
- DNS サービスとホスト名の計画
- IdM と AD との間のフォレスト間の信頼の計画
- IdM サーバーに必要なパッケージのインストール
- Upgrading from RHEL 7 to RHEL 8

1.2. RHEL 8 レプリカのインストール

1. RHEL 7 環境に存在するサーバーのリストを表示します。

```
[root@rhel7 ~]# ipa server-role-find --status enabled --server rhel7.example.com
----------------------
3 server roles matched
----------------------
  Server name: rhel7.example.com
  Role name: CA server
  Role status: enabled

  Server name: rhel7.example.com
  Role name: DNS server
  Role status: enabled

  Server name: rhel7.example.com
```

```
  Role name: NTP server
  Role status: enabled
[... output truncated ...]
```

2. (必要に応じて)**rhel8.example.com** に **rhel7.example.com** が使用しているものと同じサーバーごとのフォワーダーを使用する場合は、**rhel7.example.com** のサーバーごとのフォワーダーを確認します。

```
[root@rhel7 ~]# ipa dnsserver-show rhel7.example.com
----------------------------
1 DNS server matched
----------------------------
  Server name: rhel7.example.com
  SOA mname: rhel7.example.com.
  Forwarders: 192.0.2.20
  Forward policy: only
----------------------------------------------------
Number of entries returned 1
----------------------------------------------------
```

3. IdM RHEL 7 サーバーのレプリカとして **rhel8.example.com** に IdM サーバーをインストールします。これには、NTP サーバーロールを除き、**rhel7.example.com** のサーバーロールをすべて含みます。上記の例からロールをインストールするには、**ipa-replica-install** コマンドで以下のオプションを使用します。

   - **--setup-ca**: Certificate System コンポーネントを設定する
   - **--setup-dns** および **--forwarder**: 統合 DNS サーバーを設定し、IdM ドメインの外に出る DNS クエリーを処理するようにサーバーごとのフォワーダーを設定する

注記

また、IdM デプロイメントが Active Directory (AD) と信頼関係にある場合は、**--setup-adtrust** オプションを **ipa-replica-install** コマンドに追加し、**rhel8.example.com** に AD 信頼機能を設定します。

IP アドレス 192.0.2.20 のサーバーごとのフォワーダーを使用する IP アドレス 192.0.2.1 の IdM サーバーを設定するには、以下を行います。

```
[root@rhel8 ~]# ipa-replica-install --setup-ca --ip-address 192.0.2.1 --setup-dns --forwarder 192.0.2.20
```

DNS が正常に機能している場合は、**rhel8.example.com** が DNS の自動検出を使用してそれを見つけるため、RHEL 7 IdM サーバー自体を指定する必要はありません。

4. 必要に応じて、外部 **NTP** タイムサーバーの **_ntp._udp** サービス (SRV) レコードを、新しくインストールした IdM サーバーの DNS (**rhel8.example.com**) に追加します。RHEL 8 では IdM が独自の時刻サービスを提供しないため、この設定を行うことが推奨されます。IdM DNS に時刻サーバーの SRV レコードが存在すると、今後の RHEL 8 レプリカおよびクライアントインストールが、**rhel8.example.com**が使用する時刻サーバーと同期するように自動的に設定されます。これは、**--ntp-server** または **--ntp-pool** オプションがインストールコマンドラインインターフェイス (CLI) で指定されていない限り、**ipa-client-install** は **_ntp._udp** DNS エントリーを検索するためです。

検証

検証

1. IdM サービスが **rhel8.example.com** で稼働していることを確認します。

```
[root@rhel8 ~]# ipactl status
Directory Service: RUNNING
[... output truncated ...]
ipa: INFO: The ipactl command was successful
```

2. NTP サーバーロールを除き、**rhel8.example.com** のサーバーロールが、**rhel7.example.com** と同じであることを確認します。

```
[root@rhel8 ~]$ kinit admin
[root@rhel8 ~]$ ipa server-role-find --status enabled --server rhel8.example.com
----------------------
2 server roles matched
----------------------
  Server name: rhel8.example.com
  Role name: CA server
  Role status: enabled

  Server name: rhel8.example.com
  Role name: DNS server
  Role status: enabled
```

3. 必要に応じて、**rhel7.example.com** と **rhel8.example.com** 間のレプリカ合意の詳細を表示します。

```
[root@rhel8 ~]# ipa-csreplica-manage list --verbose rhel8.example.com
Directory Manager password:

rhel7.example.com
last init status: None
last init ended: 1970-01-01 00:00:00+00:00
last update status: Error (0) Replica acquired successfully: Incremental update succeeded
last update ended: 2019-02-13 13:55:13+00:00
```

4. (必要に応じて)IdM デプロイメントが AD と信頼関係にある場合は、それが機能していることを確認します。

   a. リンク: Kerberos 設定の確認

   b. **rhel8.example.com** で AD ユーザーの解決を試みます。

```
[root@rhel8 ~]# id aduser@ad.domain
```

5. **rhel8.example.com** が **NTP** サーバーと同期されていることを確認します。

```
[root@rhel8 ~]# chronyc tracking
Reference ID    : CB00710F (ntp.example.com)
Stratum         : 3
Ref time (UTC)  : Tue Nov 16 09:49:17 2021
[... output truncated ...]
```

関連情報

- DNS 設定の優先順位
- IdM のタイムサービス要件
- chrony への移行

1.3. RHEL 8 IDM サーバーへの CA 更新サーバーロールの割り当て

RHEL 8 サーバーを認証局 (CA) 更新サーバーにするには、次の手順に従います。

注記

IdM デプロイメントで組み込みの認証局 (CA) を使用する場合にのみ、これらの手順に従ってください。

rhel8.example.com で、新しい CA 更新サーバーとして **rhel8.example.com** を設定します。

1. CA サブシステム証明書の更新を処理するように **rhel8.example.com** を設定します。

```
[root@rhel8 ~]# ipa config-mod --ca-renewal-master-server rhel8.example.com
...
IPA masters: rhel7.example.com, rhel8.example.com
IPA CA servers: rhel7.example.com, rhel8.example.com
IPA NTP servers: rhel7.example.com, rhel8.example.com
IPA CA renewal master: rhel8.example.com
```

   出力で更新が成功したことを確認します。

2. **rhel8.example.com** で、証明書アップデータータスクを有効にします。
   a. **/etc/pki/pki-tomcat/ca/CS.cfg** 設定ファイルを開いて編集します。
   b. **ca.certStatusUpdateInterval** エントリーを削除するか、適切な間隔 (秒単位) に設定します。デフォルト値は **600** です。
   c. **/etc/pki/pki-tomcat/ca/CS.cfg** 設定ファイルを保存して閉じます。
   d. IdM サービスを再起動します。

```
[user@rhel8 ~]$ ipactl restart
```

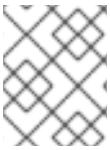
3. **rhel7.example.com** で、証明書アップデータータスクを無効にします。
   a. **/etc/pki/pki-tomcat/ca/CS.cfg** 設定ファイルを開いて編集します。
   b. **ca.certStatusUpdateInterval** を **0** に変更するか、以下のエントリーを追加します (存在しない場合)。

```
ca.certStatusUpdateInterval=0
```

   c. **/etc/pki/pki-tomcat/ca/CS.cfg** 設定ファイルを保存して閉じます。
   d. IdM サービスを再起動します。

```
[user@rhel7 ~]$ ipactl restart
```

1.4. RHEL 7 IDM CA サーバーでの CRL 生成の停止



注記

IdM デプロイメントで組み込みの認証局 (CA) を使用する場合にのみ、これらの手順に従ってください。

ipa-crlgen-manage コマンドを使用して、**rhel7.example.com** CA サーバーで証明書失効リスト (CRL) の生成を停止するには、次の手順に従います。

前提条件

- root としてログインしている。

手順

1. 必要に応じて、**rhel7.example.com** が CRL を生成しているかどうかを確認します。

```
[root@rhel7 ~]# ipa-crlgen-manage status
CRL generation: enabled
Last CRL update: 2019-10-31 12:00:00
Last CRL Number: 6
The ipa-crlgen-manage command was successful
```

2. **rhel7.example.com** サーバー上で CRL の生成を停止します。

```
[root@rhel7 ~]# ipa-crlgen-manage disable
Stopping pki-tomcatd
Editing /var/lib/pki/pki-tomcat/conf/ca/CS.cfg
Starting pki-tomcatd
Editing /etc/httpd/conf.d/ipa-pki-proxy.conf
Restarting httpd
CRL generation disabled on the local host. Please make sure to configure CRL generation on another master with ipa-crlgen-manage enable.
The ipa-crlgen-manage command was successful
```

3. 必要に応じて、**rhel7.example.com** サーバーが CRL の生成を停止しているかどうかを確認します。

```
[root@rhel7 ~]# ipa-crlgen-manage status
```

rhel7.example.com サーバーが CRL の生成を停止しました。次の手順では、 **rhel8.example.com** で CRL の生成を有効にします。

1.5. 新しい RHEL 8 IDM CA サーバーでの CRL 生成の開始

注記

IdM デプロイメントで組み込みの認証局 (CA) を使用する場合にのみ、これらの手順に従ってください。

前提条件

- **rhel8.example.com** マシンに root としてログインしている必要があります。

手順

1. **rhel8.example.com** で CRL の生成を開始するには、 **ipa-crlgen-manage enable** コマンドを使用します。

```
[root@rhel8 ~]# ipa-crlgen-manage enable
Stopping pki-tomcatd
Editing /var/lib/pki/pki-tomcat/conf/ca/CS.cfg
Starting pki-tomcatd
Editing /etc/httpd/conf.d/ipa-pki-proxy.conf
Restarting httpd
Forcing CRL update
CRL generation enabled on the local host. Please make sure to have only a single CRL generation master.
The ipa-crlgen-manage command was successful
```

2. CRL 生成が有効になっているかどうかを確認するには、**ipa-crlgen-manage status** コマンドを使用します。

```
[root@rhel8 ~]# ipa-crlgen-manage status
CRL generation: enabled
Last CRL update: 2019-10-31 12:10:00
Last CRL Number: 7
The ipa-crlgen-manage command was successful
```

1.6. RHEL 7 サーバーの停止および使用停止

1. 最新の変更を含むすべてのデータが **rhel7.example.com** から **rhel8.example.com** に正しく移行されていることを確認します。以下に例を示します。

   a. **rhel7.example.com** に新しいユーザーを追加します。

```
[root@rhel7 ~]# ipa user-add random_user
First name: random
Last name: user
```

   b. ユーザーが **rhel8.example.com** に複製されていることを確認します。

```
[root@rhel8 ~]# ipa user-find random_user
--------------
1 user matched
--------------
  User login: random_user
  First name: random
  Last name: user
```

2. Distributed Numeric Assignment (DNA) ID 範囲が **rhel8.example.com** に割り当てられていることを確認します。以下の方法のいずれかを使用します。

- 別のテストユーザーを作成して、**rhel8.example.com** で直接 DNA プラグインを有効にします。

```
[root@rhel8 ~]# ipa user-add another_random_user
First name: another
Last name: random_user
```

- 特定の DNA ID 範囲を **rhel8.example.com** に割り当てます。

  i. **rhel7.example.com** で、IdM ID 範囲を表示します。

```
[root@rhel7 ~]# ipa idrange-find
----------------
3 ranges matched
----------------
  Range name: EXAMPLE.COM_id_range
  First Posix ID of the range: 196600000
  Number of IDs in the range: 200000
  First RID of the corresponding RID range: 1000
  First RID of the secondary RID range: 100000000
  Range type: local domain range
```

  ii. **rhel7.example.com** で、割り当てられた DNA ID 範囲を表示します。

```
[root@rhel7 ~]# ipa-replica-manage dnarange-show
rhel7.example.com: 196600026-196799999
rhel8.example.com: No range set
```

  iii. セクションが **rhel8.example.com** で使用可能になるように、**rhel7.example.com** に割り当てられた DNA ID 範囲を減らします。

```
[root@rhel7 ~]# ipa-replica-manage dnarange-set rhel7.example.com
196600026-196699999
```

  iv. IdM ID 範囲の残りの部分を **rhel8.example.com** に割り当てます。

```
[root@rhel7 ~]# ipa-replica-manage dnarange-set rhel8.example.com
196700000-196799999
```

3. **rhel7.example.com** 上の全 IdM サービスを停止して、新しい **rhel8.example.com** サーバーへのドメイン検索を実施します。

```
[root@rhel7 ~]# ipactl stop
Stopping CA Service
Stopping pki-ca:                                   [  OK  ]
Stopping HTTP Service
Stopping httpd:                                    [  OK  ]
Stopping MEMCACHE Service
Stopping ipa_memcached:                            [  OK  ]
Stopping DNS Service
Stopping named: .                                  [  OK  ]
Stopping KPASSWD Service
Stopping Kerberos 5 Admin Server:                  [  OK  ]
Stopping KDC Service
```

```
Stopping Kerberos 5 KDC:                                [  OK  ]
Stopping Directory Service
Shutting down dirsrv:
    EXAMPLE-COM...                                      [  OK  ]
    PKI-IPA...                                          [  OK  ]
```

この後に、**ipa** ユーティリティーを使用すると、Remote Procedure Call (RPC) で新規サーバーに接続します。

4. RHEL 8 サーバーで削除コマンドを実行して、トポロジーから RHEL 7 サーバーを削除します。詳細は、IdM サーバーのアンインストール を参照してください。

関連情報

- ID 範囲を手動で調整

第2章 RHEL 7 から RHEL 8 への IDM クライアントのアップグレード

IdM サーバーとは異なり、IdM クライアントの RHEL 7 から RHEL 8 へのインプレースアップグレードはサポート対象です。

RHEL 8 では、IdM 環境の認証を担当するサービスである System Security Services Daemon (SSSD) から一般的でないオプションおよび未使用の機能が削除されました。これらのオプションを削除する手順については、以下のセクションを参照してください。

- RHEL 8 へのアップグレード後の SSSD 設定の更新
- RHEL 8 で削除された SSSD 機能のリスト

2.1. RHEL 8 へのアップグレード後の SSSD 設定の更新

Identity Management (IdM) クライアントを Red Hat Enterprise Linux (RHEL) 7 から RHEL 8 にアップグレードした後、特定の SSSD 設定オプションがサポートされなくなる場合があります。**leapp** アップグレードアプリケーションは、生成されたアップグレード前レポートで、そのようなオプションに関する詳細を提供する場合があります。

以下の手順では、これらの問題に対処するために SSSD 設定を更新する方法を説明します。

前提条件

- IdM クライアントが RHEL 7 から RHEL 8 にアップグレードしている。
- **/etc/sssd/sssd.conf** を編集する **root** 権限がある。

2.1.1. ローカル ID プロバイダーから ファイル ID プロバイダーへの切り替え

以下のエラーが表示される場合は、**ローカル** ID プロバイダーは **ファイル** ID プロバイダーに置き換えます。

> SSSD Domain "example.com": local provider is no longer supported and the domain will be ignored.
> **Local provider is no longer supported.**

手順

1. **ローカル** ID プロバイダーで取得したユーザーおよびグループは、**/etc/passwd** と **/etc/group** ファイルにも含まれていることを確認します。このファイルに存在していると、**ファイル** プロバイダーがユーザーおよびグループにアクセスできるようになります。

   a. ユーザーを作成する必要がある場合は、**useradd** コマンドを使用します。UID を指定する必要がある場合は、**-u** オプションを追加します。

   ```
   [root@client ~]# useradd -u 3001 username
   ```

   b. グループを作成する必要がある場合は、**groupadd** コマンドを使用します。GID を指定する必要がある場合は、**-g** オプションを追加します。

   ```
   [root@client ~]# groupadd -g 5001 groupname
   ```

2. テキストエディターで **/etc/sssd/sssd.conf** 設定ファイルを開きます。
3. **id_provider=local** は **id_provider=files** に置き換えます。

```
[domain/example.com]
id_provider = files
...
```

4. **/etc/sssd/sssd.conf** 設定ファイルを保存します。
5. SSSD を再起動して、設定の変更を読み込みます。

```
[root@client ~]# systemctl restart sssd
```

2.1.2. 非推奨のオプションの削除

非推奨のオプションに関する以下のエラーのいずれかが表示される場合は、このようなオプションを **/etc/sssd/sssd.conf** 設定ファイルからを削除することを推奨します。

```
SSSD Domain "example.com": option ldap_groups_use_matching_rule_in_chain has no longer any effect
Option ldap_groups_use_matching_rule_in_chain was removed and it will be ignored.
```

```
SSSD Domain "example.com": option ldap_initgroups_use_matching_rule_in_chain has no longer any effect
Option ldap_initgroups_use_matching_rule_in_chain was removed and it will be ignored.
```

手順

1. テキストエディターで **/etc/sssd/sssd.conf** 設定ファイルを開きます。
2. **ldap_groups_use_matching_rule_in_chain** または **ldap_initgroups_use_matching_rule_in_chain** オプションが存在する場合は削除します。
3. **/etc/sssd/sssd.conf** 設定ファイルを保存します。
4. SSSD を再起動して、設定の変更を読み込みます。

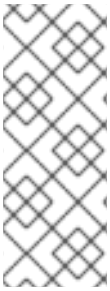
```
[root@client ~]# systemctl restart sssd
```

2.1.3. sudo ルールでのワイルドカードの使用の有効化

以下の警告は、RHEL 8 のデフォルトでワイルドカードが含まれる **sudo** ルールは機能しないことを示します。これは、**ldap_sudo_include_regexp** オプションがデフォルトで **false** に設定されるようになったためです。

```
SSSD Domain "example.com": sudo rules containing wildcards will stop working.
Default value of ldap_sudo_include_regexp changed from true to false for performance reason.
```

sudo ルールでワイルドカードを使用して、ワイルドカードの検索を有効化する場合は、**ldap_sudo_include_regexp** オプションを **true** に設定してください。



注記

Red Hat では **sudo** ルールでワイルドカードを使用した検索の利用は推奨していません。

ldap_sudo_include_regexp オプションが **true** に設定されている場合には、SSSD は **sudoHost** 属性にワイルドカードが含まれるすべての **sudo** ルールをダウンロードするため、LDAP 検索パフォーマンスに悪影響があります。

手順

1. テキストエディターで **/etc/sssd/sssd.conf** 設定ファイルを開きます。
2. **example.com** ドメインで **ldap_sudo_include_regexp=true** と設定します。

```
[domain/example.com]
...
ldap_sudo_include_regexp = true
...
```

3. **/etc/sssd/sssd.conf** 設定ファイルを保存します。
4. SSSD を再起動して、設定の変更を読み込みます。

```
[root@client ~]# systemctl restart sssd
```

2.2. RHEL 8 で削除された SSSD 機能のリスト

以下の SSSD 機能は、RHEL 8 では削除されました。

ローカル ID プロバイダーが削除される

ローカル SSSD キャッシュからユーザー情報を提供していた **ローカル** ID プロバイダーが RHEL 7 で非推奨になり、RHEL 8 でサポート対象外になりました。**/etc/sssd/sssd.conf** 設定でドメインが **id_provider=local** に指定されている場合には、SSSD はこのドメインを無視して通常どおりに起動します。

ローカル ドメインのユーザーおよびグループを管理するコマンドラインツールが削除される

ローカル ドメインだけが対象の以下のコマンドが削除されました。

- **sss_useradd**
- **sss_userdel**
- **sss_groupadd**
- **sss_groupdel**

ldap_groups_use_matching_rule_in_chain オプションのサポートが削除される

この Active Directory 固有のオプションは、パフォーマンスに好影響がないので、RHEL 8 **sssd.conf** 設定では無視されます。

ldap_initgroups_use_matching_rule_in_chain オプションのサポートが削除される

この Active Directory 固有のオプションは、パフォーマンスに好影響がないので、RHEL 8 **sssd.conf** 設定では無視されます。

ldap_sudo_include_regexp オプションがデフォルトで false に設定されるように

RHEL 7 では、このオプションはデフォルトで **true** に設定されていました。このオプションが **true** に設定されている場合には、SSSD は **sudoHost** 属性にワイルドカードが含まれるすべての **sudo** ルールをダウンロードするため、LDAP 検索パフォーマンスに悪影響があります。

sssd-secrets レスポンダーが削除される

Kerberos Cache Manager (KCM) は **sssd-secrets** レスポンダーに依存しなくなり、他の IdM プロセスで使用されないため、削除されました。

2.3. 関連情報

- RHEL 8 へのアップグレードの詳細は RHEL 7 から RHEL 8 へのアップグレード を参照してください。

パート II. 外部ソースから IDM への移行

第3章 非 RHEL LINUX ディストリビューション上の FREEIPA から RHEL 8 上の IDM への移行

非 RHEL Linux ディストリビューション上の FreeIPA デプロイメントを RHEL 8 サーバー上の Identity Management (IdM) デプロイメントに移行するには、最初に新しい RHEL 8 IdM 認証局 (CA) レプリカを既存の FreeIPA 環境に追加し、証明書関連の転送を行い、非 RHEL FreeIPA サーバーを廃止する必要があります。

警告

Convert2RHEL ツールを使用した、RHEL 以外の FreeIPA サーバーから RHEL 8 IdM サーバーへのインプレース変換の実行はサポートされていません。

移行を実行するには、RHEL 7 サーバーとして機能する非 RHEL FreeIPA CA レプリカを使用して、IdM 環境を RHEL 7 サーバーから RHEL 8 サーバーに移行する と同じ手順に従います。

1. RHEL 8 サーバーを設定し、非 RHEL Linux ディストリビューション上の現在の FreeIPA 環境に IdM レプリカとして追加します。詳細は、RHEL 8 レプリカのインストール を参照してください。
2. RHEL 8 レプリカを認証局 (CA) 更新サーバーにします。詳細は、RHEL 8 IdM サーバーへの CA 更新サーバーロールの割り当てを参照してください。
3. 非 RHEL サーバーでの証明書失効リスト (CRL) の生成を停止し、CRL 要求を RHEL 8 レプリカにリダイレクトします。詳細は RHEL 7 IdM CA サーバーでの CRL 生成の停止 を参照してください。
4. RHEL 8 サーバーで CRL の生成を開始します。詳細は 新しい RHEL 8 IdM CA サーバーでの CRL 生成の開始 を参照してください。
5. 元の非 RHEL FreeIPA CA 更新サーバーを停止して廃止します。詳細は RHEL 7 サーバーの停止および使用停止 を参照してください。

関連情報

- RHEL 7 サーバーから RHEL 8 サーバーへの IdM 環境の移行

第4章 LDAP ディレクトリーから IDM への移行

ID および認証ルックアップ用に LDAP サーバーをデプロイしている場合は、ルックアップサービスを Identity Management (IdM) に移行できます。IdM では、以下のタスクに役立つ移行ツールを利用できます。

- データを失うことなく、パスワードやグループメンバーシップなどのユーザーアカウントを転送するタスク。
- クライアントで高価な設定更新を回避するタスク。

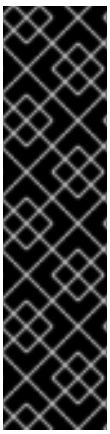
ここで説明する移行プロセスは、LDAP に 1 つ、IdM に 1 つの名前空間がある単純な導入シナリオを想定しています。複数の名前空間やカスタムスキーマがある場合など、より複雑な環境については、Red Hat サポートサービスにお問い合わせください。

4.1. LDAP から IDM への移行に関する考慮事項

LDAP サーバーから Identity Management (IdM) に移行するプロセスには、以下の段階があります。

- **クライアント**の移行このステージは慎重に計画してください。現在のインフラストラクチャーの各クライアントが使用するサービスを判断します。これには、Kerberos や Systems Security Services Daemon (SSSD) などが含まれます。次に、最終的な IdM デプロイメントで使用できるサービスを決定します。詳細は、Planning the client configuration when migrating from LDAP to IdM を参照してください。
- **データ** の移行
- **パスワード** の移行このステージは慎重に計画してください。IdM では、パスワードのほかに、すべてのユーザーアカウントに Kerberos ハッシュが必要です。パスワードに関する考慮事項と移行パスの一部は、Planning password migration when migrating from LDAP to IdM で説明しています。

最初にサーバー部分を移行してからクライアントを移行するか、最初にクライアントを移行してからサーバーを移行することができます。2 つのタイプの移行の詳細は、LDAP to IdM migration sequence を参照してください。



重要

実際に LDAP 環境の移行に入る前に、LDAP のテスト環境を設定して移行プロセスを検証することを強く推奨します。環境をテストする場合は、以下を行います。

1. IdM でテストユーザーを作成し、移行したユーザーの出力を、テストユーザーの出力と比較します。移行したユーザーに、テストユーザーに存在する属性およびオブジェクトクラスの最小セットが含まれていることを確認します。
2. IdM にあるように、移行したユーザーの出力を、元の LDAP サーバーにあるように、ソースユーザーと比較します。インポートされた属性が 2 回コピーされていないこと、およびそれらが正しい値を持っていることを確認してください。

4.2. LDAP から IDM への移行時のクライアント設定の計画

Identity Management は、さまざまなレベルの機能性、柔軟性、安全性で多数の異なるクライアント設定に対応することができます。オペレーティングシステムと、IT メンテナンスの優先度に基づいて、各クライアントに最適な設定を決定します。クライアントの機能領域も考慮してください。開発マシンは通常、実稼働サーバーやユーザーラップトップとは異なる設定を必要とします。

重要

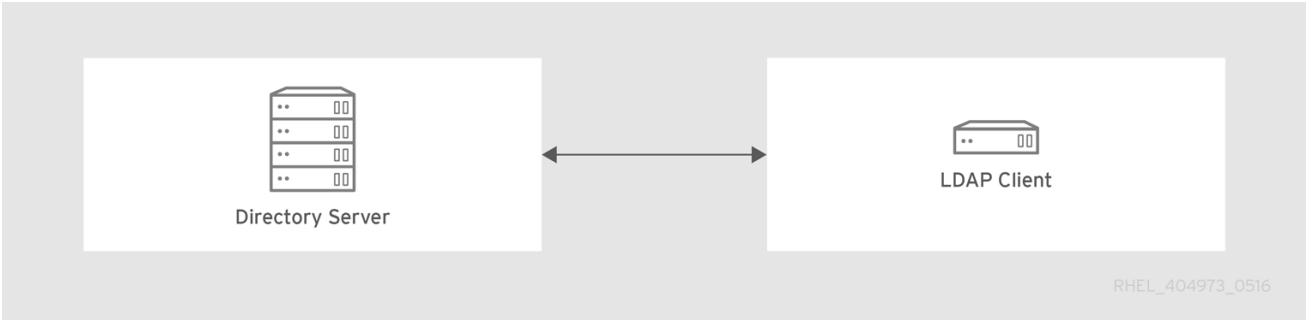
ほとんどの環境では、クライアントが IdM ドメインに接続する方法が混在しています。管理者は各クライアント別に最適となるシナリオを決定しなければなりません。

4.2.1. 初期の移行前のクライアント設定

Identity Management (IdM) でクライアント設定の詳細を決定する前に、現在の移行前設定の詳細を確立します。

移行予定の LDAP デプロイメントの初期の状態の場合、ほとんど全てに ID および認証サービスを提供している LDAP サービスがあります。

図4.1 基本的な LDAP ディレクトリーとクライアント設定



Linux および Unix のクライアントは PAM_LDAP と NSS_LDAP ライブラリーを使用して、LDAP サービスに直接接続します。これらのライブラリーにより、クライアントは、**/etc/passwd** または **/etc/shadow** にデータが格納されているかのように LDAP ディレクトリーからユーザー情報を取得できます。現実的には ID 検索に LDAP、認証に Kerberos や別の設定を使用している場合など、インフラストラクチャーはもう少し複雑になる場合があります。

LDAP ディレクトリーと Identity Management (IdM) サーバーの間には、特にスキーマサポートとディレクトリーツリーに構造的な違いがあります。これらの相違点の背景については、LDAP から IdM への移行時のクライアント設定の計画 の**IdM と標準 LDAP ディレクトリー**の比較 セクションを参照してください。このような相違は、特にディレクトリーツリーのデータに影響を及ぼし、エントリー名に影響を及ぼします。ただし、この相違はクライアントの設定、およびクライアントの IdM への移行にはほとんど影響を及ぼしません。

4.2.2. RHEL クライアントに推奨される設定

注記

説明されているクライアント設定は、最新バージョンの SSSD および **ipa-client** パッケージに対応する RHEL 6.1 以降および RHEL 5.7 以降でのみ対応しています。古いバージョンの RHEL は、対応している別の設定 の説明に従って設定できます。

Red Hat Enterprise Linux (RHEL) の System Security Services Daemon (SSSD) は、特殊な PAM ライブラリーおよび NSS ライブラリー (**pam_sss** および **nss_sss**) を使用します。このライブラリーを使用すると、SSSD が Identity Management (IdM) と非常に密接に統合し、完全な認証機能および ID 機能の利点を活用できます。SSSD には、ID 情報のキャッシュなど、便利な機能が多数あります。これにより、中央サーバーへの接続が失われた場合でも、ユーザーがログインできるようになります。

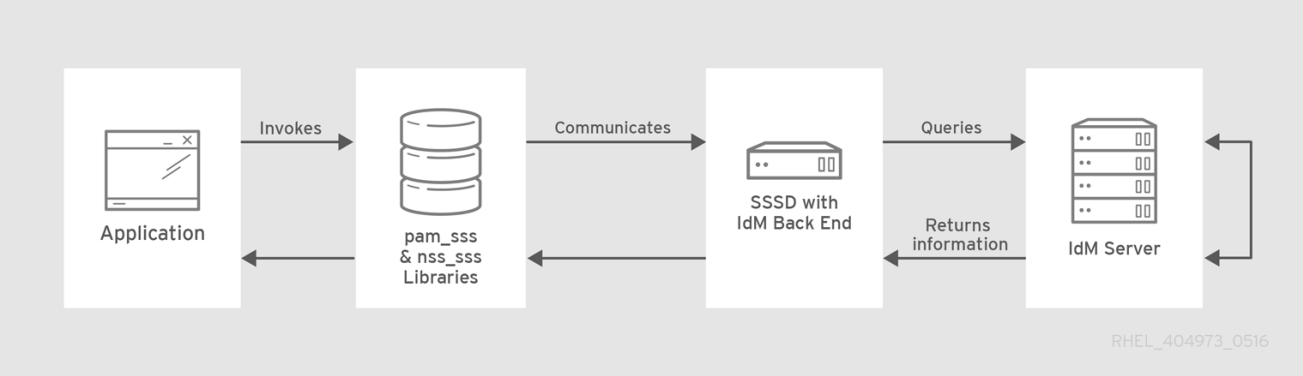
汎用の LDAP ディレクトリーサービス (**pam_ldap** および **nss_ldap** を使用する) とは異なり、SSSD は **ドメイン** 定義によって ID 情報と認証情報間の関係を確立します。SSSD のドメインは、以下のバックエンド機能を定義します。

- 認証
- Identity ルックアップ
- アクセス
- パスワードの変更

次に、SSSD ドメインは、**プロバイダー** を使用してこれらの機能のいずれかまたはすべてに情報を提供するように設定されます。ドメイン環境設定には常に**ID**プロバイダーが必要です。他の 3 つのプロバイダーはオプションです。認証、アクセス、またはパスワードプロバイダーが定義されていない場合は ID プロバイダーがその機能に使用されます。

SSSD は、そのすべてのバックエンド機能に IdM を使用できます。これは、LDAP ID の汎用プロバイダーや Kerberos 認証とは異なり、IdM 機能のすべての範囲を提供するため理想的な設定です。たとえば、SSSD では 日常的な運用時に IdM でセキュリティー機能やホストベースのアクセス制御ルールを有効化させることができます。

図4.2 IdM バックエンドのあるクライアントおよび SSSD



ipa-client-install スクリプトは、すべてのバックエンドサービスに IdM を使用するように SSSD を自動的に設定するため、RHEL クライアントはデフォルトで推奨される設定で設定されます。

関連情報

- SSSD とその利点について

4.2.3. 推奨設定以外で対応している設定

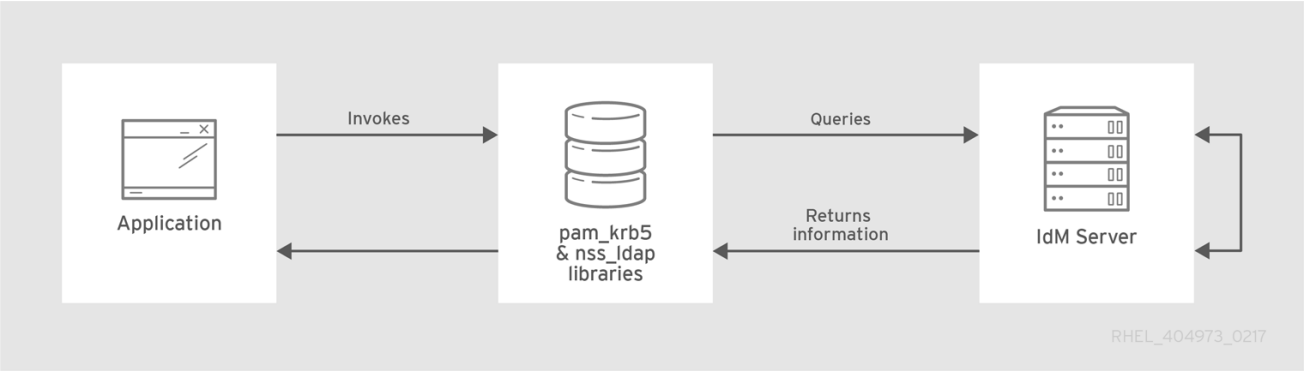
Mac、Solaris、HP-UX、AIX、Scientific Linux などの Unix および Linux システムでは IdM で管理されるすべてのサービスに対応していますが SSSD は使用しません。同様に、古い Red Hat Enterprise Linux (RHEL) バージョン (特に 6.1 および 5.6) は SSSD に対応していますが、IdM を ID プロバイダーとしてサポートしていない古いバージョンがあります。

システムで最新バージョンの SSSD を使用できない場合は、以下の方法でクライアントを設定できます。

- クライアントは、**nss_ldap** を使用して、ID 検索で LDAP ディレクトリーサーバーであるかのように IdM サーバーに接続します。
- クライアントは、**pam_krb5** を使用して、通常の Kerberos KDC であるかのように IdM サーバーに接続します。

IdM サーバーを ID プロバイダーおよび Kerberos 認証ドメインとして使用するように**古いバージョンの SSSD を持つ RHEL クライアント**を設定する方法は、RHEL 7 **システムレベルの認証ガイド** の SSSD の ID プロバイダーおよび認証プロバイダー セクションを参照してください。

図4.3 LDAP および Kerberos を使用するクライアントおよび IdM



一般的には、クライアントで可能な限り安全な設定を使用することがベストプラクティスとなります。これは、ID の SSSD または LDAP、および認証の Kerberos を意味します。ただし、メンテナンス状況や IT 構造によっては、最も単純な例 (クライアントで **nss_ldap** ライブラリーおよび **pam_ldap** ライブラリーを使用して ID と認証の両方を提供するように LDAP を設定) に頼る必要があります。

4.3. LDAP から IDM への移行時のパスワードの移行の計画

ユーザーを LDAP から Identity Management (IdM) に移行する前に決定すべき重大な問題は、ユーザーパスワードを移行するかどうかです。以下のオプションを設定できます。

パスワードを使用しないユーザーの移行

より迅速に実行できますが、管理者とユーザーによるより多くの手作業が必要です。特定の状況では、これが唯一の選択肢となります (元の LDAP 環境にクリアテキストのユーザーパスワードが保存されている 場合や パスワードが IdM で定義されているパスワードポリシーの要件を満たしていない 場合など)。
パスワードを使用せずにユーザーアカウントを移行する場合は、すべてのユーザーパスワードをリセットします。移行したユーザーには、最初のログイン時に変更する一時パスワードが割り当てられます。パスワードのリセット方法は、RHEL 7 IdM ドキュメント Changing and resetting user passwords を参照してください。

パスワードを使用したユーザーの移行

移行はよりスムーズになりますが、移行および移行プロセスで LDAP ディレクトリーと IdM を並列に管理することも必要になります。これは、IdM がデフォルトで認証に Kerberos を使用し、各ユーザーには、標準ユーザーパスワードのほかに、IdM Directory Server に保存されている Kerberos ハッシュが必要であるためです。このハッシュを生成するには、IdM サーバー側でユーザーのパスワードがクリアテキストで利用可能である必要があります。新しいユーザーパスワードを作成すると、パスワードはハッシュされて IdM に保存される前に、クリアテキストで利用できるようになります。ただし、ユーザーを LDAP ディレクトリーから移行する場合には関連するユーザーパスワードがすでにハッシュ化されているため該当する Kerberos キーは生成できません。

重要

デフォルトでは、ユーザーアカウントが存在しても、Kerberos ハッシュが発生するまで、ユーザーは IdM ドメインに認証したり、IdM リソースにアクセスしたりできません。回避策の 1 つが利用できます。Kerberos 認証の代わりに、IdM で LDAP 認証を使用します。この回避策では、ユーザーに Kerberos ハッシュは必要ありません。ただし、この回避策により IdM の機能が制限されるため、推奨できません。

次のセクションでは、ユーザーとそのパスワードを移行する方法を説明します。

- Methods for migrating passwords when migrating LDAP to IdM
  - Web ページの使用
  - SSSD の使用
- Planning the migration of cleartext LDAP passwords
- Planning the migration of LDAP passwords that do not meet the IdM requirements

4.3.1. Methods for migrating passwords when migrating LDAP to IdM

ユーザーにパスワードの変更を強制せずに、ユーザーアカウントを LDAP から Identity Management (IdM) に移行するには、以下の方法を使用できます。

方法 1: 移行 Web ページの使用

ユーザーに、IdM Web UI (**https://ipaserver.example.com/ipa/migration**) のスペシャルページに LDAP 認証情報を一度入力するように指示します。バックグラウンドで実行しているスクリプトが、クリアテキストパスワードをキャプチャーして、パスワードと適切な Kerberos ハッシュを使用してユーザーアカウントを適切に更新します。

方法 2 (推奨) - SSSD の使用

SSSD (System Security Services Daemon) を使用して必要なユーザーキーを生成することで、移行によるユーザーへの影響を軽減します。大量のユーザーを導入する場合やユーザーにパスワード変更の面倒をかけさせない場合に最適なシナリオです。

ワークフロー

1. ユーザーが SSSD でマシンにログインします。
2. SSSD は、IdM サーバーに対して Kerberos 認証の実行を試みます。
3. ユーザーがシステムに存在していも Kerberos ハッシュがないため **key type is not supported** エラーで認証に失敗します。
4. SSSD は、セキュアな接続でプレーンテキストの LDAP バインドを実行します。
5. IdM はこのバインド要求をインターセプトします。ユーザーが Kerberos プリンシパルは持っているのに Kerberos ハッシュを持っていない場合、IdM ID プロバイダーはハッシュを生成してユーザーのエンティティーに格納します。
6. 認証に成功すると SSSD は IdM との接続を切断し Kerberos 認証を再試行します。この場合、エンティティーにハッシュが存在しているため要求は成功します。

方法 2 では、ユーザーにはプロセス全体が表示されません。ユーザーは、パスワードが LDAP から IdM に移動したことに気が付かずにクライアントサービスにログインします。

4.3.2. Planning the migration of cleartext LDAP passwords

ほとんどのデプロイメントでは暗号化された LDAP パスワードが格納されますが、ユーザーまたは環境によってユーザーエンティティーにクリアテキストのパスワードが使用される場合があります。

ユーザーを LDAP サーバーから IdM サーバーに移行する場合、IdM ではクリアテキストパスワードが許可されていないため、クリアテキストパスワードは移行されません。代わりに、Kerberos プリンシパルがユーザーごとに作成され、キータブは true に設定されます。また、パスワードは期限が切れたときに設定されます。つまり、IdM では、次回のログイン時にユーザーがパスワードをリセットする必要があります。詳細は、Planning the migration of LDAP passwords that do not meet the IdM requirements を参照してください。

4.3.3. Planning the migration of LDAP passwords that do not meet the IdM requirements

元のディレクトリーのユーザーパスワードが、Identity Management (IdM) で定義されているパスワードポリシーに一致しない場合、そのパスワードは移行後に無効になります。

パスワードのリセットは、ユーザーが **kinit** と入力して IdM ドメイン内の Kerberos チケット付与チケット (TGT) を最初に取得しようとするときに自動的に行われます。ユーザーはパスワードの変更を強制されます。

```
[migrated_idm_user@idmclient ~]$ kinit
Password for migrated_idm_user@IDM.EXAMPLE.COM:
Password expired.  You must change it now.
Enter new password:
Enter it again:
```

4.4. 移行における考慮事項と要件

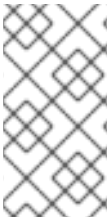
LDAP サーバーから Identity Management (IdM) への移行を計画している場合は、LDAP 環境が IdM の移行スクリプトで機能できることを確認してください。

4.4.1. 移行に対応している LDAP サーバー

LDAP サーバーから Identity Management への移行プロセスは、特別なスクリプト **ipa migrate-ds** を使用して移行を実行します。このスクリプトには、LDAP ディレクトリーと LDAP エントリーの構造に関する特定の要件があります。移行に対応しているのは複数の共通ディレクトリーを含む LDAPv3 準拠のディレクトリーサービスのみになります。

- Sun ONE Directory Server
- Apache Directory Server
- OpenLDAP

LDAP サーバーから IdM への移行は、Red Hat Directory Server および OpenLDAP でテストされています。



注記

Microsoft Active Directory の場合、移行用スクリプトを使用した移行には**対応していません**。これは、LDAPv3-コンプライアントディレクトリーではないためです。Active Directory からの移行については、Red Hat Professional Services にお問い合わせください。

4.4.2. 移行のための LDAP 環境要件

LDAP サーバーと Identity Management (IdM) にはさまざまな設定シナリオが存在し、これが移行プロセスの円滑さに影響します。移行手順の例では、環境に関する前提条件を以下に示します。

- 1 つの LDAP ディレクトリードメインが、1 つの IdM レルムに移行中です。統合はありません。
- ユーザーパスワードは、LDAP ディレクトリーにハッシュ形式で保存されます。対応しているハッシュのリストは、Red Hat Directory Server Documentation の Red Hat Directory Server 10 で利用可能な **Configuration, Command, and File Reference** タイトルの Password Storage Schemes セクションを参照してください。
- LDAP ディレクトリーインスタンスは ID 格納および認証方法の両方になります。クライアントマシンは、**pam_ldap** または **nss_ldap** を使用して LDAP サーバーに接続するように設定されます。
- エントリーは標準の LDAP スキーマのみを使用します。カスタムオブジェクトクラスまたはカスタム属性を含むエントリーは、IdM に移行されません。
- **migrate-ds** は、以下のアカウントのみを移行します。
  - **gidNumber** 属性を含むもの。この属性は、**posixAccount** オブジェクトクラスに必要です。
  - **sn** 属性を含むもの。この属性は、**person** オブジェクトクラスに必要です。

4.4.3. 移行のための IdM システム要件

中程度のサイズのディレクトリー (約 10,000 ユーザー、および 10 グループ) では、移行を続行するのに十分な強力なターゲット IdM システムが必要です。移行の最小要件は以下のとおりです。

- 4 コア
- 4 GB のメモリー
- 30GB のディスク領域
- 2MB の SASL バッファーサイズ。これは IdM サーバーのデフォルトです。
  移行エラーが発生した場合は、バッファーサイズを大きくします。

```
[root@ipaserver ~]# ldapmodify -x -D 'cn=directory manager' -w password -h
ipaserver.example.com -p 389

dn: cn=config
changetype: modify
replace: nsslapd-sasl-max-buffer-size
nsslapd-sasl-max-buffer-size: 4194304

modifying entry "cn=config"
```

nsslapd-sasl-max-buffer-size をバイト単位で設定します。

関連情報

- IdM server hardware recommendations

4.4.4. ユーザーおよびグループ ID 番号

LDAP から IdM デプロイメントに移行する場合は、デプロイメント間でユーザー ID (UID) とグループ ID (GID) の競合が存在しないことを確認してください。移行前に、以下を確認します。

- LDAP ID 範囲を把握している。
- IdM ID 範囲を把握している。
- LDAP サーバーの UID と GID と、RHEL システムまたは IdM デプロイメント上の既存の UID または GID の間で重複は存在しません。
- 移行された LDAP UID および GID は、IdM ID 範囲に適合します。
  - 必要に応じて、移行前に新しい IdM ID 範囲を作成します。

関連情報

- 新しい IdM ID 範囲の追加

4.4.5. sudo ルールに関する考慮事項

LDAP で **sudo** を使用している場合は、LDAP に保存されている **sudo** ルールを Identity Management (IdM) に手動で移行する必要があります。Red Hat では、IdM のネットグループをホストグループとして再作成することを推奨しています。IdM は、SSSD **sudo** プロバイダーを使用しない **sudo** 設定で、従来のネットグループとしてホストグループを自動的に表示します。

4.4.6. LDAP から IdM への移行ツール

LDAP ディレクトリーのデータが正しくフォーマット化され、IdM サーバーに適切にインポートされるように、Identity Management (IdM) は特定の **ipa migrate-ds** コマンドを使用して移行プロセスを進めます。**ipa migrate-ds** を使用する場合は、**--bind-dn** オプションで指定するリモートシステムユーザーに、**userPassword** 属性への読み取りアクセスが必要です。読み取りアクセスがないと、パスワードが移行されません。

Identity Management サーバーは移行モードで実行するように設定してから、移行スクリプトを使用することができます。詳しくは、Migrating an LDAP server to IdM を参照してください。

4.4.7. LDAP から IdM への移行パフォーマンスの改善

LDAP の移行は、基本的には、IdM サーバー内の 389 Directory Server (DS) インスタンスに対する特殊なインポート操作です。インポート操作のパフォーマンスを向上させるために、389 DS インスタンスをチューニングすると、移行パフォーマンス全体を改善できます。

インポートのパフォーマンスに直接影響を与えるパラメーターは、以下の 2 つです。

- **nsslapd-cachememsize** 属性。エントリーキャッシュに使用できるサイズを定義します。これは、キャッシュメモリーの合計サイズの 80% に自動的に設定されるバッファーです。大規模なインポート操作の場合は、このパラメーターを増やし、メモリーキャッシュ自体も増やすことができます。これにより、多数のエントリーや、大きな属性を持つエントリーを処理する際に、ディレクトリーサービスの効率が向上します。
  dsconf コマンドを使用して属性を変更する方法の詳細は、Adjusting the entry cache size を参照してください。
- システム **ulimit** 設定オプションは、システムユーザーに許可されるプロセスの最大数を設定します。大規模なデータベースの処理が制限を超える可能性があります。これが発生した場合は、値を上げます。

```
[root@server ~]# ulimit -u 4096
```

関連情報

- Adjusting IdM Directory Server performance

4.4.8. LDAP から IdM への移行シーケンス

IdM に移行する場合は、主に 4 つの手順がありますが、順序は、**サーバー** と**クライアント** のどちらを最初に移行するかによって異なります。

重要

クライアントファーストおよびサーバーファーストの両方の移行では、一般的な移行手順が提供されますが、すべての環境で機能するとは限りません。実際の LDAP 環境を移行する前に、テスト用の LDAP 環境を設定して移行プロセスの検証を行ってください。

クライアントファースト移行

SSSD は、Identity Management (IdM) サーバーが設定される際に、クライアント設定を変更するために使用されます。

1. SSSD をディプロイします。
2. クライアントが現在の LDAP サーバーに接続し IdM にフェイルオーバーするよう再設定を行います。
3. IdM サーバーをインストールします。
4. IdM **ipa migrate-ds** スクリプトを使用してユーザーデータを移行します。これによりデータが LDAP ディレクトリーからエクスポートされ、IdM スキーマ用にフォーマット化されて IdM にインポートされます。
5. LDAP サーバーをオフラインにし、クライアントが IdM に透過的にフェイルオーバーできるようにします。

サーバーファースト移行

LDAP から IdM への移行が最初に行われます。

1. IdM サーバーをインストールします。
2. IdM **ipa migrate-ds** スクリプトを使用してユーザーデータを移行します。これによりデータが LDAP ディレクトリーからエクスポートされ、IdM スキーマ用にフォーマット化されて IdM にインポートされます。
3. **オプション:**SSSD をディプロイします。
4. クライアントが IdM に接続するよう再設定を行います。LDAP サーバーと単純に差し替えることはできません。IdM ディレクトリーツリー (およびユーザーエントリーの DN) は以前のディレクトリーツリーとは異なります。
   クライアントの再設定は必要ですが、直ちに再設定を行う必要はありません。更新したクライアントは IdM サーバーをポイントし、他のクライアントは旧 LDAP ディレクトリーをポイントするためデータ移植後に適度なテストと移行段階を持たせることができます。

注記

LDAP ディレクトリーと IdM サーバーを長期に渡っては並行稼動させないでください。2 つのサービス間でユーザーデータの整合性が失われる危険を招くことになります。

4.5. LDAP から IDM への移行のカスタマイズ

ipa migrate-ds コマンドを使用して、LDAP サーバーから Identity Management (IdM) に認証サービスと認可サービスを移行できます。一番単純な例では移行するディレクトリーの LDAP URL を取得し、共通デフォルト設定をもとにデータをエクスポートします。

別の **ipa migrate-ds** コマンドオプションを使用すると、移行プロセスをカスタマイズし、データの識別およびエクスポート方法をカスタマイズできます。LDAP ディレクトリーツリーに一意の構造がある場合、またはエントリー内の特定のエントリーまたは属性を除外する必要がある場合は、移行をカスタマイズします。

4.5.1. LDAP から IdM への移行時にバインド DN およびベース DN をカスタマイズする例

ipa migrate-ds コマンドを使用して、LDAP から Identity Management (IdM) に移行します。一番単純な例では移行するディレクトリーの LDAP URL を取得し、共通デフォルト設定をもとにデータをエクスポートします。以下に、デフォルト設定を変更する例を示します。

```
# ipa migrate-ds ldap://ldap.example.com:389
```

バインド DN のカスタマイズ

デフォルトでは、DN "**cn=Directory Manager**" は、リモート LDAP ディレクトリーにバインドするために使用されます。**--bind-dn** オプションを使用して、カスタムバインド DN を指定します。

```
# ipa migrate-ds ldap://ldap.example.com:389 --bind-dn=cn=Manager,dc=example,dc=com
```

命名コンテキストのカスタマイズ

LDAP サーバーの命名コンテキストが IdM で使用されているものと異なる場合は、オブジェクトのベース DN が変換されます。たとえば、**uid=user,ou=people,dc=ldap,dc=example,dc=com** は、**uid=user,ou=people,dc=idm,dc=example,dc=com** に移行されます。**--base-dn** を使用して、コンテナーサブツリーのターゲットを変更し、移行にリモート LDAP サーバーで使用するベース DN を設定できます。

```
# ipa migrate-ds --base-dn="ou=people,dc=example,dc=com" ldap://ldap.example.com:389
```

関連情報

- **ipa migrate-ds --help**

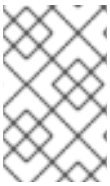
4.5.2. 特定のサブツリーの移行

デフォルトのディレクトリー構造の場合、人のエントリーは **ou=People** サブツリーに配置され、グループのエントリーは **ou=Groups** サブツリーに配置されます。こうしたサブツリーは異なるタイプのディレクトリーデータ用のコンテナーエントリーになります。**migrate-ds** コマンドでオプションが渡

されていない場合、ユーティリティーは、指定の LDAP ディレクトリーが **ou=People** および **ou=Groups** 構造を使用していることを前提とします。

多くのデプロイメントは完全に異なるディレクトリー構造をしている場合があります。または、元のディレクトリーツリーの特定部分のみをエクスポートする場合もあります。管理者は、以下のオプションを使用して、ソース LDAP サーバーの別のユーザーまたはグループのサブツリーの RDN を指定できます。

- **--user-container**
- **--group-container**



注記

いずれの場合も、サブツリーは相対識別名 (RDN) でなければならず、ベース DN に相対的である必要があります。たとえば、**--user-container=ou=Employees** を使用して、**>ou=Employees,dc=example,dc=com** ディレクトリーツリーを移行できます。

以下に例を示します。

```
[ipaserver ~]# ipa migrate-ds --user-container=ou=employees \
--group-container="ou=employee groups" ldap://ldap.example.com:389
```

必要に応じて、**--scope** オプションを **ipa migrate-ds** コマンドに追加して、スコープを設定します。

- **onelevel**: デフォルト。指定したコンテナーのエントリーのみが移行されます。
- **subtree**: 指定したコンテナーおよびすべてのサブコンテナーのエントリーが移行されます。
- **base**: 指定されたオブジェクト自体のみが移行されます。

4.5.3. エントリーの追加と除外

デフォルトでは、**ipa migrate-ds** スクリプトは、**person** オブジェクトクラスを持つすべてのユーザーエントリーと、**groupOfUniqueNames** オブジェクトクラスまたは **groupOfNames** オブジェクトクラスを持つすべてのグループエントリーをインポートします。

一部の移行パスでは、特定のタイプのユーザーとグループのみをエクスポートする必要がある場合や、あるいは特定のユーザーとグループを除外する必要がある場合があります。ユーザーまたはグループのエントリーを検索する際に、検索するオブジェクトクラスを設定することで、含めるユーザーとグループの**タイプ**を選択できます。

このオプションは、さまざまな**ユーザー**タイプにカスタムオブジェクトクラスを使用する場合にとりわけ役立ちます。たとえば、次のコマンドは、カスタム **fullTimeEmployee** オブジェクトクラスを持つユーザーのみを移行します。

```
[root@ipaserver ~]# ipa migrate-ds --user-objectclass=fullTimeEmployee
ldap://ldap.example.com:389
```

グループの種類が異なるため、これは、ユーザーグループなどの特定の種類の**グループ**のみを移行し、証明書グループなどの他の種類のグループを除外する場合にも非常に役立ちます。以下に例を示します。

```
[root@ipaserver ~]# ipa migrate-ds --group-objectclass=groupOfNames --group-
objectclass=groupOfUniqueNames ldap://ldap.example.com:389
```

オブジェクトクラスに基づいて移行するユーザーとグループのエントリーを指定すると、他のすべてのユーザーとグループが移行から暗黙的に除外されます。

また、ごく少数のエントリー以外、すべてのユーザーとグループのエントリーを移行する場合にも便利です。そのタイプの他のすべてを移行するときに、特定のユーザーまたはグループアカウントを除外できます。たとえば、これは趣味のグループと 2 人のユーザーのみを除外します。

```
[root@ipaserver ~]# ipa migrate-ds --exclude-groups="Golfers Group" --exclude-
users=idmuser101 --exclude-users=idmuser102 ldap://ldap.example.com:389
```

exclude ステートメントは、**uid** でパターンに一致するユーザーと、**cn** 属性でパターンに一致するグループに適用されます。

一般的なオブジェクトクラスを移行できますが、そのクラスの特定のエントリーは除外できます。たとえば、これには特に **fullTimeEmployee** オブジェクトクラスを持つユーザーが含まれますが、3 つのマネージャーは除外されます。

```
[root@ipaserver ~]# ipa migrate-ds --user-objectclass=fullTimeEmployee --exclude-
users=jsmith --exclude-users=bjensen --exclude-users=mreynolds
ldap://ldap.example.com:389
```

4.5.4. エントリー属性の除外

デフォルトではユーザーやグループエントリーのすべての属性とオブジェクトクラスが移行されます。特定のシナリオでは、帯域幅とネットワークの制約のため、または属性データが関連しなくなったために、これは現実的ではない場合があります。たとえば、ユーザーが Identity Management (IdM) ドメインに参加する際に新しいユーザー証明書が割り当てられる場合は、**userCertificate** 属性を移行しても意味がありません。

migrate-ds コマンドで以下のオプションを使用すると、特定のオブジェクトクラスおよび属性を無視できます。

- **--user-ignore-objectclass**
- **--user-ignore-attribute**
- **--group-ignore-objectclass**
- **--group-ignore-attribute**

たとえば、ユーザーの **userCertificate** 属性および **strongAuthenticationUser** オブジェクトクラスとグループの **groupOfCertificates** オブジェクトクラスを除外するには、次のコマンドを実行します。

```
[root@ipaserver ~]# ipa migrate-ds --user-ignore-attribute=userCertificate --user-ignore-
objectclass=strongAuthenticationUser --group-ignore-objectclass=groupOfCertificates
ldap://ldap.example.com:389
```

注記

必要な属性が無視されていないか必ず確認します。また、オブジェクトクラスを除外する場合は、そのオブジェクトクラスのみがサポートする属性を必ず除外してください。

関連情報

- 移行のための LDAP 環境要件

4.5.5. LDAP から IdM への移行時に使用するスキーマとスキーマ互換機能

Identity Management (IdM) は、RFC2307bis スキーマを使用して、ユーザー、ホスト、ホストグループ、およびその他のネットワーク ID を定義します。ただし、移行のソースとして使用する LDAP サーバーが、代わりに RFC2307 スキーマを使用する場合は、**ipa migrate-ds** コマンドで **--schema** オプションを指定します。

```
[root@ipaserver ~]# ipa migrate-ds --schema=RFC2307 ldap://ldap.example.com:389
```

また、IdM には組み込みの**スキーマの互換機能** があるため、RFC2307bis に対応していないシステムのデータを IdM が再フォーマットできます。compat プラグインはデフォルトで有効になっています。つまり、ディレクトリーサーバーは、ユーザーとグループの代替ビューを計算し、**cn=users,cn=compat,dc=example,dc=com** コンテナーエントリーにこのビューを提供します。これは、システムの起動時にエントリーの内容を事前に計算して、必要に応じてエントリーを更新することで行われます。

システムのオーバーヘッドを低減するために、この機能は移行中に無効にすることが推奨されます。

4.6. LDAP サーバーの IDM への移行

ipa migrate-ds コマンドを使用して、LDAP サーバーから Identity Management (IdM) に認証サービスと認可サービスを移行できます。

警告

この例は一般的な移行手順のため、あらゆる環境に対応するわけではありません。

実際に LDAP 環境の移行に入る前に、LDAP のテスト環境を設定して移行プロセスを検証することを強く推奨します。環境をテストする場合は、以下を行います。

1. IdM でテストユーザーを作成し、移行したユーザーの出力を、テストユーザーの出力と比較します。
2. IdM にあるように、移行したユーザーの出力を、元の LDAP サーバーにあるように、ソースユーザーと比較します。

詳細なガイダンスは、以下の **検証** のセクションを参照してください。

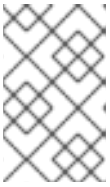
前提条件

- LDAP ディレクトリーに対する管理者特権がある。
- IdM がインストールされている場合は、IdM の管理者特権がある。
- 以下の手順を実行している RHEL システムに **root** としてログインしている。

- 以下の章を読み、理解している。
  - Considerations in migrating from LDAP to IdM .
  - Planning the client configuration when migrating from LDAP to IdM .
  - Planning password migration when migrating from LDAP to IdM .
  - Further migration considerations and requirements.
  - Customizing the migration from LDAP to IdM .

手順

1. IdM がインストールされていない場合: 既存の LDAP ディレクトリーがインストールされているマシンとは別のマシンに、IdM サーバー (カスタム LDAP ディレクトリースキーマを含む) をインストールします。詳細は、Identity Management のインストール を参照してください。



注記

カスタムユーザースキーマまたはカスタムグループスキーマの IdM でのサポートは限られています。互換性のないオブジェクト定義があると、移行中に問題が発生する可能性があります。

2. パフォーマンスの理由から、互換性プラグインを無効にします。

```
# ipa-compat-manage disable
```

スキーマ互換性機能の詳細と、移行時にスキーマ互換性機能を無効にする利点は、The schema to use when migrating from LDAP to IdM and the schema compat feature を参照してください。

3. IdM Directory Server インスタンスを再起動します。

```
# systemctl restart dirsrv.target
```

4. IdM サーバーが移行を許可できるように設定します。

```
# ipa config-mod --enable-migration=TRUE
```

--enable-migration を TRUE に設定すると、以下のようになります。

- LDAP の追加操作時に、ハッシュ前のパスワードを許可します。
- 初期 Kerberos 認証に失敗した場合に、パスワードの移行シーケンスを試行するように SSSD を設定します。詳細は、Using SSSD when migrating passwords from LDAP to IdM の Workflow セクションを参照してください。

5. ユースケースに応じたオプションを指定して、IdM 移行スクリプト **ipa migrate-ds** を実行します。詳細は、Customizing the migration from LDAP to IdM を参照してください。

```
# ipa migrate-ds --your-options ldap://ldap.example.com:389
```

注記

上記のいずれかの手順で compat プラグインを無効にしなかった場合は、**--with-compat** オプションを **ipa migrate-ds** に追加します。

```
# ipa migrate-ds --your-options --with-compat
ldap://ldap.example.com:389
```

6. 互換性プラグインを再度有効にします。

```
# ipa-compat-manage enable
```

7. IdM Directory Server を再起動します。

```
# systemctl restart dirsrv.target
```

8. すべてのユーザーのパスワードが移行したら、移行モードを無効にします。

```
# ipa config-mod --enable-migration=FALSE
```

9. [オプション] すべてのユーザーが移行されたら、非 SSSD クライアントを再設定して、LDAP 認証 (**pam_ldap**) ではなく Kerberos 認証 (**pam_krb5**) を使用します。詳細は、RHEL 7 **System-level Authentication Guide** の Configuring a Kerberos Client を参照してください。

10. ユーザーにハッシュされた Kerberos パスワードを生成させます。Planning password migration when migrating from LDAP to IdM で説明されている方法のいずれかを選択します。

- SSSD メソッド を決定した場合は、以下を行います。
  - SSSD がインストールされているクライアントを、LDAP ディレクトリーから IdM ディレクトリーに移動し、IdM でクライアントとして登録します。これにより必要なキーと証明書がダウンロードされます。
    Red Hat Enterprise Linux クライアントでは、この **ipa-client-install** コマンドを使用して実行できます。以下に例を示します。

```
# ipa-client-install --enable-dns-update
```

- IdM 移行 Web ページ メソッドを決定した場合は、以下を行います。
  - 移行 Web ページを使用して IdM にログインするようにユーザーに指示します。

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

11. ユーザーの移行プロセスを監視するには、パスワードは持っているが Kerberos プリンシパルキーはまだないユーザーアカウントを表示するよう既存の LDAP ディレクトリーに問い合わせます。

```
$ ldapsearch -LL -x -D 'cn=Directory Manager' -w secret -b
'cn=users,cn=accounts,dc=example,dc=com' '(&(!(krbprincipalkey=))(userpassword=))'
uid
```

注記

フィルターの前後に一重引用符を付けてシェルで解釈されないようにします。

12. クライアントとユーザーすべての移行が完了したら LDAP ディレクトリーを廃止します。

検証

1. **ipa user-add** を使用して、IdM にテストユーザーを作成します。移行したユーザーの出力を、テストユーザーの出力と比較します。移行したユーザーに、テストユーザーに存在する属性およびオブジェクトクラスの最小セットが含まれていることを確認します。以下に例を示します。

```
$ ipa user-show --all testing_user
dn: uid=testing_user,cn=users,cn=accounts,dc=idm,dc=example,dc=com
User login: testing_user
First name: testing
Last name: user
Full name: testing user
Display name: testing user
Initials: tu
Home directory: /home/testing_user
GECOS: testing user
Login shell: /bin/sh
Principal name: testing_user@IDM.EXAMPLE.COM
Principal alias: testing_user@IDM.EXAMPLE.COM
Email address: testing_user@idm.example.com
UID: 1689700012
GID: 1689700012
Account disabled: False
Preserved user: False
Password: False
Member of groups: ipausers
Kerberos keys available: False
ipauniqueid: 843b1ac8-6e38-11ec-8dfe-5254005aad3e
mepmanagedentry: cn=testing_user,cn=groups,cn=accounts,dc=idm,dc=example,dc=com
objectclass: top, person, organizationalperson, inetorgperson, inetuser, posixaccount,
krbprincipalaux, krbticketpolicyaux, ipaobject,
        ipasshuser, ipaSshGroupOfPubKeys, mepOriginEntry
```

2. IdM にあるように、移行したユーザーの出力を、元の LDAP サーバーにあるように、ソースユーザーと比較します。インポートされた属性が 2 回コピーされていないこと、およびそれらが正しい値を持っていることを確認してください。

関連情報

- Migrating from LDAP to IdM over SSL

4.7. MIGRATING FROM LDAP TO IDM OVER SSL

ipa migrate-ds コマンドを使用して、LDAP サーバーから Identity Management (IdM) に認証サービスと認可サービスを移行できます。移行中に送信されるデータを暗号化するには、次の手順に従います。

警告

この例は一般的な移行手順のため、あらゆる環境に対応するわけではありません。

実際に LDAP 環境の移行に入る前に、LDAP のテスト環境を設定して移行プロセスを検証することを強く推奨します。環境をテストする場合は、以下を行います。

1. IdM でテストユーザーを作成し、移行したユーザーの出力を、テストユーザーの出力と比較します。
2. IdM にあるように、移行したユーザーの出力を、元の LDAP サーバーにあるように、ソースユーザーと比較します。

詳細なガイダンスは、以下の **検証** のセクションを参照してください。

前提条件

- LDAP ディレクトリーに対する管理者特権がある。
- IdM がインストールされている場合は、IdM の管理者特権がある。
- 以下の手順を実行している RHEL システムに **root** としてログインしている。
- 以下の章を読み、理解している。
  - Considerations in migrating from LDAP to IdM .
  - Planning the client configuration when migrating from LDAP to IdM .
  - Planning password migration when migrating from LDAP to IdM .
  - Further migration considerations and requirements.
  - Customizing the migration from LDAP to IdM .

手順

1. リモート LDAP サーバー証明書を発行した CA の証明書を、今後使用する IdM サーバーのファイルに保存します。たとえば、**/tmp/remote.crt** です。
2. Migrating an LDAP server to IdM に記載されている手順に従います。ただし、移行時に暗号化された LDAP 接続の場合、URL で **ldaps** プロトコルを使用し、**ipa migrate-ds** コマンドに **--ca-cert-file** オプションを渡します。以下に例を示します。

```
# ipa migrate-ds --ca-cert-file=/tmp/remote.crt --your-other-options
ldaps://ldap.example.com:636
```

検証

1. **ipa user-add** を使用して、IdM にテストユーザーを作成します。移行したユーザーの出力を、テストユーザーの出力と比較します。移行したユーザーに、テストユーザーに存在する属性およびオブジェクトクラスの最小セットが含まれていることを確認します。以下に例を示しま

す。

```
$ ipa user-show --all testing_user
dn: uid=testing_user,cn=users,cn=accounts,dc=idm,dc=example,dc=com
User login: testing_user
First name: testing
Last name: user
Full name: testing user
Display name: testing user
Initials: tu
Home directory: /home/testing_user
GECOS: testing user
Login shell: /bin/sh
Principal name: testing_user@IDM.EXAMPLE.COM
Principal alias: testing_user@IDM.EXAMPLE.COM
Email address: testing_user@idm.example.com
UID: 1689700012
GID: 1689700012
Account disabled: False
Preserved user: False
Password: False
Member of groups: ipausers
Kerberos keys available: False
ipauniqueid: 843b1ac8-6e38-11ec-8dfe-5254005aad3e
mepmanagedentry: cn=testing_user,cn=groups,cn=accounts,dc=idm,dc=example,dc=com
objectclass: top, person, organizationalperson, inetorgperson, inetuser, posixaccount,
krbprincipalaux, krbticketpolicyaux, ipaobject,
        ipasshuser, ipaSshGroupOfPubKeys, mepOriginEntry
```

2. IdM にあるように、移行したユーザーの出力を、元の LDAP サーバーにあるように、ソースユーザーと比較します。インポートされた属性が 2 回コピーされていないこと、およびそれらが正しい値を持っていることを確認してください。

パート III. LINUX ドメインと ACTIVE DIRECTORY ドメインを統合する際の同期から信頼への既存環境の移行

RHEL 7 では、**同期** と **信頼** は、RHEL システムを Active Directory (AD) へ間接的に統合する場合に考えられる 2 つの方法でした。RHEL 8 では、同期は非推奨になりました。RHEL 7 で RHEL Identity Management (IdM) と AD 間の 同期を設定 している場合、IdM-AD 信頼に基づくアプローチに移行することを推奨します。

本章では、既存の同期ベースの設定を AD 信頼に移行する方法を説明しています。以下の移行オプションは IdM で利用可能です。

- ipa-winsync-migrate を使用した、同期から信頼への自動移行
- ID ビューを使用した、同期から信頼への手動移行

第5章 IPA-WINSYNC-MIGRATE を使用した、同期から信頼への自動移行

RHEL 8 では、RHEL システムを Active Directory (AD) に間接的に統合する同期アプローチは非推奨になりました。Red Hat では、代わりに Identity Management (IdM) と AD の間の信頼に基づくアプローチに移行することを推奨しています。この章では、**ipa-winsync-migrate** ユーティリティーを使用して、同期から信頼に自動的に移行する方法について説明します。

5.1. IPA-WINSYNC-MIGRATE を使用した、同期から信頼への自動移行

ipa-winsync-migrate ユーティリティーは、Winsync 環境の既存の設定を保持し、それを AD 信頼に転送しながら、同期されたすべてのユーザーを AD フォレストから移行します。Winsync 合意で作成された各 AD ユーザーには、**ipa-winsync-migrate** がデフォルト信頼ビュー内に ID オーバーライドを作成します。

移行完了後には、以下のようになります。

- AD ユーザーの ID 上書きには、Winsync 内の元のエントリーから以下の属性がコピーされます。
  - ログイン名 (**uid**)
  - UID 番号 (**uidnumber**)
  - GID 番号 (**gidnumber**)
  - ホームディレクトリー (**homedirectory**)
  - GECOS エントリー (**gecos**)
- AD 信頼内のユーザーアカウントは、以下を含む IdM 内の元の設定を保持します。
  - POSIX 属性
  - ユーザーグループ
  - ロールベースのアクセス制御ルール
  - ホストベースのアクセス制御ルール
  - SELinux メンバーシップ
  - **sudo** ルール
- 新規 AD ユーザーが外部 IdM グループのメンバーとして追加されます。
- 元の Winsync レプリケーション合意、元の同期済みユーザーアカウント、およびユーザーアカウントのローカルコピーがすべて削除されます。

関連情報

- デフォルト信頼ビューの仕組み

5.2. IPA-WINSYNC-MIGRATE を使用した、同期から信頼への移行

前提条件

- RHEL 7 で、RHEL Identity Management (IdM) と AD の間の 同期を設定 している。

手順

1. **ipa-backup** ユーティリティーを使用して IdM 設定をバックアップします。IdM のバックアップおよび復元 を参照してください。

   注記

   移行は、IdM 設定および多くのユーザーアカウントに多大な影響を及ぼします。バックアップを作成することで、必要な場合は元の設定を復元することができます。

2. 同期されたドメインで信頼を作成します。詳細は、IdM と AD との間の信頼のインストール を参照してください。

3. **ipa-winsync-migrate** を実行して、AD レルムと、AD ドメインコントローラーのホスト名を指定します。

   ```
   # ipa-winsync-migrate --realm example.com --server ad.example.com
   ```

   ipa-winsync-migrate が作成したオーバーライド内で競合が発生した場合は、この競合についての情報が表示されますが、移行は継続されます。

4. AD サーバーからのパスワード同期サービスをアンインストールします。これにより、AD ドメインコントローラーから同期合意が削除されます。

関連情報

- ipa-winsync-migrate(1)
- Ansible Playbook を使用した IdM サーバーのバックアップおよび復元

第6章 ID ビューを使用した、同期から信頼への手動移行

RHEL 8 では、RHEL システムを Active Directory (AD) に間接的に統合する同期アプローチは非推奨になりました。Red Hat では、代わりに Identity Management (IdM) と AD の間の信頼に基づくアプローチに移行することを推奨しています。この章では、ID ビューを使用して同期から信頼に手動で移行する方法について説明します。

6.1. ID ビューを使用した、同期から信頼への手動移行

ID ビューを使用すると、Active Directory (AD) が以前に AD ユーザー向けに生成した POSIX 属性を手動で変更できます。

前提条件

- RHEL 7 で、RHEL Identity Management (IdM) と AD の間の 同期を設定 している。

手順

1. 元の同期したユーザーおよびグループエントリーのバックアップを作成します。
2. 同期されたドメインで信頼を作成します。詳細は、IdM と AD との間の信頼のインストール を参照してください。
3. 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 を保持するために以下のいずれかを実行します。
   - 特定のホストに適用される ID ビューを個別に作成し、ユーザー ID 上書きをビューに追加する。
   - デフォルト信頼ビューでユーザー ID 上書きを作成する。

注記

ID ビューは、IdM ユーザーのみが管理できます。AD ユーザーはできません。

4. 元の同期したユーザーまたはグループのエントリーを削除します。

関連情報

- Active Directory ユーザーの ID ビューの使用
- 異なるホストのユーザーアカウントに対する異なる属性値の定義