Red Hat Enterprise Linux 9

Identity Management でのレプリケーションの管理

レプリケーション環境の準備および検証

Last Updated: 2024-05-03

Red Hat Enterprise Linux 9 Identity Management でのレプリケーションの管理

レプリケーション環境の準備および検証

概要

Red Hat Identity Management (IdM) 環境では、レプリケーションによりフェイルオーバーとロードバランシングが可能になります。コマンドライン、Web UI、および Ansible Playbook を使用して、サーバー間のレプリケーションを設定、検証、および停止できます。

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参ります。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。

第1章 レプリケーショントポロジーの管理

本章では、Identity Management(IdM) ドメイン内のサーバー間のレプリケーションを管理する方法を説明します。

関連情報

- レプリカトポロジーの計画

1.1. レプリカ合意、トポロジー接尾辞、およびトポロジーセグメントの説明

レプリカを作成すると、Identity Management (IdM) が初期サーバーとレプリカ間にレプリカ合意を作成します。複製されるデータはトポロジーの接尾辞に保存され、2 つのレプリカの接尾辞間でレプリカ合意があると、接尾辞がトポロジーセグメントを形成します。これらの概念は、以下のセクションで詳細に説明されています。

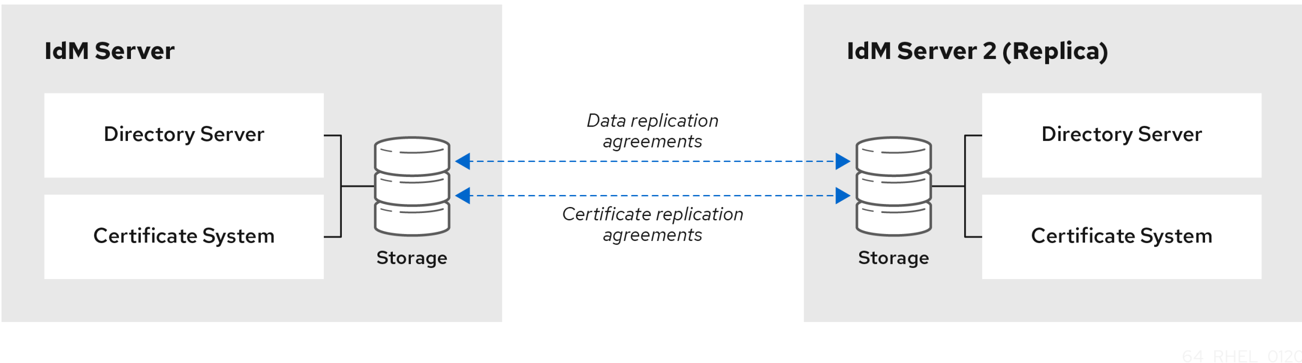
- レプリカ合意
- トポロジー接尾辞
- トポロジーセグメント

1.1.1. IdM レプリカ間のレプリカ合意

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

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

図1.1 サーバーとレプリカ合意



IdM は、2 種類のレプリカ合意を使用します。

ドメインのレプリカ合意

この合意は、識別情報を複製します。

証明書のレプリカ合意

この合意は、証明書情報を複製します。

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

1.1.2. トポロジー接尾辞

トポロジー接尾辞は、レプリケートされるデータを保存します。IdM は、**domain** と **ca** の 2 種類のトポロジー接尾辞に対応します。それぞれの接尾辞は、個別のサーバーである個別のレプリケーショントポロジーを表します。

レプリカ合意が設定されると、同じタイプのトポロジー接尾辞を 2 つの異なるサーバーに結合します。

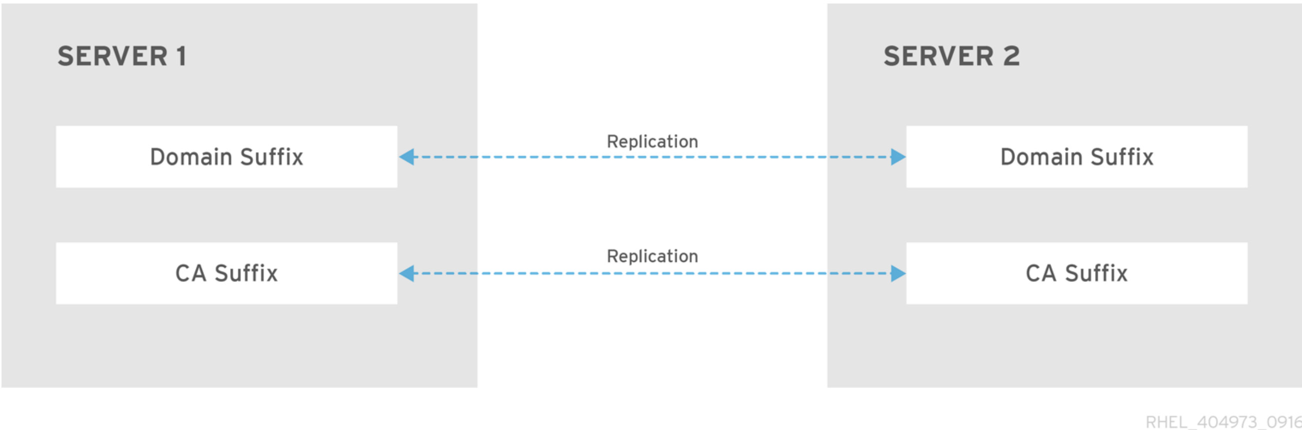
domain 接尾辞: dc=example,dc=com

: **domain** 接尾辞には、ドメイン関連のデータがすべて含まれています。
2 つのレプリカの **domain** 接尾辞間でレプリカ合意が設定されると、ユーザー、グループ、およびポリシーなどのディレクトリーデータが共有されます。

ca 接尾辞: o=ipaca

: **ca** 接尾辞には、Certificate System コンポーネントのデータが含まれます。これは認証局 (CA) がインストールされているサーバーにのみ存在します。
2 つのレプリカの **ca** 接尾辞間でレプリカ合意が設定されると、証明書データが共有されます。

図1.2 トポロジー接尾辞



新規レプリカのインストール時には、**ipa-replica-install** スクリプトが 2 つのサーバー間に初期トポロジーレプリカ合意をセットアップします。

例1.1 トポロジー接尾辞の表示

ipa topologysuffix-find コマンドでトポロジー接尾辞のリストが表示されます。

```
$ ipa topologysuffix-find
---------------------------
2 topology suffixes matched
---------------------------
  Suffix name: ca
  Managed LDAP suffix DN: o=ipaca

  Suffix name: domain
  Managed LDAP suffix DN: dc=example,dc=com
```

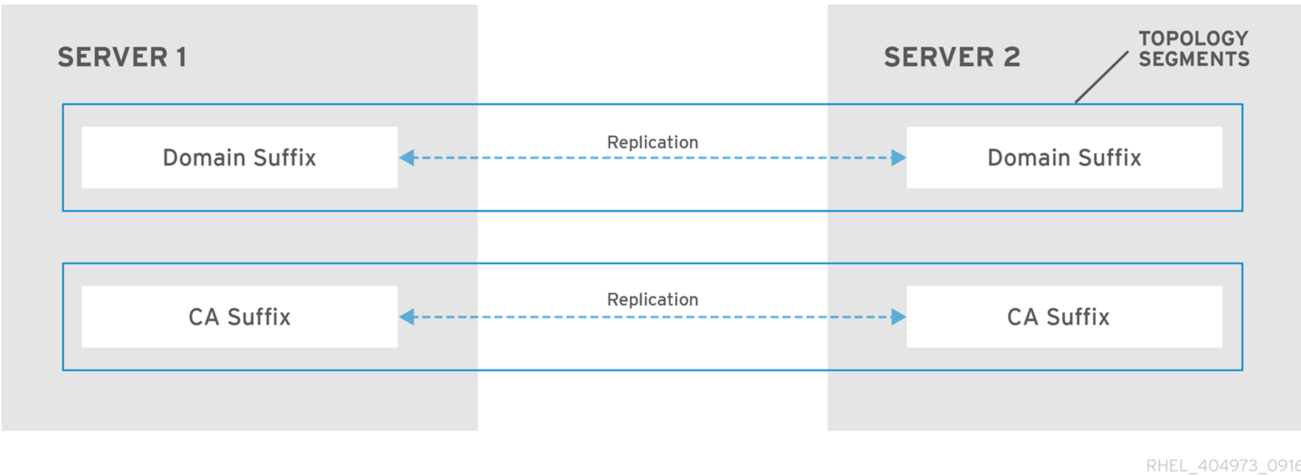
```
----------------------------
Number of entries returned 2
----------------------------
```

1.1.3. トポロジーセグメント

2 つのレプリカの接尾辞間でレプリカ合意があると、接尾辞は **トポロジーセグメント** を形成します。各トポロジーセグメントは、**左ノード**と**右ノード** で設定されます。ノードは、レプリカ合意に参加しているサーバーを表します。

IdM のトポロジーセグメントは常に双方向です。各セグメントは、サーバー A からサーバー B、およびサーバー B からサーバー A への 2 つのレプリカ合意を表します。そのため、データは両方の方向でレプリケートされます。

図1.3 トポロジーセグメント



例1.2 トポロジーセグメントの表示

ipa topologysegment-find コマンドで、ドメインまたは CA 接尾辞に設定されたトポロジーセグメントが表示されます。たとえば、ドメイン接尾辞の場合は、以下のようになります。

```
$ ipa topologysegment-find
Suffix name: domain
-----------------
1 segment matched
-----------------
  Segment name: server1.example.com-to-server2.example.com
  Left node: server1.example.com
  Right node: server2.example.com
  Connectivity: both
----------------------------
Number of entries returned 1
----------------------------
```

この例では、ドメイン関連のデータのみが **server1.example.com** と **server2.example.com** の 2 つのサーバー間で複製されます。

特定セグメントの詳細を表示するには、**ipa topologysegment-show** コマンドを使用します。

```
$ ipa topologysegment-show
```

```
Suffix name: domain
Segment name: server1.example.com-to-server2.example.com
  Segment name: server1.example.com-to-server2.example.com
  Left node: server1.example.com
  Right node: server2.example.com
  Connectivity: both
```

1.2. トポロジーグラフを使用したレプリケーショントポロジーの管理

Web UI のトポロジーグラフは、ドメイン内のサーバー間の関係を表示します。Web UI を使用すると、トポロジーの表現を操作および変換できます。

トポロジーグラフへのアクセス

トポロジーグラフにアクセスするには、以下を実行します。

1. **IPA Server → Topology → Topology Graph** を選択します。
2. トポロジーに加えた変更がグラフに反映されていない場合は、**Refresh** をクリックします。

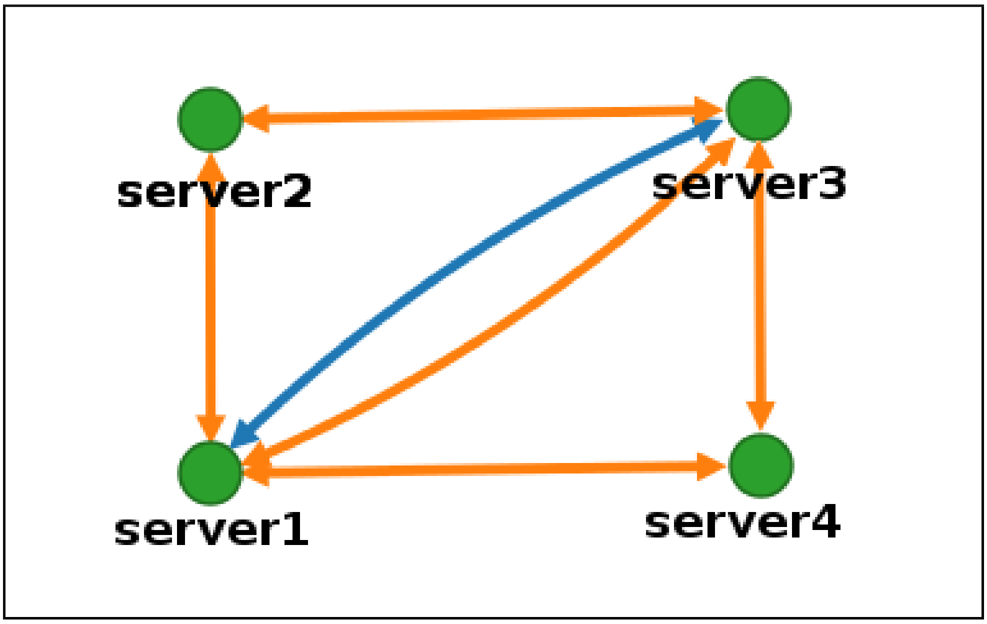
トポロジーグラフの解釈

ドメインのレプリカ合意に参加しているサーバーは、オレンジ色の矢印によって接続されます。CA のレプリカ合意に参加しているサーバーは、青色の矢印によって接続されます。

トポロジーグラフの例: 推奨されるトポロジー

以下の推奨トポロジーの例は、4 台のサーバーに対して考えられる推奨トポロジーの 1 つを示しています。各サーバーは少なくとも 2 つの他のサーバーに接続されており、複数のサーバーが CA サーバーです。

図1.4 推奨されるトポロジーの例

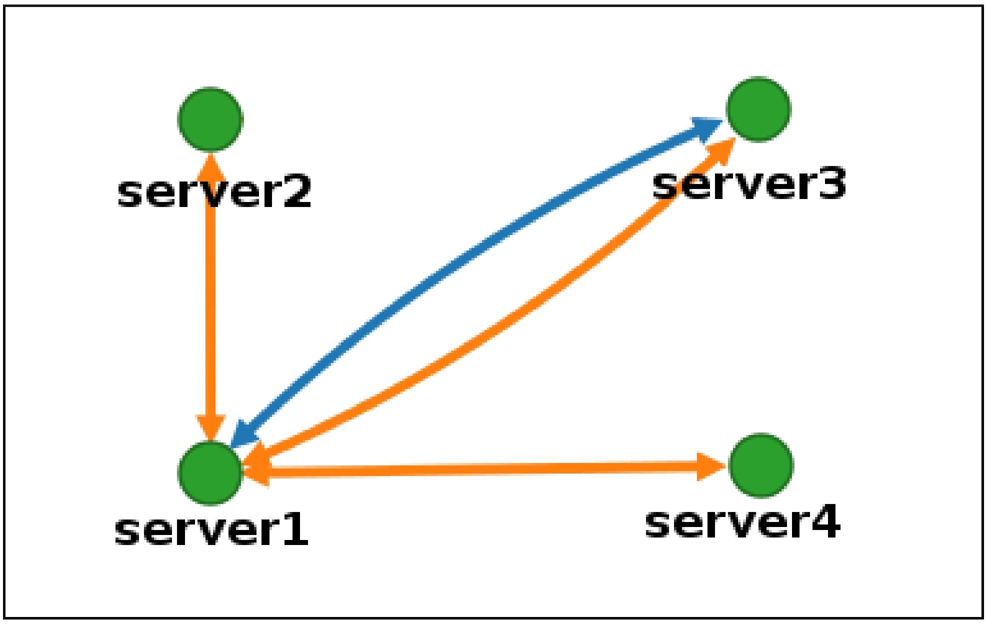


トポロジーグラフの例: 推奨されないトポロジー

推奨されないトポロジーの例では、**server1** が単一障害点になります。その他のすべてのサーバーは、このサーバーとのレプリカ合意がありますが、他のサーバーとは合意がありません。したがって、**server1** が失敗すると、他のすべてのサーバーは分離されます。

このようなトポロジーの作成は避けてください。

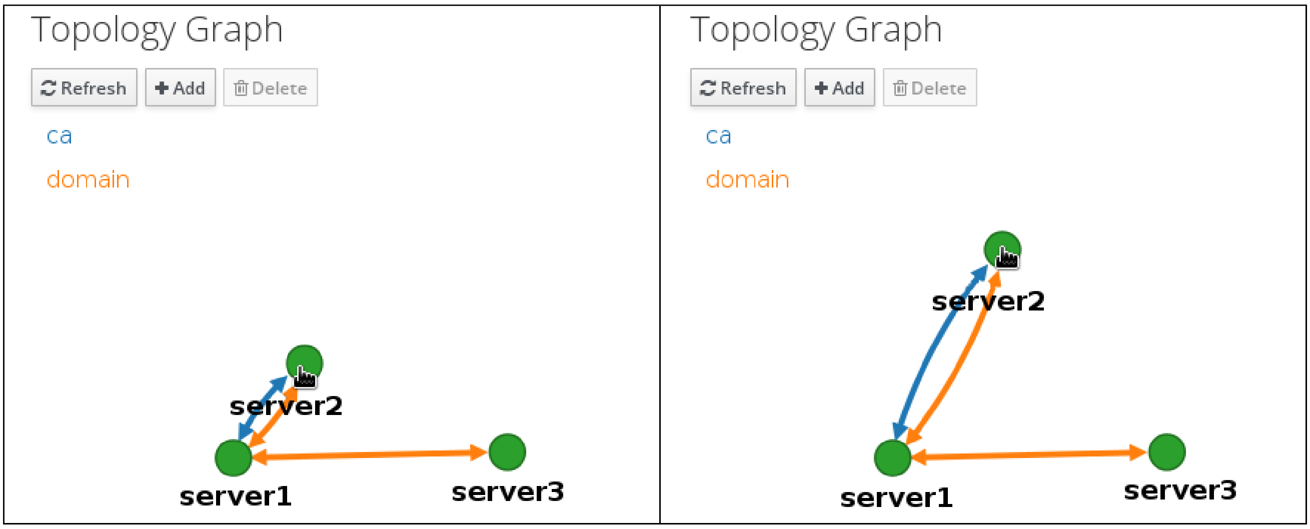
図1.5 推奨されないトポロジーの例: 単一障害点



トポロジービューのカスタマイズ

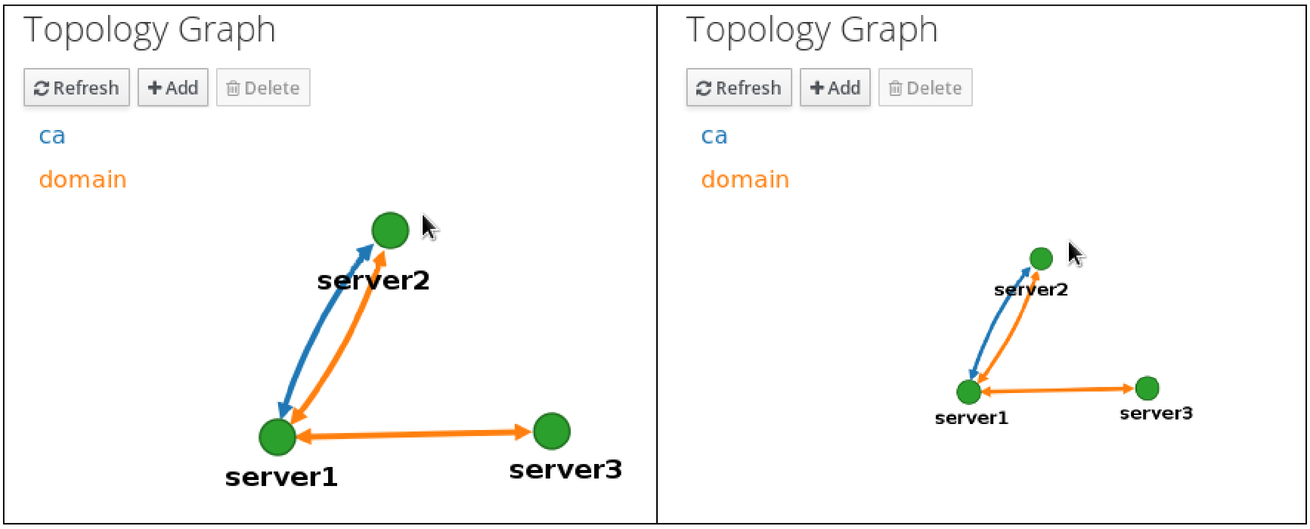
マウスをドラッグして、個別のトポロジーノードを移動できます。

図1.6 トポロジーグラフのノードの移動



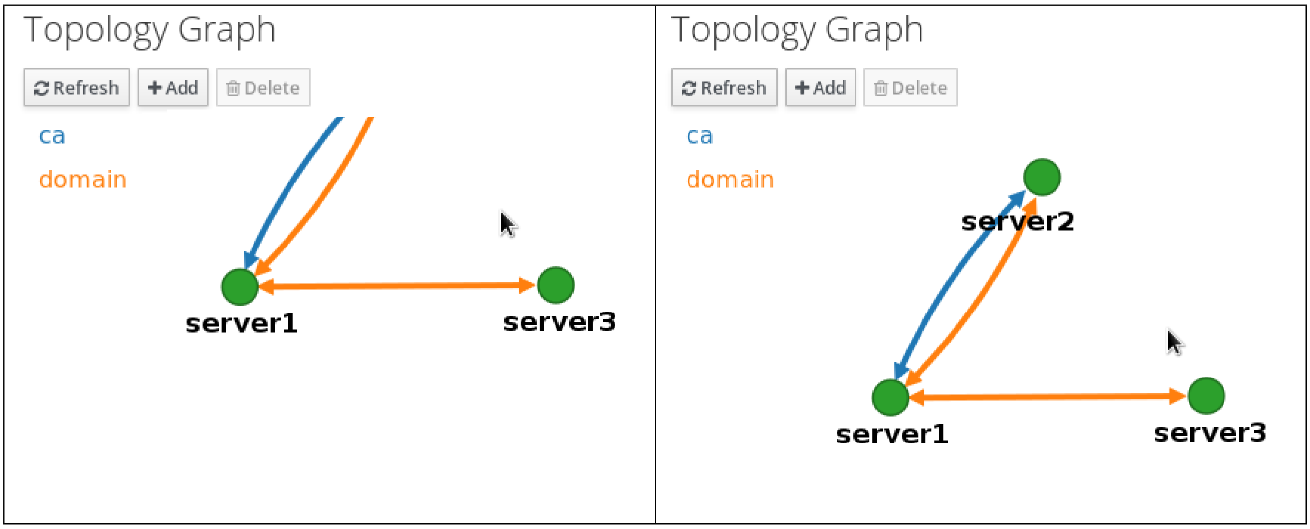
マウスのホイールを使用して、トポロジーグラフを拡大および縮小できます。

図1.7 トポロジーグラフのズーム



マウスの左ボタンを保持することで、トポロジーグラフのキャンバスを移動できます。

図1.8 トポロジーグラフのキャンバスの移動



1.3. WEB UI を使用した 2 台のサーバー間のレプリケーションの設定

Identity Management (IdM) の Web インターフェイスを使用すると、2 つのサーバーを選択し、そのサーバー間に新しいレプリカ合意を作成できます。

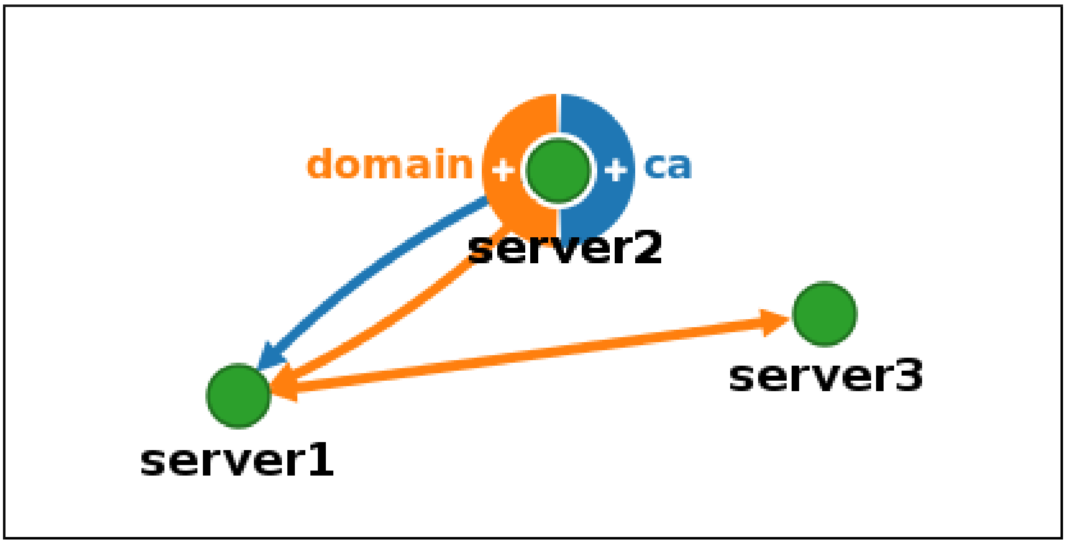
前提条件

- IdM 管理者認証情報がある。

手順

1. トポロジーグラフで、サーバーノードの 1 つにマウスを合わせます。

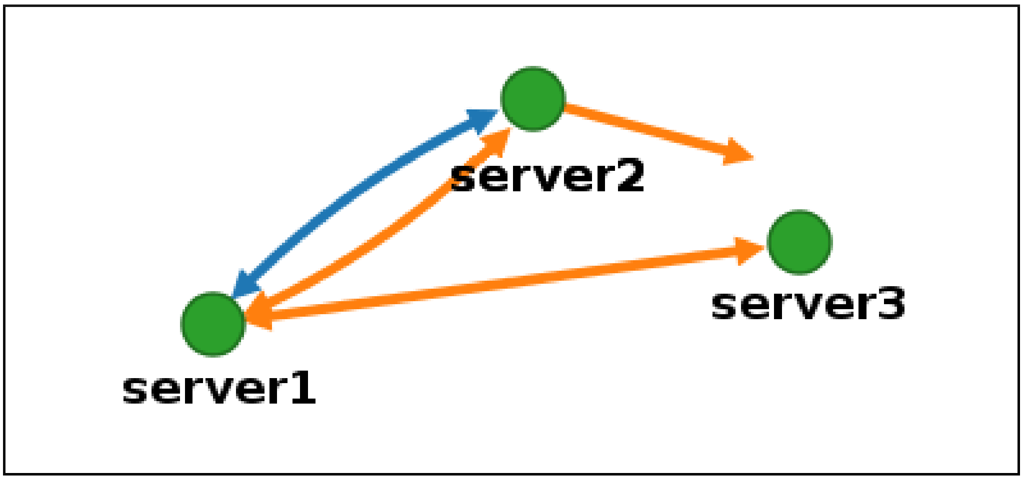
図1.9 ドメインまたは CA オプション



2. 作成するトポロジーセグメントのタイプに応じて、**domain** または円の **ca** 部分をクリックします。

3. 新しいレプリカ合意を表す新しい矢印が、マウスポインターの下に表示されます。マウスを他のサーバーノードに移動し、そこでクリックします。

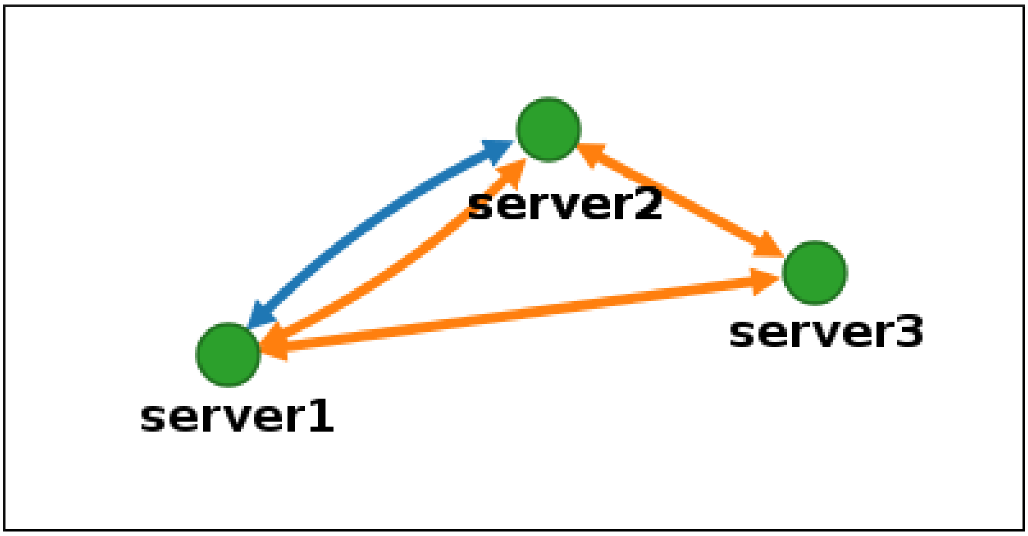
図1.10 新規セグメントの作成



4. **Add Topology Segment** ウィンドウで **Add** をクリックして、新規セグメントのプロパティーを確認します。

2 台のサーバー間の新しいトポロジーセグメントは、サーバーをレプリカ合意に参加させます。トポロジーグラフには、更新されたレプリケーショントポロジーが表示されるようになりました。

図1.11 新規に作成されたセグメント



1.4. WEB UI を使用した 2 台のサーバー間のレプリケーションの停止

Identity Management (IdM) の Web インターフェイスを使用して、サーバーからレプリカ合意を削除できます。

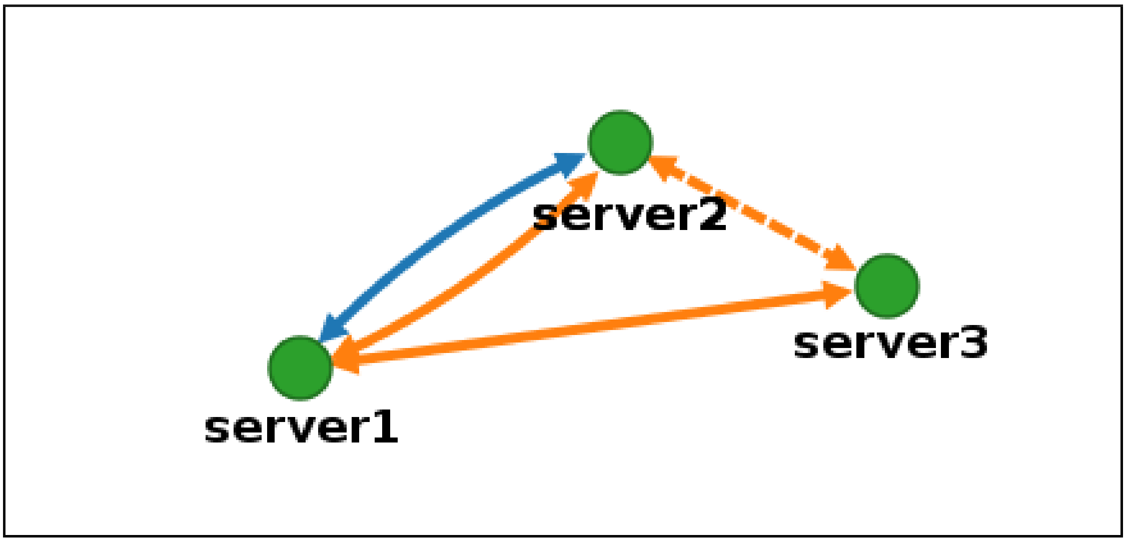
前提条件

- IdM 管理者認証情報がある。

手順

1. 削除するレプリカ合意を表す矢印をクリックします。これにより、矢印がハイライト表示されます。

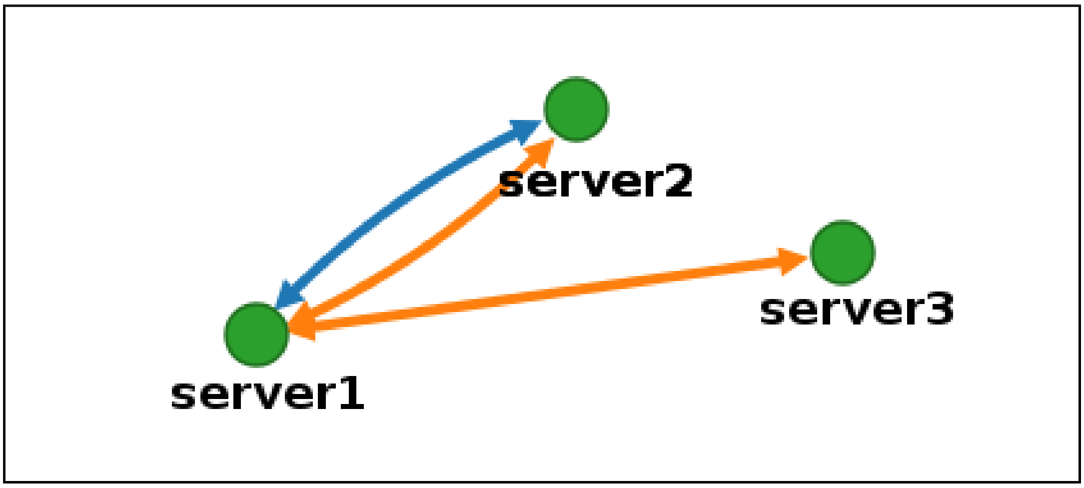
   図1.12 トポロジーセグメントのハイライト表示

2. **Delete** をクリックします。
3. **Confirmation** ウィンドウで **OK** をクリックします。

IdM は、2 台のサーバー間のトポロジーセグメントを削除します。これにより、そのレプリカ合意が削除されます。トポロジーグラフには、更新されたレプリケーショントポロジーが表示されるようになりました。

図1.13 トポロジーセグメントの削除



1.5. CLI を使用した 2 つのサーバー間のレプリケーションの設定

ipa topologysegment-add コマンドを使用して、2 台のサーバー間のレプリカ合意を設定できます。

前提条件

- IdM 管理者認証情報がある。

手順

1. **ipa topologysegment-add** コマンドを使用して、2 つのサーバーのトポロジーセグメントを作成します。プロンプトが表示されたら、以下を指定します。
   - 必要なトポロジー接尾辞: **domain** または **ca**
   - 2 つのサーバーを表す、左ノードと右のノード
   - オプションで、セグメントのカスタム名
     以下に例を示します。

```
$ ipa topologysegment-add
Suffix name: domain
Left node: server1.example.com
Right node: server2.example.com
Segment name [server1.example.com-to-server2.example.com]: new_segment
---------------------------
Added segment "new_segment"
---------------------------
  Segment name: new_segment
  Left node: server1.example.com
  Right node: server2.example.com
  Connectivity: both
```

   新しいセグメントを追加すると、サーバーをレプリカ合意に参加させます。

2. **オプション:ipa topologysegment-show** コマンドを使用して、新しいセグメントが設定されたことを確認します。

```
$ ipa topologysegment-show
```

```
Suffix name: domain
Segment name: new_segment
  Segment name: new_segment
  Left node: server1.example.com
  Right node: server2.example.com
  Connectivity: both
```

1.6. CLI を使用した 2 つのサーバー間のレプリケーションの停止

ipa topology_segment-del コマンドを使用して、コマンドラインからレプリカ合意を終了できます。

前提条件

- IdM 管理者認証情報がある。

手順

1. レプリケーションを停止するには、サーバー間の対応するレプリケーションセグメントを削除する必要があります。これを実行するには、セグメント名を知っている必要があります。
   名前が分からない場合は、**ipa topologysegment-find** コマンドを使用してすべてのセグメントを表示し、出力で必要なセグメントを見つけます。プロンプトが表示されたら、必要なトポロジー接尾辞 (**domain** または **ca**) を指定します。以下に例を示します。

   ```
   $ ipa topologysegment-find
   Suffix name: domain
   ------------------
   8 segments matched
   ------------------
     Segment name: new_segment
     Left node: server1.example.com
     Right node: server2.example.com
     Connectivity: both

   ...

   ----------------------------
   Number of entries returned 8
   ----------------------------
   ```

2. **ipa topologysegment-del** コマンドを使用して、2 台のサーバー間のトポロジーセグメントを削除します。

   ```
   $ ipa topologysegment-del
   Suffix name: domain
   Segment name: new_segment
   -----------------------------
   Deleted segment "new_segment"
   -----------------------------
   ```

   セグメントを削除すると、レプリカ合意が削除されます。

3. **オプション:ipa topologysegment-find** コマンドを使用して、セグメントが表示されなくなったことを確認します。

```
$ ipa topologysegment-find
Suffix name: domain
------------------
7 segments matched
------------------
  Segment name: server2.example.com-to-server3.example.com
  Left node: server2.example.com
  Right node: server3.example.com
  Connectivity: both

...

----------------------------
Number of entries returned 7
----------------------------
```

1.7. WEB UI を使用したトポロジーからのサーバーの削除

Identity Management (IdM) の Web インターフェイスを使用して、トポロジーからサーバーを削除できます。

前提条件

- IdM 管理者認証情報がある。
- 削除するサーバーが、残りのトポロジーで他のサーバーに接続する **唯一のサーバーではない。** この場合、他のサーバーが分離されますが、これは許可されていません。
- 削除するサーバーが、最後の CA または DNS サーバー **ではない。**

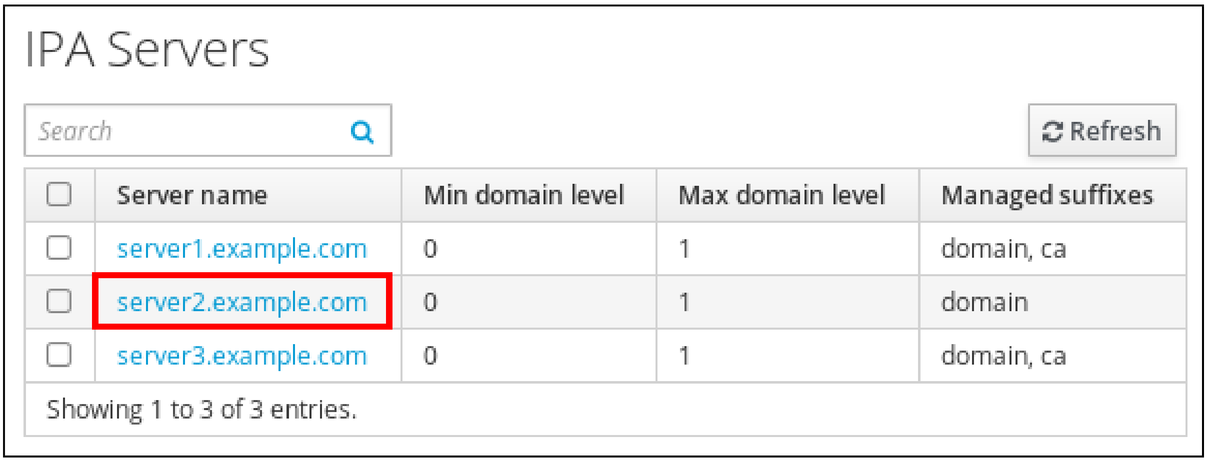
> **警告**
>
> サーバーの削除は元に戻せないアクションです。サーバーを削除すると、トポロジーに戻す唯一の方法は、マシンに新しいレプリカをインストールすることです。

手順

サーバーコンポーネントをマシンからアンインストールせずにトポロジーからサーバーを削除するには、以下を実行します。

1. **IPA Server** → **Topology** → **IPA Servers** を選択します。
2. 削除するサーバーの名前をクリックします。

図1.14 サーバーの選択



3. **Delete Server** をクリックします。

1.8. CLI を使用したトポロジーからのサーバーの削除

コマンドラインインターフェイスを使用して、トポロジーからサーバーを削除できます。

前提条件

- IdM 管理者認証情報がある。
- 削除するサーバーが、残りのトポロジーで他のサーバーに接続する **唯一のサーバーではない**。この場合、他のサーバーが分離されますが、これは許可されていません。
- 削除するサーバーが、最後の CA または DNS サーバー **ではない**。

> **重要**
>
> サーバーの削除は元に戻せないアクションです。サーバーを削除すると、トポロジーに戻す唯一の方法は、マシンに新しいレプリカをインストールすることです。

手順

server1.example.com を削除するには、次のコマンドを実行します。

1. 別のサーバーで **ipa server-del** コマンドを実行して、**server1.example.com** を削除します。このコマンドは、サーバーを参照するすべてのトポロジーセグメントを削除します。

```
[user@server2 ~]$ ipa server-del
Server name: server1.example.com
Removing server1.example.com from replication topology, please wait...
----------------------------------------------------------
Deleted IPA server "server1.example.com"
----------------------------------------------------------
```

2. **オプション: server1.example.com** で、**ipa server-install --uninstall** コマンドを実行して、マシンからサーバーコンポーネントをアンインストールします。

```
[root@server1 ~]# ipa server-install --uninstall
```

1.9. WEB UI を使用した IDM サーバーでのサーバーロールの表示

IdM サーバーにインストールされるサービスに基づいて、さまざまな **サーバーロール** を実行できます。以下に例を示します。

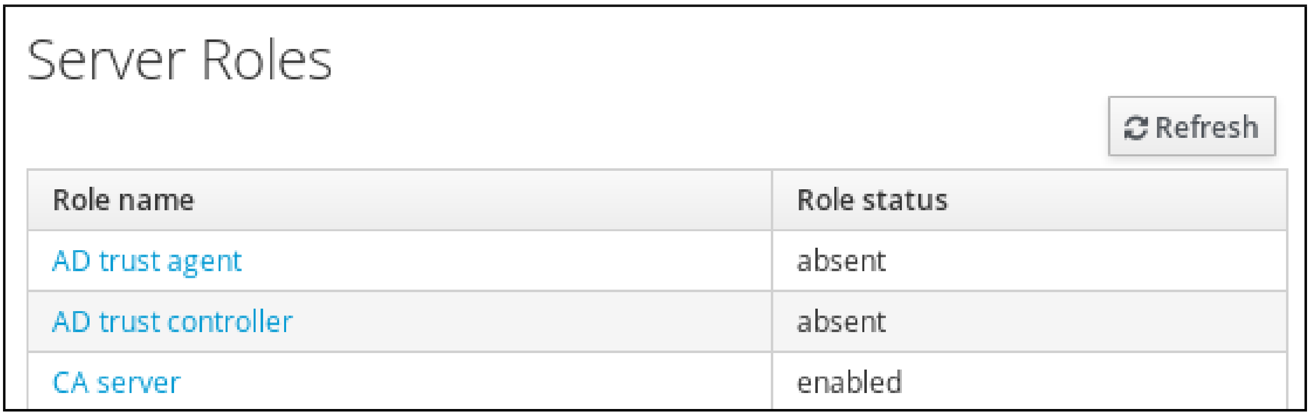
- CA サーバー
- DNS サーバー
- キーリカバリー認証局 (KRA) サーバー

サポートされるサーバーロールの完全なリストは、**IPA Server → Topology → Server Roles**を参照してください。

注記

- Role status が **absent** の場合は、トポロジー内でそのロールを実行しているサーバーがないことを示しています。
- Role status が **enabled** の場合は、トポロジー内でそのロールを実行しているサーバーが 1 台以上あることを示しています。

図1.15 Web UI でのサーバーロール



1.10. CLI を使用した IDM サーバーでのサーバーロールの表示

IdM サーバーにインストールされるサービスに基づいて、さまざまな **サーバーロール** を実行できます。以下に例を示します。

- CA サーバー
- DNS サーバー
- キーリカバリー認証局 (KRA) サーバー

以下のコマンドを使用して、トポロジー内でどのサーバーがどのロールを実行するかを表示できます。

- **ipa config-show** コマンドを実行すると、すべての CA サーバーおよび現行の CA 更新サーバーが表示されます。

```
$ ipa config-show
  ...
  IPA masters: server1.example.com, server2.example.com, server3.example.com
```

```
IPA CA servers: server1.example.com, server2.example.com
IPA CA renewal master: server1.example.com
```

- **ipa server-show** コマンドは、特定のサーバーで有効なロールのリストを表示します。たとえば、server.example.com で有効にしたロールのリストは、以下のようになります。

```
$ ipa server-show
Server name: server.example.com
 ...
 Enabled server roles: CA server, DNS server, KRA server
```

- **ipa server-find --servrole** は、特定のサーバーロールが有効になっているすべてのサーバーを検索します。たとえば、すべての CA サーバーを検索するには、以下を実行します。

```
$ ipa server-find --servrole "CA server"
---------------------
2 IPA servers matched
---------------------
  Server name: server1.example.com
  ...

  Server name: server2.example.com
  ...
----------------------------
Number of entries returned 2
----------------------------
```

1.11. レプリカの CA 更新サーバーおよび CRL パブリッシャーサーバーへのプロモート

IdM デプロイメントで組み込み認証局 (CA) を使用する場合は、IdM CA サーバーの 1 つが CA サブシステム証明書の更新を管理する CA 更新サーバーとして機能します。IdM CA サーバーの 1 つは、証明書失効リストを生成する IdM CRL パブリッシャーサーバーとしても機能します。デフォルトでは、CA 更新サーバーおよび CRL パブリッシャーサーバーロールは、システム管理者が **ipa-server-install** または **ipa-ca-install** コマンドを使用して CA ロールをインストールした最初のサーバーにインストールされます。

前提条件

- IdM 管理者認証情報がある。

1.12. 非表示レプリカの降格または昇格

手順

レプリカのインストール後、レプリカの表示状態を設定できます。

非表示のレプリカの詳細は、非表示のレプリカモード を参照してください。

レプリカが CA 更新サーバーである場合は、このレプリカを非表示にする前に、サービスを別のレプリカに移動します。

詳細は以下を参照してください。

手順

- レプリカを非表示にするには、次のコマンドを実行します。

  ```
  # ipa server-state replica.idm.example.com --state=hidden
  ```

  次のコマンドを実行すれば、レプリカを表示できます

  ```
  # ipa server-state replica.idm.example.com --state=enabled
  ```

  トポロジー内のすべての非表示のレプリカのリストを表示するには、次のコマンドを実行します。

  ```
  # ipa config-show
  ```

  すべてのレプリカが有効になっている場合は、コマンドの出力に非表示のレプリカは記載されません。

第2章 ANSIBLE PLAYBOOK を使用して IDM を管理する環境の準備

Identity Management (IdM) を管理するシステム管理者は、Red Hat Ansible Engine を使用する際に以下を行うことが推奨されます。

- ホームディレクトリーに Ansible Playbook 専用のサブディレクトリー (例: **~/MyPlaybooks**) を作成します。
- **/usr/share/doc/ansible-freeipa/*** と **/usr/share/doc/rhel-system-roles/*** ディレクトリーおよびサブディレクトリーから **~/MyPlaybooks** ディレクトリーにサンプル Ansible Playbook をコピーして調整します。
- **~/MyPlaybooks** ディレクトリーにインベントリーファイルを追加します。

このプラクティスを使用すると、すべての Playbook を 1 カ所で見つけることができます。また、root 権限を呼び出しなくても Playbook を実行できます。

注記

マネージドノードで **root** 権限があれば、**ipaserver**、**ipareplica**、**ipaclient**、および **ipabackup ansible-freeipa** ロールを実行できます。これらのロールには、ディレクトリーおよび **dnf** ソフトウェアパッケージマネージャーへの特権アクセスが必要です。

~/MyPlaybooks ディレクトリーを作成し、それを使用して Ansible Playbook を保存および実行できるように設定するには、次の手順に従います。

前提条件

- 管理対象ノードに IdM サーバー (**server.idm.example.com** および **replica.idm.example.com**) をインストールしている。
- DNS およびネットワークを設定し、コントロールノードから直接管理対象ノード (**server.idm.example.com** および **replica.idm.example.com**) にログインすることができる。
- IdM **admin** のパスワードを把握している。

手順

1. Ansible 設定および Playbook のディレクトリーをホームディレクトリーに作成します。

```
$ mkdir ~/MyPlaybooks/
```

2. **~/MyPlaybooks/** ディレクトリーに移動します。

```
$ cd ~/MyPlaybooks
```

3. **~/MyPlaybooks/ansible.cfg** ファイルを以下の内容で作成します。

```
[defaults]
inventory = /home/your_username/MyPlaybooks/inventory
```

```
[privilege_escalation]
become=True
```

4. **~/MyPlaybooks/inventory** ファイルを以下の内容で作成します。

```
[eu]
server.idm.example.com

[us]
replica.idm.example.com

[ipaserver:children]
eu
us
```

この設定は、これらの場所にあるホストの 2 つのホストグループ (**eu** と **us**) を定義します。さらに、この設定は、**eu** および **us** グループのすべてのホストを含む **ipaserver** ホストグループを定義します。

5. [オプション] SSH 公開鍵および秘密鍵を作成します。テスト環境でのアクセスを簡素化するには、秘密鍵にパスワードを設定しないでください。

```
$ ssh-keygen
```

6. 各マネージドノードの IdM **admin** アカウントに SSH 公開鍵をコピーします。

```
$ ssh-copy-id admin@server.idm.example.com
$ ssh-copy-id admin@replica.idm.example.com
```

これらのコマンドでは、IdM **管理者** パスワードを入力します。

関連情報

- Ansible Playbook を使用した Identity Management サーバーのインストール を参照してください。
- インベントリーの構築方法 を参照してください。

第3章 ANSIBLE を使用した IDM でのレプリケーショントポロジーの管理

複数の Identity Management (IdM) サーバーを維持し、冗長性の目的で相互に複製して、サーバーの損失を軽減または防止することができます。たとえば、1 台のサーバーに障害が発生しても、その他のサーバーがドメインにサービスを提供し続けます。障害が発生していないサーバーの 1 台から新しいレプリカを作成し、失われたサーバーを回復することもできます。

IdM サーバーに保存されているデータは、レプリカ合意に基づいて複製されます。2 台のサーバーでレプリカ合意が設定されている場合は、データを共有します。レプリケートされるデータはトポロジーの **suffix** に保存されます。2 つのレプリカに接尾辞間でレプリカ合意があると、接尾辞はトポロジー **segment** を形成します。

本章では、**Red Hat Ansible Engine** を使用して IdM レプリカ合意、トポロジーセグメント、およびトポロジー接尾辞を管理する方法を説明します。本章は以下のセクションで設定されます。

- Ansible を使用して、レプリカ合意が IdM に存在することを確認
- Ansible を使用して複数の IdM レプリカ間でレプリカ合意を存在させる手順
- Ansible を使用して 2 つのレプリカ間でレプリカ合意が存在するかどうかの確認
- Ansible を使用してトポロジーの接尾辞が IdM に存在することを確認
- Ansible を使用した IdM レプリカの再初期化
- Ansible を使用して IdM にレプリカ合意がないことを確認する手順

3.1. ANSIBLE を使用して、レプリカ合意が IDM に存在することを確認

Identity Management (IdM) サーバーに保存されているデータは、レプリカ合意に基づいて複製されます。2 台のサーバーでレプリカ合意が設定されている場合は、データを共有します。レプリカ合意は常に双方向のものです。最初のレプリカからサーバーから別のレプリカにデータが複製されるだけでなく、別ののレプリカから最初のレプリカにもデータが複製されます。

この手順に従い、Ansible Playbook を使用して、**server.idm.example.com** と **replica.idm.example.com** との間で **domain** タイプのレプリカ合意が存在することを確認説明します。

前提条件

- トポロジーで IdM レプリカを接続するためのガイドライン に記載されている IdM トポロジーを設計するための推奨事項を確実に理解している。
- IdM **admin** のパスワードを把握している。
- 次の要件を満たすように Ansible コントロールノードを設定している。
    - Ansible バージョン 2.14 以降を使用している。
    - Ansible コントローラーに **ansible-freeipa** パッケージがインストールされている。
    - **~/MyPlaybooks/** ディレクトリーに、IdM サーバーの完全修飾ドメイン名 (FQDN) を使用して Ansible インベントリーファイル を作成している (この例の場合)。
    - この例では、**secret.yml** Ansible ボールトに **ipaadmin_password** が保存されていることを前提としている。

- **ansible-freeipa** モジュールが実行されるノードであるターゲットノードは、IdM クライアント、サーバー、またはレプリカとしての IdM ドメインの一部です。

手順

1. **~/MyPlaybooks/** ディレクトリーに移動します。

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

2. **/usr/share/doc/ansible -freeipa/playbooks/topology/** ディレクトリーにある **add-topologysegment.yml** Ansible Playbook ファイルをコピーします。

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/topology/add-topologysegment.yml
add-topologysegment-copy.yml
```

3. **add-topologysegment-copy.yml** ファイルを開いて編集します。

4. **ipatopologysegment** タスクセクションに以下の変数を設定して、ファイルを調整します。

    - **ipaadmin_password** 変数は IdM **admin** のパスワードに設定します。
    - 追加するセグメントのタイプに応じて、**suffix** 変数を **domain** または **ca** のいずれかに設定します。
    - **left** の変数をレプリカ合意の左ノードに設定する IdM サーバーの名前に設定します。
    - レプリカ合意の適切なノードとなる IdM サーバーの名前に **right** 変数を設定します。
    - **state** 変数は **present** に設定されていることを確認します。

    以下は、今回の例で使用するように変更した Ansible Playbook ファイルです。

```
---
- name: Playbook to handle topologysegment
  hosts: ipaserver

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
- name: Add topology segment
    ipatopologysegment:
      ipaadmin_password: "{{ ipaadmin_password }}"
      suffix: domain
      left: server.idm.example.com
      right: replica.idm.example.com
      state: present
```

5. ファイルを保存します。

6. Ansible Playbook を実行します。Playbook ファイル、**secret.yml** ファイルを保護するパスワードを格納するファイル、およびインベントリーファイルを指定します。

```
$ ansible-playbook --vault-password-file=password_file -v -i inventory add-
topologysegment-copy.yml
```

関連情報

- レプリカ合意、トポロジー接尾辞、およびトポロジーセグメントの説明 を参照してください。
- **/usr/share/doc/ansible-freeipa/** ディレクトリーの **README-topology.md** ファイルを参照してください。
- **/usr/share/doc/ansible-freeipa/playbooks/topology** ディレクトリーのサンプルの Playbook を参照してください。

3.2. ANSIBLE を使用して複数の IDM レプリカ間でレプリカ合意を存在させる手順

Identity Management (IdM) サーバーに保存されているデータは、レプリカ合意に基づいて複製されます。2 台のサーバーでレプリカ合意が設定されている場合は、データを共有します。レプリカ合意は常に双方向のものです。最初のレプリカからサーバーから別のレプリカにデータが複製されるだけでなく、別ののレプリカから最初のレプリカにもデータが複製されます。

以下の手順に従って、IdM の複数のレプリカのペア間でレプリカ合意が存在することを確認します。

前提条件

- トポロジーで IdM レプリカを接続するためのガイドライン に記載されている IdM トポロジーを設計するための推奨事項を確実に理解している。
- IdM **admin** のパスワードを把握している。
- 次の要件を満たすように Ansible コントロールノードを設定している。
  - Ansible バージョン 2.14 以降を使用している。
  - Ansible コントローラーに **ansible-freeipa** パッケージがインストールされている。
  - **~/MyPlaybooks/** ディレクトリーに、IdM サーバーの完全修飾ドメイン名 (FQDN) を使用して Ansible インベントリーファイル を作成している (この例の場合)。
  - この例では、**secret.yml** Ansible ボールトに **ipaadmin_password** が保存されていることを前提としている。
- **ansible-freeipa** モジュールが実行されるノードであるターゲットノードは、IdM クライアント、サーバー、またはレプリカとしての IdM ドメインの一部です。

手順

1. **~/MyPlaybooks/** ディレクトリーに移動します。

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

2. **/usr/share/doc/ansible -freeipa/playbooks/topology/** ディレクトリーにある **add-topologysegments.yml** Ansible Playbook ファイルをコピーします。

   ```
   $ cp /usr/share/doc/ansible-freeipa/playbooks/topology/add-topologysegments.yml
   add-topologysegments-copy.yml
   ```

3. **add-topologysegments-copy.yml** ファイルを開いて編集します。

4. **vars** セクションに以下の変数を設定して、ファイルを調整します。
    - **ipaadmin_password** 変数は IdM **admin** のパスワードに設定します。
    - すべてのトポロジーセグメントについて、**ipatopology_segments** セクションに行を追加し、以下の変数を設定します。
        - 追加するセグメントのタイプに応じて、**suffix** 変数を **domain** または **ca** のいずれかに設定します。
        - **left** の変数をレプリカ合意の左ノードに設定する IdM サーバーの名前に設定します。
        - レプリカ合意の適切なノードとなる IdM サーバーの名前に **right** 変数を設定します。
5. **add-topologysegments-copy.yml** ファイルの **tasks** セクションで、**state** 変数が **present** に設定されていることを確認します。
以下は、今回の例で使用するように変更した Ansible Playbook ファイルです。

```
---
- name: Add topology segments
  hosts: ipaserver
  gather_facts: false

  vars:
    ipaadmin_password: "{{ ipaadmin_password }}"
    ipatopology_segments:
    - {suffix: domain, left: replica1.idm.example.com , right: replica2.idm.example.com }
    - {suffix: domain, left: replica2.idm.example.com , right: replica3.idm.example.com }
    - {suffix: domain, left: replica3.idm.example.com , right: replica4.idm.example.com }
    - {suffix: domain+ca, left: replica4.idm.example.com , right: replica1.idm.example.com }

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Add topology segment
    ipatopologysegment:
      ipaadmin_password: "{{ ipaadmin_password }}"
      suffix: "{{ item.suffix }}"
      name: "{{ item.name | default(omit) }}"
      left: "{{ item.left }}"
      right: "{{ item.right }}"
      state: present
      #state: absent
      #state: checked
      #state: reinitialized
    loop: "{{ ipatopology_segments | default([]) }}"
```

6. ファイルを保存します。
7. Ansible Playbook を実行します。Playbook ファイル、**secret.yml** ファイルを保護するパスワードを格納するファイル、およびインベントリーファイルを指定します。

```
$ ansible-playbook --vault-password-file=password_file -v -i inventory add-topologysegments-copy.yml
```

関連情報

- レプリカ合意、トポロジー接尾辞、およびトポロジーセグメントの説明 を参照してください。
- **/usr/share/doc/ansible-freeipa/** ディレクトリーの **README-topology.md** ファイルを参照してください。
- **/usr/share/doc/ansible-freeipa/playbooks/topology** ディレクトリーのサンプルの Playbook を参照してください。

3.3. ANSIBLE を使用して 2 つのレプリカ間でレプリカ合意が存在するかどうかの確認

Identity Management (IdM) サーバーに保存されているデータは、レプリカ合意に基づいて複製されます。2 台のサーバーでレプリカ合意が設定されている場合は、データを共有します。レプリカ合意は常に双方向のものです。最初のレプリカからサーバーから別のレプリカにデータが複製されるだけでなく、別ののレプリカから最初のレプリカにもデータが複製されます。

以下の手順に従って、IdM のレプリカのペア間でレプリカ合意が存在することを確認します。

前提条件

- トポロジーで IdM レプリカを接続するためのガイドライン に記載されている Identity Management (IdM) トポロジーを設計するための推奨事項を確実に理解している。
- IdM **admin** のパスワードを把握している。
- 次の要件を満たすように Ansible コントロールノードを設定している。
  - Ansible バージョン 2.14 以降を使用している。
  - Ansible コントローラーに **ansible-freeipa** パッケージがインストールされている。
  - **~/MyPlaybooks/** ディレクトリーに、IdM サーバーの完全修飾ドメイン名 (FQDN) を使用して Ansible インベントリーファイル を作成している (この例の場合)。
  - この例では、**secret.yml** Ansible ボールトに **ipaadmin_password** が保存されていることを前提としている。
- **ansible-freeipa** モジュールが実行されるノードであるターゲットノードは、IdM クライアント、サーバー、またはレプリカとしての IdM ドメインの一部です。

手順

1. **~/MyPlaybooks/** ディレクトリーに移動します。

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

2. **/usr/share/doc/ansible -freeipa/playbooks/topology/** ディレクトリーにある **check-topologysegments.yml** Ansible Playbook ファイルをコピーします。

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/topology/check-topologysegments.yml
check-topologysegments-copy.yml
```

3. **check-topologysegments-copy.yml** ファイルを開いて編集します。

4. **vars** セクションに以下の変数を設定して、ファイルを調整します。
   - **ipaadmin_password** 変数は IdM **admin** のパスワードに設定します。
   - すべてのトポロジーセグメントについて、**ipatopology_segments** セクションに行を追加し、以下の変数を設定します。
     - 追加するセグメントのタイプに応じて、**suffix** 変数を **domain** または **ca** のいずれかに設定します。
     - **left** の変数をレプリカ合意の左ノードに設定する IdM サーバーの名前に設定します。
     - レプリカ合意の適切なノードとなる IdM サーバーの名前に **right** 変数を設定します。
5. **check-topologysegments-copy.yml** ファイルの **tasks** セクションで、**state** 変数が **present** に設定されていることを確認します。
   以下は、今回の例で使用するように変更した Ansible Playbook ファイルです。

```
---
- name: Add topology segments
  hosts: ipaserver
  gather_facts: false

  vars:
    ipaadmin_password: "{{ ipaadmin_password }}"
    ipatopology_segments:
    - {suffix: domain, left: replica1.idm.example.com, right: replica2.idm.example.com }
    - {suffix: domain, left: replica2.idm.example.com , right: replica3.idm.example.com }
    - {suffix: domain, left: replica3.idm.example.com , right: replica4.idm.example.com }
    - {suffix: domain+ca, left: replica4.idm.example.com , right:
replica1.idm.example.com }

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Check topology segment
    ipatopologysegment:
      ipaadmin_password: "{{ ipaadmin_password }}"
      suffix: "{{ item.suffix }}"
      name: "{{ item.name | default(omit) }}"
      left: "{{ item.left }}"
      right: "{{ item.right }}"
      state: checked
    loop: "{{ ipatopology_segments | default([]) }}"
```

6. ファイルを保存します。
7. Ansible Playbook を実行します。Playbook ファイル、**secret.yml** ファイルを保護するパスワードを格納するファイル、およびインベントリーファイルを指定します。

```
$ ansible-playbook --vault-password-file=password_file -v -i inventory check-topologysegments-copy.yml
```

関連情報

- トポロジー合意、接尾辞、およびセグメントの概念の詳細は、レプリカ合意、トポロジー接尾辞、およびトポロジーセグメント を参照してください。
- **/usr/share/doc/ansible-freeipa/** ディレクトリーの **README-topology.md** ファイルを参照してください。
- **/usr/share/doc/ansible-freeipa/playbooks/topology** ディレクトリーのサンプルの Playbook を参照してください。

3.4. ANSIBLE を使用してトポロジーの接尾辞が IDM に存在することを確認

Identity Management (IdM) のレプリカ合意のコンテキストでは、トポロジー接尾辞はレプリケートされるデータを保存します。IdM は、**domain** と **ca** の 2 種類のトポロジー接尾辞に対応します。それぞれの接尾辞は、個別のバックエンドである個別のレプリケーショントポロジーを表します。レプリカ合意が設定されると、同じタイプのトポロジー接尾辞を 2 つの異なるサーバーに結合します。

domain 接尾辞には、ユーザー、グループ、ポリシーなどのドメイン関連のデータがすべて含まれます。**ca** 接尾辞には、Certificate System コンポーネントのデータが含まれます。これは認証局 (CA) がインストールされているサーバーにのみ存在します。

以下の手順に従って、Ansible Playbook を使用して、トポロジー接尾辞が IdM に存在することを確認します。この例では、**domain** 接尾辞が IdM に存在することを確認する方法を説明します。

前提条件

- IdM **admin** のパスワードを把握している。
- 次の要件を満たすように Ansible コントロールノードを設定している。
  - Ansible バージョン 2.14 以降を使用している。
  - Ansible コントローラーに **ansible-freeipa** パッケージがインストールされている。
  - **~/MyPlaybooks/** ディレクトリーに、IdM サーバーの完全修飾ドメイン名 (FQDN) を使用して Ansible インベントリーファイル を作成している (この例の場合)。
  - この例では、**secret.yml** Ansible ボールトに **ipaadmin_password** が保存されていることを前提としている。
- **ansible-freeipa** モジュールが実行されるノードであるターゲットノードは、IdM クライアント、サーバー、またはレプリカとしての IdM ドメインの一部です。

手順

1. **~/MyPlaybooks/** ディレクトリーに移動します。

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

2. **/usr/share/doc/ansible-freeipa/playbooks/topology/** ディレクトリーにある **verify-topologysuffix.yml** Ansible Playbook ファイルをコピーします。

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/topology/ verify-topologysuffix.yml verify-topologysuffix-copy.yml
```

3. Ansible Playbook ファイル **verify-topologysuffix-copy.yml** を開きます。

4. **ipatopologysuffix** セクションに以下の変数を設定して、ファイルを調整します。
   - **ipaadmin_password** 変数は IdM **admin** のパスワードに設定します。
   - **suffix** 変数は **domain** に設定し ます。**ca** 接尾辞が存在することを確認する場合は、変数を **ca** に設定します。
   - **state** 変数が **verified** に設定されていることを確認します。他のオプションは使用できません。

   以下は、今回の例で使用するように変更した Ansible Playbook ファイルです。

```
---
- name: Playbook to handle topologysuffix
  hosts: ipaserver

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Verify topology suffix
    ipatopologysuffix:
      ipaadmin_password: "{{ ipaadmin_password }}"
      suffix: domain
      state: verified
```

5. ファイルを保存します。
6. Ansible Playbook を実行します。Playbook ファイル、**secret.yml** ファイルを保護するパスワードを格納するファイル、およびインベントリーファイルを指定します。

```
$ ansible-playbook --vault-password-file=password_file -v -i inventory verify-topologysuffix-copy.yml
```

関連情報

- レプリカ合意、トポロジー接尾辞、およびトポロジーセグメントの説明 を参照してください。
- **/usr/share/doc/ansible-freeipa/** ディレクトリーの **README-topology.md** ファイルを参照してください。
- **/usr/share/doc/ansible-freeipa/playbooks/topology** ディレクトリーのサンプルの Playbook を参照してください。

3.5. ANSIBLE を使用した IDM レプリカの再初期化

レプリカが長期間オフラインである場合や、そのデータベースが破損している場合は、初期化できます。初期化により、更新リストのデータでレプリカが更新されます。たとえば、バックアップからの権威復元が必要な場合に使用できます。

注記

レプリケーションの更新とは対照的に、レプリカが変更エントリーのみを送信する間、データベース全体を再初期化します。

コマンドを実行するローカルホストは、再初期化されたレプリカです。データの取得元となるレプリカを指定するには、**direction** オプションを使用します。

以下の手順に従って、Ansible Playbook を使用して **server.idm.example.com** から **replica.idm.example.com** の **domain** データを再初期化します。

前提条件

- IdM **admin** のパスワードを把握している。
- 次の要件を満たすように Ansible コントロールノードを設定している。
  - Ansible バージョン 2.14 以降を使用している。
  - Ansible コントローラーに **ansible-freeipa** パッケージがインストールされている。
  - **~/MyPlaybooks/** ディレクトリーに、IdM サーバーの完全修飾ドメイン名 (FQDN) を使用して Ansible インベントリーファイル を作成している (この例の場合)。
  - この例では、**secret.yml** Ansible ボールトに **ipaadmin_password** が保存されていることを前提としている。
- **ansible-freeipa** モジュールが実行されるノードであるターゲットノードは、IdM クライアント、サーバー、またはレプリカとしての IdM ドメインの一部です。

手順

1. **~/MyPlaybooks/** ディレクトリーに移動します。

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

2. **/usr/share/doc/ansible -freeipa/playbooks/topology/** ディレクトリーにある **reinitialize-topologysegment.yml** Ansible Playbook ファイルをコピーします。

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/topology/reinitialize-topologysegment.yml reinitialize-topologysegment-copy.yml
```

3. **reinitialize-topologysegment-copy.yml** ファイルを開いて編集します。

4. **ipatopologysegment** セクションに以下の変数を設定して、ファイルを調整します。

   - **ipaadmin_password** 変数は IdM **admin** のパスワードに設定します。
   - **suffix** 変数は **domain** に設定し ます。**ca** データを再初期化する場合は、変数を **ca** に設定します。
   - **left** の変数をレプリカ合意の左ノードに設定します。
   - レプリカ合意の **right** なノードに正しい変数を設定します。
   - **direction** 変数は再初期化されるデータの方向に設定します。**left-to-right** は、左のノードから適切なノードにデータフローがあることを意味します。
   - **state** 変数が **reinitialized** に設定されていることを確認します。
     以下は、今回の例で使用するように変更した Ansible Playbook ファイルです。

```
---
- name: Playbook to handle topologysegment
  hosts: ipaserver

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Reinitialize topology segment
    ipatopologysegment:
      ipaadmin_password: "{{ ipaadmin_password }}"
      suffix: domain
      left: server.idm.example.com
      right: replica.idm.example.com
      direction: left-to-right
      state: reinitialized
```

5. ファイルを保存します。

6. Ansible Playbook を実行します。Playbook ファイル、**secret.yml** ファイルを保護するパスワードを格納するファイル、およびインベントリーファイルを指定します。

```
$ ansible-playbook --vault-password-file=password_file -v -i inventory reinitialize-topologysegment-copy.yml
```

関連情報

- レプリカ合意、トポロジー接尾辞、およびトポロジーセグメントの説明 を参照してください。
- **/usr/share/doc/ansible-freeipa/** ディレクトリーの **README-topology.md** ファイルを参照してください。
- **/usr/share/doc/ansible-freeipa/playbooks/topology** ディレクトリーのサンプルの Playbook を参照してください。

3.6. ANSIBLE を使用して IDM にレプリカ合意がないことを確認する手順

Identity Management (IdM) サーバーに保存されているデータは、レプリカ合意に基づいて複製されます。2 台のサーバーでレプリカ合意が設定されている場合は、データを共有します。レプリカ合意は常に双方向のものです。最初のレプリカからサーバーから別のレプリカにデータが複製されるだけでなく、別ののレプリカから最初のレプリカにもデータが複製されます。

以下の手順に従って、2 つのレプリカ間のレプリカ合意が IdM に存在しないことを確認します。この例では、**domain** タイプのレプリカ合意が、**replica01.idm.example.com** と **replica02.idm.example.com** 間で存在させないようにする方法を説明します。

前提条件

- トポロジーで IdM レプリカを接続するためのガイドライン に記載されている IdM トポロジーを設計するための推奨事項を確実に理解している。
- IdM **admin** のパスワードを把握している。
- 次の要件を満たすように Ansible コントロールノードを設定している。
  - Ansible バージョン 2.14 以降を使用している。

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

- **ansible-freeipa** モジュールが実行されるノードであるターゲットノードは、IdM クライアント、サーバー、またはレプリカとしての IdM ドメインの一部です。

手順

1. **~/MyPlaybooks/** ディレクトリーに移動します。

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

2. **/usr/share/doc/ansible -freeipa/playbooks/topology/** ディレクトリーにある **delete-topologysegment.yml** Ansible Playbook ファイルをコピーします。

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/topology/delete-topologysegment.yml
delete-topologysegment-copy.yml
```

3. **delete-topologysegment-copy.yml** ファイルを開いて編集します。

4. **ipatopologysegment** タスクセクションに以下の変数を設定して、ファイルを調整します。

   - **ipaadmin_password** 変数は IdM **admin** のパスワードに設定します。
   - **suffix** 変数は **domain** に設定し ます。また、**ca** データが左右ノードと右のノード間で複製されないようにするには、変数を **ca** に設定します。
   - **left** の変数を、レプリカ合意の左ノードである IdM サーバーの名前に設定します。
   - **右側** の変数を、レプリカ合意の右のノードである IdM サーバーの名前に設定します。
   - **state** 変数は、**absent** に設定されていることを確認します。

   以下は、今回の例で使用するように変更した Ansible Playbook ファイルです。

```
---
- name: Playbook to handle topologysegment
  hosts: ipaserver

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
- name: Delete topology segment
    ipatopologysegment:
      ipaadmin_password: "{{ ipaadmin_password }}"
      suffix: domain
      left: replica01.idm.example.com
      right: replica02.idm.example.com:
      state: absent
```

5. ファイルを保存します。

6. Ansible Playbook を実行します。Playbook ファイル、**secret.yml** ファイルを保護するパスワードを格納するファイル、およびインベントリーファイルを指定します。

```
$ ansible-playbook --vault-password-file=password_file -v -i inventory delete-topologysegment-copy.yml
```

関連情報

- レプリカ合意、トポロジー接尾辞、およびトポロジーセグメントの説明 を参照してください。
- **/usr/share/doc/ansible-freeipa/** ディレクトリーの **README-topology.md** ファイルを参照してください。
- **/usr/share/doc/ansible-freeipa/playbooks/topology** ディレクトリーのサンプルの Playbook を参照してください。

3.7. 関連情報

- Planning the replica topology を参照してください。
- Installing an IdM replica を参照してください。

第4章 非表示レプリカの降格または昇格

レプリカのインストール後、レプリカの表示状態を設定できます。

非表示のレプリカの詳細は、非表示のレプリカモード を参照してください。

レプリカが CA 更新サーバーである場合は、このレプリカを非表示にする前に、サービスを別のレプリカに移動します。

詳細は以下を参照してください。

手順

- レプリカを非表示にするには、次のコマンドを実行します。

  ```
  # ipa server-state replica.idm.example.com --state=hidden
  ```

  次のコマンドを実行すれば、レプリカを表示できます

  ```
  # ipa server-state replica.idm.example.com --state=enabled
  ```

  トポロジー内のすべての非表示のレプリカのリストを表示するには、次のコマンドを実行します。

  ```
  # ipa config-show
  ```

  すべてのレプリカが有効になっている場合は、コマンドの出力に非表示のレプリカは記載されません。

第5章 HEALTHCHECK を使用した IDM レプリケーションの確認

Healthcheck ツールを使用して、Identity Management (IdM) レプリケーションをテストできます。

詳細は IdM の Healthcheck を参照してください。

5.1. レプリケーションの HEALTHCHECK テスト

Healthcheck ツールは、Identity Management (IdM) トポロジーの設定をテストして、レプリケーションの競合問題を検索します。

テストのリストを表示するには、**--list-sources** オプションを指定して、**ipa-healthcheck** を実行します。

```
# ipa-healthcheck --list-sources
```

トポロジーのテストは、**ipahealthcheck.ipa.topology** ソースおよび **ipahealthcheck.ds.replication** ソースの下にあります。

IPATopologyDomainCheck

このテストでは、以下が検証されます。

- トポロジーが切断されておらず、すべてのサーバー間にレプリケーションパスがあるかどうか。
- サーバーに推奨される数以上のレプリカ合意がないかどうか。
  テストに失敗すると、接続エラーやレプリカ合意が多すぎるなどのエラーが返されます。

  テストに成功すると、設定済みのドメインが返されます。

  注記

  このテストでは、ドメインおよび ca 接尾辞の両方で **ipa topologysuffix-verify** コマンドを実行します (認証局がこのサーバーに設定されていることを前提としています)。

ReplicationConflictCheck

このテストでは、**(&(!(objectclass=nstombstone))(nsds5ReplConflict=*))** に一致する LDAP エントリーを検索します。

注記

問題を確認するには、すべての IdM サーバーで上記のテストを実行します。

LDAP レプリケーションの競合を解決する方法の詳細は、一般的なレプリケーションの問題解決 を参照してください。

5.2. HEALTHCHECK を使用したレプリケーションのスクリーニング

Healthcheck ツールを使用して Identity Management (IdM) レプリケーショントポロジーと設定のスタンドアロン手動テストを実行するには、次の手順に従います。

Healthcheck ツールには多くのテストが含まれているため、以下の方法で結果を短くすることができます。

- レプリケーションの競合テスト - **--source=ipahealthcheck.ds.replication**
- 正確なトポロジーテスト - **--source=ipahealthcheck.ipa.topology**

前提条件

- **root** ユーザーとして Healthcheck テストを実行する必要があります。

手順

- Healthcheck のレプリケーションの競合とトポロジーの確認を実行するには、次のコマンドを実行します。

```
# ipa-healthcheck --source=ipahealthcheck.ds.replication --
source=ipahealthcheck.ipa.topology
```

以下のような 4 つの結果が取得できます。

- SUCCESS - テストに成功

```
{
  "source": "ipahealthcheck.ipa.topology",
  "check": "IPATopologyDomainCheck",
  "result": "SUCCESS",
  "kw": {
    "suffix": "domain"
  }
}
```

- WARNING - テストには成功したが、問題の可能性あり
- ERROR - テストが失敗

```
{
  "source": "ipahealthcheck.ipa.topology",
  "check": "IPATopologyDomainCheck",
  "result": "ERROR",
  "uuid": d6ce3332-92da-423d-9818-e79f49ed321f
  "when": 20191007115449Z
  "duration": 0.005943
  "kw": {
    "msg": "topologysuffix-verify domain failed, server2 is not connected
(server2_139664377356472 in MainThread)"
  }
}
```

- CRITICAL - テストが失敗し、IdM サーバー機能に影響が及ぶ

関連情報

- **man ipa-healthcheck** を参照してください。