Red Hat Enterprise Linux 9

Gestion des utilisateurs, des groupes, des hôtes et des règles de contrôle d'accès de l'IdM

Configurer les utilisateurs et les hôtes, les gérer en groupes et contrôler l'accès à l'aide de règles de contrôle d'accès basées sur l'hôte et sur le rôle

Last Updated: 2024-02-20

Red Hat Enterprise Linux 9 Gestion des utilisateurs, des groupes, des hôtes et des règles de contrôle d'accès de l'IdM

Configurer les utilisateurs et les hôtes, les gérer en groupes et contrôler l'accès à l'aide de règles de contrôle d'accès basées sur l'hôte et sur le rôle

Notice légale

Résumé

La principale fonctionnalité de Red Hat Identity Management (IdM) est la gestion des utilisateurs, des groupes, des hôtes et des règles de contrôle d'accès, telles que le contrôle d'accès basé sur l'hôte (HBAC) et le contrôle d'accès basé sur le rôle (RBAC). Vous pouvez les configurer à l'aide de la ligne de commande, de l'interface Web IdM et des Playbooks Ansible. Les tâches de gestion comprennent la configuration des politiques et de la sécurité Kerberos, l'automatisation de l'appartenance aux groupes et la délégation des autorisations.

Table des matières

RENDRE L'OPEN SOURCE PLUS INCLUSIF

Red Hat s'engage à remplacer les termes problématiques dans son code, sa documentation et ses propriétés Web. Nous commençons par ces quatre termes : master, slave, blacklist et whitelist. En raison de l'ampleur de cette entreprise, ces changements seront mis en œuvre progressivement au cours de plusieurs versions à venir. Pour plus de détails, voir le message de notre directeur technique Chris Wright.

FOURNIR UN RETOUR D'INFORMATION SUR LA DOCUMENTATION DE RED HAT

Nous apprécions vos commentaires sur notre documentation. Faites-nous savoir comment nous pouvons l'améliorer.

Soumettre des commentaires sur des passages spécifiques

1. Consultez la documentation au format **Multi-page HTML** et assurez-vous que le bouton **Feedback** apparaît dans le coin supérieur droit après le chargement complet de la page.
2. Utilisez votre curseur pour mettre en évidence la partie du texte que vous souhaitez commenter.
3. Cliquez sur le bouton **Add Feedback** qui apparaît près du texte en surbrillance.
4. Ajoutez vos commentaires et cliquez sur **Submit**.

Soumettre des commentaires via Bugzilla (compte requis)

1. Connectez-vous au site Web de Bugzilla.
2. Sélectionnez la version correcte dans le menu **Version**.
3. Saisissez un titre descriptif dans le champ **Summary**.
4. Saisissez votre suggestion d'amélioration dans le champ **Description**. Incluez des liens vers les parties pertinentes de la documentation.
5. Cliquez sur **Submit Bug**.

CHAPITRE 1. INTRODUCTION AUX UTILITAIRES DE LA LIGNE DE COMMANDE IDM

Les sections suivantes décrivent les bases de l'utilisation des utilitaires de ligne de commande de la gestion des identités (IdM).

Conditions préalables

- Serveur IdM installé et accessible.
  Pour plus de détails, voir Installation de la gestion des identités .
- Pour utiliser l'interface de ligne de commande de l'IPA, authentifiez-vous auprès de l'IdM à l'aide d'un ticket Kerberos valide.

1.1. QU'EST-CE QUE L'INTERFACE DE LIGNE DE COMMANDE DE L'IPA ?

L'interface de ligne de commande (CLI) de l'IPA est l'interface de ligne de commande de base pour l'administration de la gestion des identités (IdM).

Il prend en charge de nombreuses sous-commandes pour la gestion de l'IdM, telles que la commande **ipa user-add** pour ajouter un nouvel utilisateur.

L'interface CLI de l'IPA vous permet de

- Ajouter, gérer ou supprimer des utilisateurs, des groupes, des hôtes et d'autres objets dans le réseau.
- Gérer les certificats.
- Entrées de recherche.
- Afficher et répertorier des objets.
- Définir les droits d'accès.
- Obtenir de l'aide sur la syntaxe correcte de la commande.

1.2. QU'EST-CE QUE L'AIDE IPA ?

L'aide IPA est un système de documentation intégré au serveur IdM.

L'interface de ligne de commande (CLI) IPA génère les rubriques d'aide disponibles à partir des modules d'extension IdM chargés. Pour utiliser l'utilitaire d'aide de l'IPA, vous devez :

- Un serveur IdM doit être installé et fonctionner.
- Être authentifié par un ticket Kerberos valide.

La saisie de la commande **ipa help** sans options affiche des informations sur l'utilisation de l'aide de base et les exemples de commandes les plus courants.

Vous pouvez utiliser les options suivantes pour les différents cas d'utilisation de **ipa help**:

```
$ ipa help [TOPIC | COMMAND | topics | commands]
```

- **[]**- Les parenthèses signifient que tous les paramètres sont facultatifs et que vous pouvez écrire seulement **ipa help** pour que la commande soit exécutée.
- |- Le caractère "pipe" signifie **or**. Par conséquent, vous pouvez spécifier un **TOPIC**, un **COMMAND**, ou un **topics**, ou un **commands**, avec la commande de base **ipa help**:
  - **topics**- Vous pouvez exécuter la commande **ipa help topics** pour afficher une liste de sujets couverts par l'aide IPA, tels que **user**, **cert**, **server** et bien d'autres.
  - **TOPIC**- Le **TOPIC** avec des lettres majuscules est une variable. Vous pouvez donc spécifier un sujet particulier, par exemple **ipa help user**.
  - **commands**- Vous pouvez entrer la commande **ipa help commands** pour afficher une liste de commandes couvertes par l'aide IPA, par exemple, **user-add**, **ca-enable**, **server-show** et bien d'autres.
  - **COMMAND**- Le **COMMAND** avec des lettres majuscules est une variable. Vous pouvez donc spécifier une commande particulière, par exemple **ipa help user-add**.

1.3. UTILISATION DES RUBRIQUES D'AIDE DE L'IPA

La procédure suivante décrit comment utiliser l'aide IPA dans l'interface de ligne de commande.

Procédure

1. Ouvrez un terminal et connectez-vous au serveur IdM.
2. Saisissez **ipa help topics** pour afficher une liste des sujets couverts par l'aide.

   ```
   $ ipa help topics
   ```

3. Sélectionnez l'un des sujets et créez une commande selon le modèle suivant : **ipa help *[topic_name]***. À la place de la chaîne **topic_name**, ajoutez l'un des thèmes énumérés à l'étape précédente.
   Dans l'exemple, nous utilisons le sujet suivant : **user**

   ```
   $ ipa help user
   ```

4. Si l'aide IPA est trop longue et que vous ne pouvez pas voir l'intégralité du texte, utilisez la syntaxe suivante :

   ```
   $ ipa help user | less
   ```

   Vous pouvez ensuite faire défiler la page et lire l'intégralité de l'aide.

L'interface de programmation IPA affiche une page d'aide pour la rubrique **user**. Après avoir lu l'aperçu, vous pouvez voir de nombreux exemples avec des modèles pour travailler avec les commandes de la rubrique.

1.4. UTILISATION DES COMMANDES D'AIDE DE L'IPA

La procédure suivante décrit comment créer des commandes d'aide IPA dans l'interface de ligne de commande.

Procédure

1. Ouvrez un terminal et connectez-vous au serveur IdM.
2. Saisissez **ipa help commands** pour afficher la liste des commandes couvertes par l'aide.

```
$ ipa help commands
```

3. Sélectionnez l'une des commandes et créez une commande d'aide selon le modèle suivant : **ipa help *<COMMAND>***. À la place de la chaîne **<COMMAND>**, ajoutez l'une des commandes énumérées à l'étape précédente.

```
$ ipa help user-add
```

Ressources supplémentaires

- La page de manuel **ipa**.

1.5. STRUCTURE DES COMMANDES IPA

L'interface de programmation IPA distingue les types de commandes suivants :

- **Built-in commands**- Les commandes intégrées sont toutes disponibles dans le serveur IdM.
- **Plug-in provided commands**

La structure des commandes IPA permet de gérer différents types d'objets. Par exemple :

- Utilisateurs,
- Hôtes,
- Enregistrements DNS,
- Certificats,

et bien d'autres.

Pour la plupart de ces objets, l'interface CLI de l'IPA comprend des commandes pour :

- Ajouter (**add**)
- Modifier (**mod**)
- Supprimer (**del**)
- Recherche (**find**)
- Affichage (**show**)

Les commandes ont la structure suivante :

ipa user-add, **ipa user-mod**, **ipa user-del**, **ipa user-find**, **ipa user-show**

ipa host-add, **ipa host-mod**, **ipa host-del**, **ipa host-find**, **ipa host-show**

ipa dnsrecord-add, **ipa dnsrecord-mod**, **ipa dnsrecord-del**, **ipa dnsrecord-find**, **ipa dnrecord-show**

Vous pouvez créer un utilisateur à l'aide de la commande **ipa user-add [options]**, où **[options]** est facultatif. Si vous n'utilisez que la commande **ipa user-add**, le script vous demande les détails un par un.

Pour modifier un objet existant, vous devez définir l'objet, c'est pourquoi la commande comprend également un objet : **ipa user-mod USER_NAME [options]**.

1.6. UTILISATION D'UNE COMMANDE IPA POUR AJOUTER UN COMPTE D'UTILISATEUR À IDM

La procédure suivante décrit comment ajouter un nouvel utilisateur à la base de données Identity Management (IdM) à l'aide de la ligne de commande.

Conditions préalables

- Vous devez disposer de privilèges d'administrateur pour ajouter des comptes d'utilisateurs au serveur IdM.

Procédure

1. Ouvrez un terminal et connectez-vous au serveur IdM.
2. Entrez la commande pour ajouter un nouvel utilisateur :

   ```
   $ ipa user-add
   ```

   La commande exécute un script qui vous invite à fournir les données de base nécessaires à la création d'un compte utilisateur.
3. Dans le champ **First name:**, entrez le prénom du nouvel utilisateur et appuyez sur la touche **Enter**.
4. Dans le champ **Last name:**, entrez le nom de famille du nouvel utilisateur et appuyez sur la touche **Enter**.
5. Dans le champ **User login [suggested user name];** entrez le nom d'utilisateur ou appuyez simplement sur la touche **Enter** pour accepter le nom d'utilisateur proposé.
   Le nom d'utilisateur doit être unique pour toute la base de données IdM. Si une erreur survient parce que ce nom d'utilisateur existe déjà, répétez le processus avec la commande **ipa user-add** et utilisez un nom d'utilisateur différent et unique.

Une fois le nom d'utilisateur ajouté, le compte d'utilisateur est ajouté à la base de données IdM et l'interface de ligne de commande (CLI) de l'API affiche la sortie suivante :

```
----------------------
Added user "euser"
----------------------
User login: euser
First name: Example
Last name: User
Full name: Example User
Display name: Example User
```

```
Initials: EU
Home directory: /home/euser
GECOS: Example User
Login shell: /bin/sh
Principal name: euser@IDM.EXAMPLE.COM
Principal alias: euser@IDM.EXAMPLE.COM
Email address: euser@idm.example.com
UID: 427200006
GID: 427200006
Password: False
Member of groups: ipausers
Kerberos keys available: False
```

NOTE

Par défaut, aucun mot de passe n'est défini pour le compte d'utilisateur. Pour ajouter un mot de passe lors de la création d'un compte utilisateur, utilisez la commande **ipa user-add** avec la syntaxe suivante :

```
$ ipa user-add --first=Example --last=User --password
```

L'interface CLI de l'IPA vous invite ensuite à ajouter ou à confirmer un nom d'utilisateur et un mot de passe.

Si l'utilisateur a déjà été créé, vous pouvez ajouter le mot de passe à l'aide de la commande **ipa user-mod**.

Ressources supplémentaires

- Exécutez la commande **ipa help user-add** pour plus d'informations sur les paramètres.

1.7. UTILISATION D'UNE COMMANDE IPA POUR MODIFIER UN COMPTE D'UTILISATEUR DANS IDM

Vous pouvez modifier de nombreux paramètres pour chaque compte d'utilisateur. Par exemple, vous pouvez ajouter un nouveau mot de passe à l'utilisateur.

La syntaxe de la commande de base est différente de celle de **user-add** car vous devez définir le compte d'utilisateur existant pour lequel vous souhaitez effectuer des modifications, par exemple, ajouter un mot de passe.

Conditions préalables

- Vous devez disposer des droits d'administrateur pour modifier les comptes d'utilisateurs.

Procédure

1. Ouvrez un terminal et connectez-vous au serveur IdM.
2. Entrez la commande **ipa user-mod**, indiquez l'utilisateur à modifier et les options éventuelles, telles que **--password** pour l'ajout d'un mot de passe :

```
$ ipa user-mod euser --password
```

La commande lance un script dans lequel vous pouvez ajouter le nouveau mot de passe.

3. Saisissez le nouveau mot de passe et appuyez sur la touche **Enter**.

L'interface de programmation de l'IPA affiche la sortie suivante :

```
----------------------
Modified user "euser"
----------------------
User login: euser
First name: Example
Last name: User
Home directory: /home/euser
Principal name: euser@IDM.EXAMPLE.COM
Principal alias: euser@IDM.EXAMPLE.COM
Email address: euser@idm.example.com
UID: 427200006
GID: 427200006
Password: True
Member of groups: ipausers
Kerberos keys available: True
```

Le mot de passe de l'utilisateur est maintenant défini pour le compte et l'utilisateur peut se connecter à IdM.

Ressources supplémentaires

- Exécutez la commande **ipa help user-mod** pour plus d'informations sur les paramètres.

1.8. COMMENT FOURNIR UNE LISTE DE VALEURS AUX UTILITAIRES IDM ?

La gestion de l'identité (IdM) stocke les valeurs des attributs à valeurs multiples dans des listes.

L'IdM prend en charge les méthodes suivantes pour fournir des listes à valeurs multiples :

- Utilisation du même argument de ligne de commande plusieurs fois dans la même invocation de commande :

```
$ ipa permission-add --right=read --permissions=write --permissions=delete ...
```

- Vous pouvez également placer la liste entre accolades, auquel cas l'interpréteur de commandes procède à l'expansion :

```
$ ipa permission-add --right={read,write,delete} ...
```

Les exemples ci-dessus montrent une commande **permission-add** qui ajoute des autorisations à un objet. L'objet n'est pas mentionné dans l'exemple. Au lieu de **...** vous devez ajouter l'objet pour lequel vous souhaitez ajouter des autorisations.

Lorsque vous mettez à jour de tels attributs à valeurs multiples à partir de la ligne de commande, l'IdM écrase complètement la liste précédente de valeurs par une nouvelle liste. Par conséquent, lorsque vous mettez à jour un attribut à valeurs multiples, vous devez spécifier l'ensemble de la nouvelle liste, et non pas une seule valeur que vous souhaitez ajouter.

Par exemple, dans la commande ci-dessus, la liste des autorisations comprend la lecture, l'écriture et la suppression. Lorsque vous décidez de mettre à jour la liste avec la commande **permission-mod** vous devez ajouter toutes les valeurs, sinon celles qui ne sont pas mentionnées seront supprimées.

Example 1:- La commande **ipa permission-mod** met à jour toutes les autorisations précédemment ajoutées.

```
$ ipa permission-mod --right=read --right=write --right=delete ...
```

ou

```
$ ipa permission-mod --right={read,write,delete} ...
```

Example 2- La commande **ipa permission-mod** supprime l'argument **--right=delete** car il n'est pas inclus dans la commande :

```
$ ipa permission-mod --right=read --right=write ...
```

ou

```
$ ipa permission-mod --right={read,write} ...
```

1.9. COMMENT UTILISER LES CARACTÈRES SPÉCIAUX AVEC LES UTILITAIRES IDM ?

Lorsque vous transmettez aux commandes **ipa** des arguments de ligne de commande comprenant des caractères spéciaux, échappez ces caractères à l'aide d'une barre oblique inverse (\N). Par exemple, les caractères spéciaux courants sont les crochets (< et >), l'esperluette (&), l'astérisque (*) ou la barre verticale (|).

Par exemple, pour échapper à un astérisque (*) :

```
$ ipa certprofile-show certificate_profile --out=exported\*profile.cfg
```

Les commandes contenant des caractères spéciaux non encapsulés ne fonctionnent pas comme prévu, car l'interpréteur de commandes ne peut pas analyser correctement ces caractères.

CHAPITRE 2. GESTION DES COMPTES D'UTILISATEURS À L'AIDE DE LA LIGNE DE COMMANDE

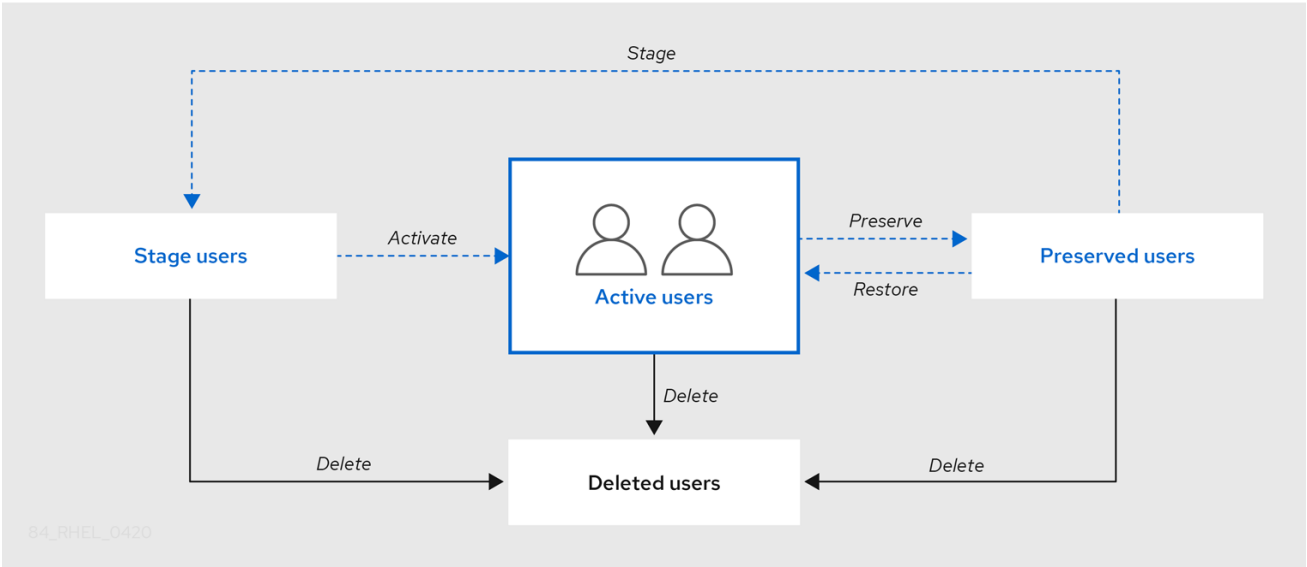
Ce chapitre comprend une description de base du cycle de vie de l'utilisateur dans le cadre de la gestion de l'identité (IdM). Les sections suivantes vous montrent comment :

- Créer des comptes d'utilisateurs
- Activer les comptes d'utilisateurs de l'étape
- Préserver les comptes d'utilisateurs
- Supprimer des comptes d'utilisateurs actifs, d'étape ou préservés
- Restaurer les comptes d'utilisateurs conservés

2.1. CYCLE DE VIE DE L'UTILISATEUR

La gestion des identités (IdM) prend en charge trois états de compte utilisateur :

- **Stage** les utilisateurs ne sont pas autorisés à s'authentifier. Il s'agit d'un état initial. Certaines propriétés du compte utilisateur requises pour les utilisateurs actifs ne peuvent pas être définies, par exemple l'appartenance à un groupe.
- **Active** les utilisateurs sont autorisés à s'authentifier. Toutes les propriétés requises du compte utilisateur doivent être définies dans cet état.
- **Preserved** sont d'anciens utilisateurs actifs qui sont considérés comme inactifs et ne peuvent pas s'authentifier auprès de l'IdM. Les utilisateurs préservés conservent la plupart des propriétés du compte qu'ils avaient en tant qu'utilisateurs actifs, mais ils ne font partie d'aucun groupe d'utilisateurs.



Vous pouvez supprimer définitivement les entrées utilisateur de la base de données IdM.

IMPORTANT

Les comptes d'utilisateurs supprimés ne peuvent pas être restaurés. Lorsque vous supprimez un compte d'utilisateur, toutes les informations associées à ce compte sont définitivement perdues.

Un nouvel administrateur ne peut être créé que par un utilisateur disposant de droits d'administrateur, tel que l'utilisateur admin par défaut. Si vous supprimez accidentellement tous les comptes d'administrateur, le gestionnaire de répertoire doit créer manuellement un nouvel administrateur dans le serveur de répertoire.

AVERTISSEMENT

Ne pas supprimer l'utilisateur **admin**. Comme **admin** est un utilisateur prédéfini requis par l'IdM, cette opération pose des problèmes avec certaines commandes. Si vous souhaitez définir et utiliser un autre utilisateur administrateur, désactivez l'utilisateur prédéfini **admin** avec **ipa user-disable admin** après avoir accordé des droits d'administrateur à au moins un autre utilisateur.

AVERTISSEMENT

N'ajoutez pas d'utilisateurs locaux à IdM. Le commutateur de service de noms (NSS) résout toujours les utilisateurs et les groupes IdM avant de résoudre les utilisateurs et les groupes locaux. Cela signifie, par exemple, que l'appartenance à un groupe IdM ne fonctionne pas pour les utilisateurs locaux.

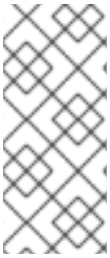
2.2. AJOUTER DES UTILISATEURS À L'AIDE DE LA LIGNE DE COMMANDE

Vous pouvez ajouter un utilisateur en tant que :

- **Active**- les comptes d'utilisateurs qui peuvent être utilisés activement par leurs utilisateurs.
- **Stage**- ne peuvent pas utiliser ces comptes. Utilisez-le si vous souhaitez préparer de nouveaux comptes d'utilisateurs. Lorsque les utilisateurs sont prêts à utiliser leurs comptes, vous pouvez les activer.

La procédure suivante décrit l'ajout d'utilisateurs actifs au serveur IdM à l'aide de la commande **ipa user-add**.

De la même manière, vous pouvez créer des comptes d'utilisateurs de scène à l'aide de la commande **ipa stageuser-add**.



NOTE

L'IdM attribue automatiquement un numéro d'identification unique (UID) aux nouveaux comptes d'utilisateurs. Vous pouvez également le faire manuellement, mais le serveur ne vérifie pas si le numéro UID est unique. Pour cette raison, plusieurs entrées d'utilisateurs peuvent avoir le même numéro d'ID assigné. Red Hat recommande d'éviter d'avoir plusieurs entrées avec le même UID.

Conditions préalables

- Privilèges d'administrateur pour la gestion de l'IdM ou rôle d'administrateur des utilisateurs.
- Obtention d'un ticket Kerberos. Pour plus de détails, voir Utilisation de kinit pour se connecter manuellement à l'IdM.

Procédure

1. Ouvrez un terminal et connectez-vous au serveur IdM.
2. Ajoutez le login de l'utilisateur, son prénom, son nom et, éventuellement, son adresse électronique.

```
$ ipa user-add user_login --first=first_name --last=last_name --email=email_address
```

IdM prend en charge les noms d'utilisateurs qui peuvent être décrits par l'expression régulière suivante :

```
[a-zA-Z0-9_.][a-zA-Z0-9_.-]{0,252}[a-zA-Z0-9_.$-]?
```

NOTE

Les noms d'utilisateur se terminant par le signe du dollar ($) sont pris en charge pour permettre la prise en charge des machines Samba 3.x.

Si vous ajoutez un nom d'utilisateur contenant des caractères majuscules, l'IdM convertit automatiquement le nom en minuscules lorsqu'il est enregistré. Par conséquent, l'IdM exige toujours que les noms d'utilisateurs soient saisis en minuscules lors de l'ouverture d'une session. En outre, il n'est pas possible d'ajouter des noms d'utilisateurs qui ne diffèrent que par la casse des lettres, comme **user** et **User**.

La longueur maximale par défaut des noms d'utilisateur est de 32 caractères. Pour la modifier, utilisez la commande **ipa config-mod --maxusername**. Par exemple, pour augmenter la longueur maximale des noms d'utilisateur à 64 caractères :

```
$ ipa config-mod --maxusername=64
 Maximum username length: 64
 ...
```

La commande **ipa user-add** comprend de nombreux paramètres. Pour les énumérer tous, utilisez la commande ipa help :

```
ipa help user-add
```

Pour plus d'informations sur la commande **ipa help**, voir Qu'est-ce que l'aide IPA ?

Vous pouvez vérifier si le nouveau compte d'utilisateur a été créé avec succès en dressant la liste de tous les comptes d'utilisateur IdM :

```
ipa user-find
```

Cette commande dresse la liste de tous les comptes d'utilisateurs avec leurs détails.

2.3. ACTIVATION DES UTILISATEURS À L'AIDE DE LA LIGNE DE COMMANDE

Pour activer un compte d'utilisateur en le faisant passer de stage à actif, utilisez la commande **ipa stageuser-activate**.

Conditions préalables

- Privilèges d'administrateur pour la gestion de l'IdM ou rôle d'administrateur des utilisateurs.
- Obtention d'un ticket Kerberos. Pour plus de détails, voir Utilisation de kinit pour se connecter manuellement à l'IdM.

Procédure

1. Ouvrez un terminal et connectez-vous au serveur IdM.
2. Activez le compte d'utilisateur à l'aide de la commande suivante :

```
$ ipa stageuser-activate user_login
-------------------------
Stage user user_login activated
-------------------------
...
```

Vous pouvez vérifier si le nouveau compte d'utilisateur a été créé avec succès en dressant la liste de tous les comptes d'utilisateur IdM :

```
ipa user-find
```

Cette commande dresse la liste de tous les comptes d'utilisateurs avec leurs détails.

2.4. PRÉSERVER LES UTILISATEURS À L'AIDE DE LA LIGNE DE COMMANDE

Vous pouvez préserver un compte d'utilisateur si vous souhaitez le supprimer, tout en conservant la possibilité de le restaurer ultérieurement. Pour préserver un compte d'utilisateur, utilisez l'option **--preserve** avec les commandes **ipa user-del** ou **ipa stageuser-del**.

Conditions préalables

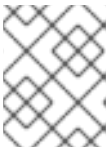
- Privilèges d'administrateur pour la gestion de l'IdM ou rôle d'administrateur des utilisateurs.

- Obtention d'un ticket Kerberos. Pour plus de détails, voir Utilisation de kinit pour se connecter manuellement à l'IdM.

Procédure

1. Ouvrez un terminal et connectez-vous au serveur IdM.
2. Préservez le compte d'utilisateur à l'aide de la commande suivante :

```
$ ipa user-del --preserve user_login
--------------------
Deleted user "user_login"
--------------------
```



NOTE

Bien que le résultat indique que le compte d'utilisateur a été supprimé, il a été préservé.

2.5. SUPPRESSION D'UTILISATEURS À L'AIDE DE LA LIGNE DE COMMANDE

IdM (Identity Management) vous permet de supprimer des utilisateurs de façon permanente. Vous pouvez supprimer :

- Les utilisateurs actifs avec la commande suivante : **ipa user-del**
- Les utilisateurs de l'étape avec la commande suivante : **ipa stageuser-del**
- Les utilisateurs préservés avec la commande suivante : **ipa user-del**

Lors de la suppression de plusieurs utilisateurs, utilisez l'option **--continue** pour forcer la commande à continuer sans tenir compte des erreurs. Un résumé des opérations réussies et échouées est imprimé sur le flux de sortie standard **stdout** lorsque la commande est terminée.

```
ipa user-del --continue user1 user2 user3
```

Si vous n'utilisez pas **--continue**, la commande procède à la suppression des utilisateurs jusqu'à ce qu'elle rencontre une erreur, après quoi elle s'arrête et sort.

Conditions préalables

- Privilèges d'administrateur pour la gestion de l'IdM ou rôle d'administrateur des utilisateurs.
- Obtention d'un ticket Kerberos. Pour plus de détails, voir Utilisation de kinit pour se connecter manuellement à l'IdM.

Procédure

1. Ouvrez un terminal et connectez-vous au serveur IdM.
2. Supprimez le compte d'utilisateur à l'aide de la commande suivante :

```
$ ipa user-del user_login
```

```
--------------------
Deleted user "user_login"
--------------------
```

Le compte d'utilisateur a été définitivement supprimé de l'IdM.

2.6. RESTAURATION DES UTILISATEURS À L'AIDE DE LA LIGNE DE COMMANDE

Vous pouvez restaurer un utilisateur préservé :

- Utilisateurs actifs : **ipa user-undel**
- Utilisateurs du stade : **ipa user-stage**

La restauration d'un compte d'utilisateur ne rétablit pas tous les attributs précédents du compte. Par exemple, le mot de passe de l'utilisateur n'est pas restauré et doit être défini à nouveau.

Conditions préalables

- Privilèges d'administrateur pour la gestion de l'IdM ou rôle d'administrateur des utilisateurs.
- Obtention d'un ticket Kerberos. Pour plus de détails, voir Utilisation de kinit pour se connecter manuellement à l'IdM.

Procédure

1. Ouvrez un terminal et connectez-vous au serveur IdM.
2. Activez le compte d'utilisateur à l'aide de la commande suivante :

```
$ ipa user-undel user_login
----------------------------
Undeleted user account "user_login"
----------------------------
```

Vous pouvez également restaurer les comptes d'utilisateurs en tant que stades :

```
$ ipa user-stage user_login
----------------------------
Staged user account "user_login"
----------------------------
```

Verification steps

- Vous pouvez vérifier si le nouveau compte d'utilisateur a été créé avec succès en dressant la liste de tous les comptes d'utilisateur IdM :

```
ipa user-find
```

Cette commande dresse la liste de tous les comptes d'utilisateurs avec leurs détails.

CHAPITRE 3. GESTION DES COMPTES D'UTILISATEURS À L'AIDE DE L'INTERFACE WEB IDM

La gestion des identités (IdM) propose plusieurs étapes qui peuvent vous aider à gérer les différentes situations de vie professionnelle des utilisateurs :

Création d'un compte utilisateur

Créer un compte d'utilisateur avant qu'un employé ne commence sa carrière dans votre entreprise et être prêt à l'avance pour le jour où l'employé se présentera au bureau et voudra activer le compte. Vous pouvez omettre cette étape et créer directement le compte d'utilisateur actif. La procédure est similaire à la création d'un compte d'utilisateur de scène.

Activation d'un compte d'utilisateur

Activation du compte le premier jour ouvrable de l'employé.

Désactivation d'un compte d'utilisateur

Si l'utilisateur part en congé parental pendant quelques mois, vous devrez désactiver le compte temporairement.

Activation d'un compte d'utilisateur

Lorsque l'utilisateur revient, vous devez réactiver le compte.

Préserver un compte d'utilisateur

Si l'utilisateur souhaite quitter l'entreprise, vous devrez supprimer le compte avec la possibilité de le restaurer, car les personnes peuvent revenir dans l'entreprise après un certain temps.

Restauration d'un compte d'utilisateur

Deux ans plus tard, l'utilisateur est de retour et vous devez restaurer le compte préservé .

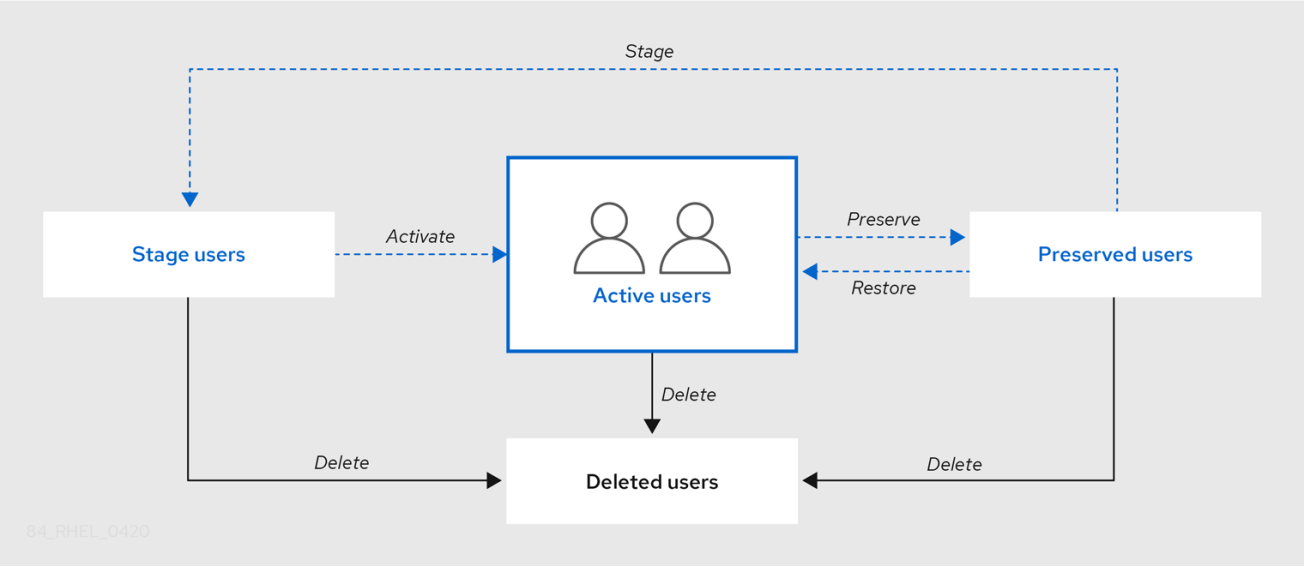
Suppression d'un compte d'utilisateur

Si l'employé est licencié, supprimez le compte sans sauvegarde.

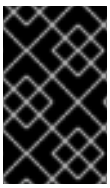
3.1. CYCLE DE VIE DE L'UTILISATEUR

La gestion des identités (IdM) prend en charge trois états de compte utilisateur :

- **Stage** les utilisateurs ne sont pas autorisés à s'authentifier. Il s'agit d'un état initial. Certaines propriétés du compte utilisateur requises pour les utilisateurs actifs ne peuvent pas être définies, par exemple l'appartenance à un groupe.
- **Active** les utilisateurs sont autorisés à s'authentifier. Toutes les propriétés requises du compte utilisateur doivent être définies dans cet état.
- **Preserved** sont d'anciens utilisateurs actifs qui sont considérés comme inactifs et ne peuvent pas s'authentifier auprès de l'IdM. Les utilisateurs préservés conservent la plupart des propriétés du compte qu'ils avaient en tant qu'utilisateurs actifs, mais ils ne font partie d'aucun groupe d'utilisateurs.



Vous pouvez supprimer définitivement les entrées utilisateur de la base de données IdM.



IMPORTANT

Les comptes d'utilisateurs supprimés ne peuvent pas être restaurés. Lorsque vous supprimez un compte d'utilisateur, toutes les informations associées à ce compte sont définitivement perdues.

Un nouvel administrateur ne peut être créé que par un utilisateur disposant de droits d'administrateur, tel que l'utilisateur admin par défaut. Si vous supprimez accidentellement tous les comptes d'administrateur, le gestionnaire de répertoire doit créer manuellement un nouvel administrateur dans le serveur de répertoire.

AVERTISSEMENT

Ne pas supprimer l'utilisateur **admin**. Comme **admin** est un utilisateur prédéfini requis par l'IdM, cette opération pose des problèmes avec certaines commandes. Si vous souhaitez définir et utiliser un autre utilisateur administrateur, désactivez l'utilisateur prédéfini **admin** avec **ipa user-disable admin** après avoir accordé des droits d'administrateur à au moins un autre utilisateur.

AVERTISSEMENT

N'ajoutez pas d'utilisateurs locaux à IdM. Le commutateur de service de noms (NSS) résout toujours les utilisateurs et les groupes IdM avant de résoudre les utilisateurs et les groupes locaux. Cela signifie, par exemple, que l'appartenance à un groupe IdM ne fonctionne pas pour les utilisateurs locaux.

3.2. AJOUTER DES UTILISATEURS DANS L'INTERFACE WEB

En général, vous devez créer un nouveau compte utilisateur avant qu'un nouvel employé ne commence à travailler. Ce compte d'étape n'est pas accessible et vous devez l'activer ultérieurement.

NOTE

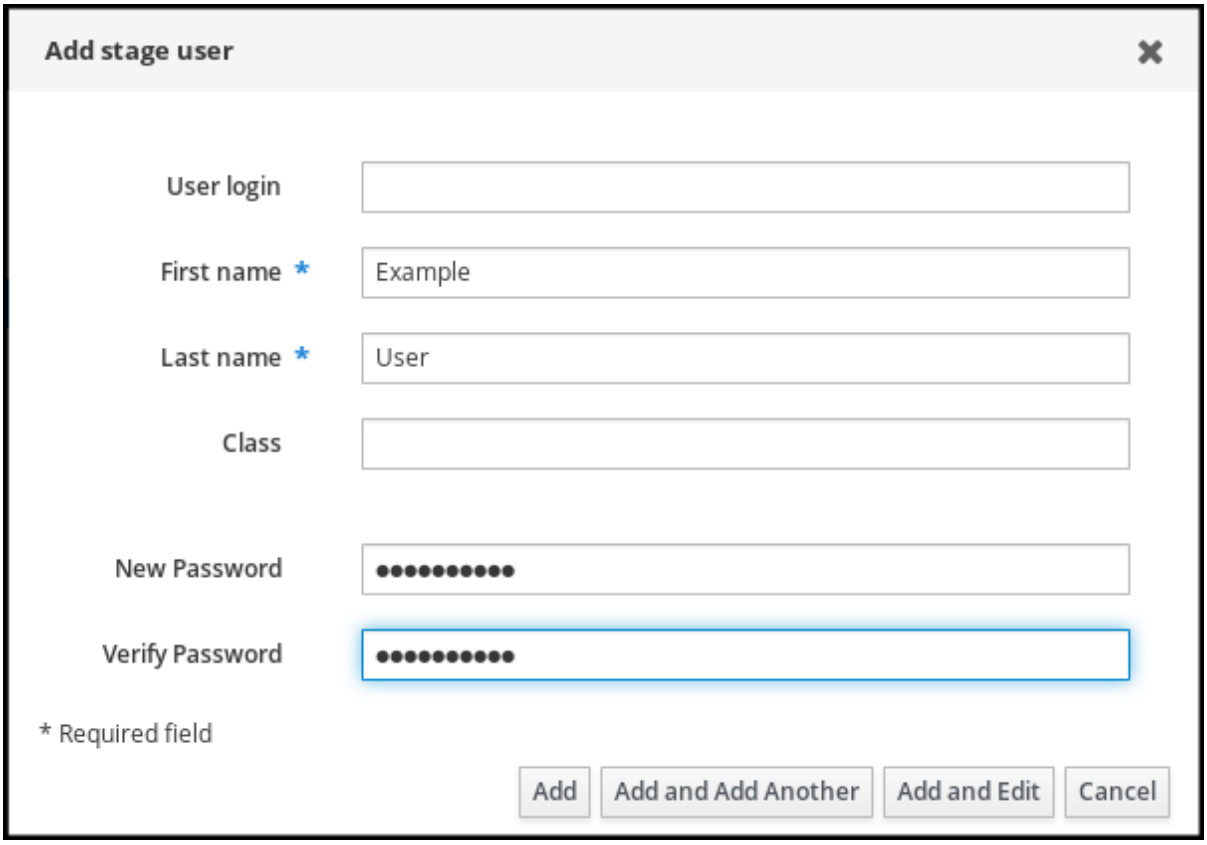
Vous pouvez également créer directement un compte d'utilisateur actif. Pour ajouter un utilisateur actif, suivez la procédure ci-dessous et ajoutez le compte utilisateur dans l'onglet **Active users**.

Conditions préalables

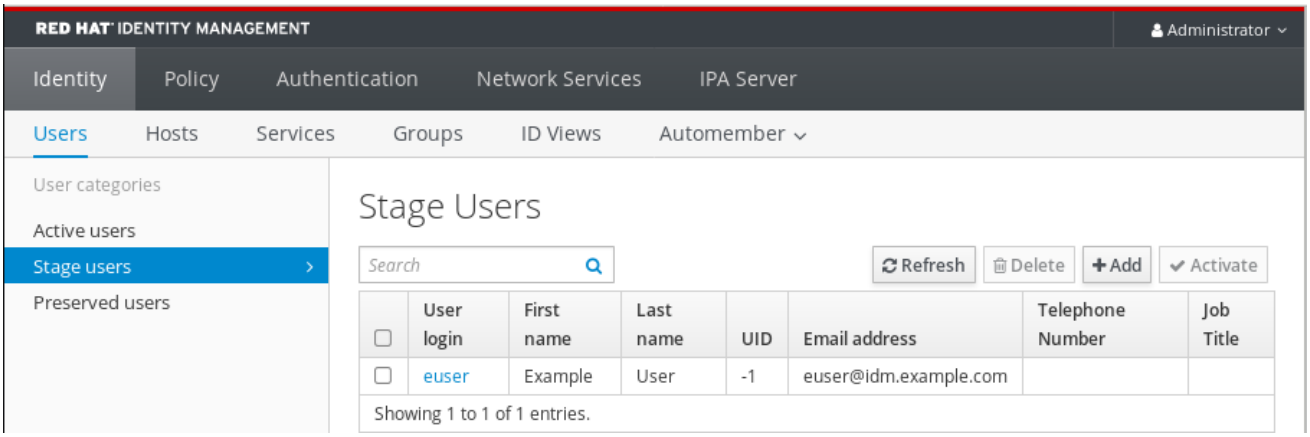
- Privilèges d'administrateur pour la gestion de l'IdM ou rôle d'administrateur des utilisateurs.

Procédure

1. Connectez-vous à l'interface Web IdM.
2. Allez sur l'onglet **Users → Stage Users**.
   Vous pouvez également ajouter le compte d'utilisateur sur le site **Users → Active users**, mais vous ne pouvez pas ajouter de groupes d'utilisateurs à ce compte.
3. Cliquez sur l'icône **Add**.
4. Dans la boîte de dialogue **Add stage user**, entrez **First name** et **Last name** du nouvel utilisateur.
5. [Facultatif] Dans le champ **User login**, ajoutez un nom de connexion.
   Si vous laissez ce champ vide, le serveur IdM crée le nom de connexion selon le modèle suivant : La première lettre du prénom et le nom de famille. Le nom de connexion complet peut comporter jusqu'à 32 caractères.
6. [Dans le menu déroulant GID, sélectionnez les groupes dans lesquels l'utilisateur doit être inclus.
7. [Facultatif] Dans les champs **Password** et **Verify password**, saisissez votre mot de passe et confirmez-le, en veillant à ce qu'il corresponde à votre mot de passe.
8. Cliquez sur le bouton **Add**.



À ce stade, vous pouvez voir le compte d'utilisateur dans la table **Stage Users**.



NOTE

Si vous cliquez sur le nom de l'utilisateur, vous pouvez modifier les paramètres avancés, tels que l'ajout d'un numéro de téléphone, d'une adresse ou d'une profession.

3.3. ACTIVATION DES UTILISATEURS D'ÉTAPE DANS L'INTERFACE WEB IDM

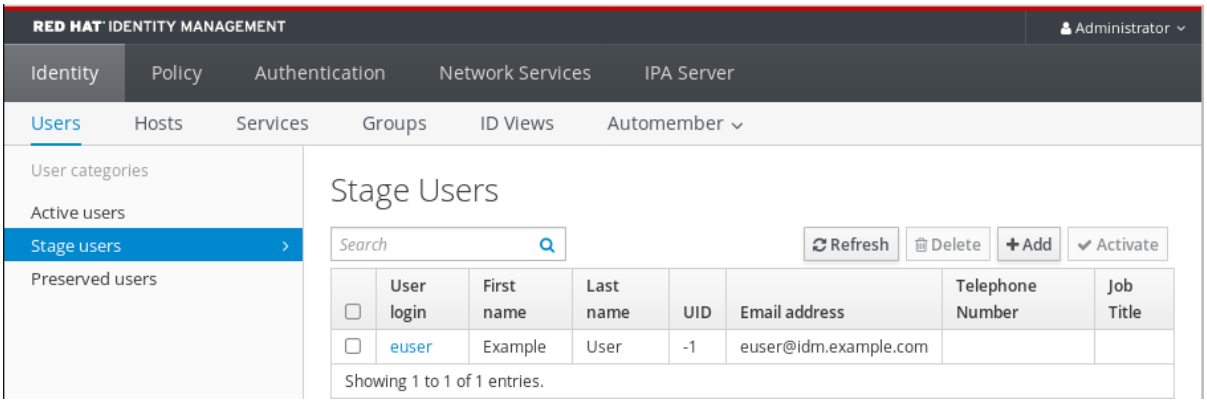
Un compte d'utilisateur de scène doit être activé avant que l'utilisateur puisse se connecter à IdM et avant qu'il puisse être ajouté à un groupe IdM. Cette section décrit comment activer les comptes d'utilisateurs de scène.

Conditions préalables

- Privilèges d'administrateur pour la gestion de l'interface Web IdM ou rôle d'administrateur des utilisateurs.
- Au moins un compte d'utilisateur en phase dans l'IdM.

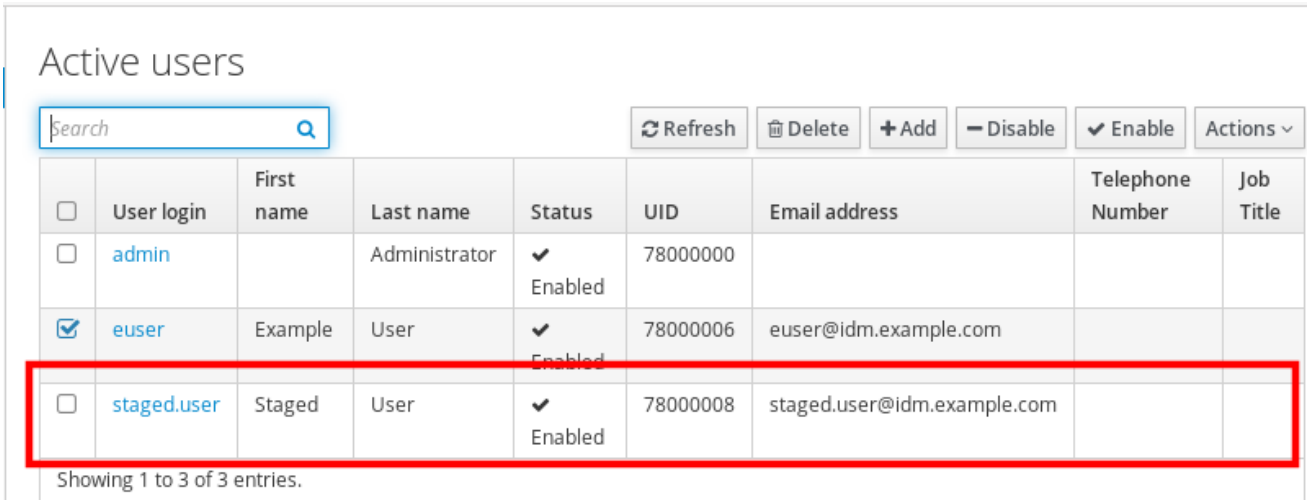
Procédure

1. Connectez-vous à l'interface Web IdM.
2. Allez sur l'onglet **Users → Stage users**.
3. Cliquez sur la case à cocher du compte utilisateur que vous souhaitez activer.
4. Cliquez sur le bouton **Activate**.



5. Dans la boîte de dialogue **Confirmation**, cliquez sur le bouton **OK**.

Si l'activation est réussie, l'interface Web IdM affiche une confirmation verte indiquant que l'utilisateur a été activé et que le compte utilisateur a été déplacé vers **Active users**. Le compte est actif et l'utilisateur peut s'authentifier auprès du domaine IdM et de l'interface Web IdM. L'utilisateur est invité à modifier son mot de passe lors de la première connexion.



NOTE

À ce stade, vous pouvez ajouter le compte d'utilisateur actif aux groupes d'utilisateurs.

3.4. DÉSACTIVATION DES COMPTES D'UTILISATEURS DANS L'INTERFACE WEB

Vous pouvez désactiver les comptes d'utilisateurs actifs. La désactivation d'un compte d'utilisateur désactive le compte. Par conséquent, les comptes d'utilisateur ne peuvent pas être utilisés pour s'authentifier et utiliser les services IdM, tels que Kerberos, ou pour effectuer des tâches.

Les comptes d'utilisateurs désactivés existent toujours dans l'IdM et toutes les informations qui leur sont associées restent inchangées. Contrairement aux comptes d'utilisateurs préservés, les comptes d'utilisateurs désactivés restent actifs et peuvent être membres de groupes d'utilisateurs.

NOTE

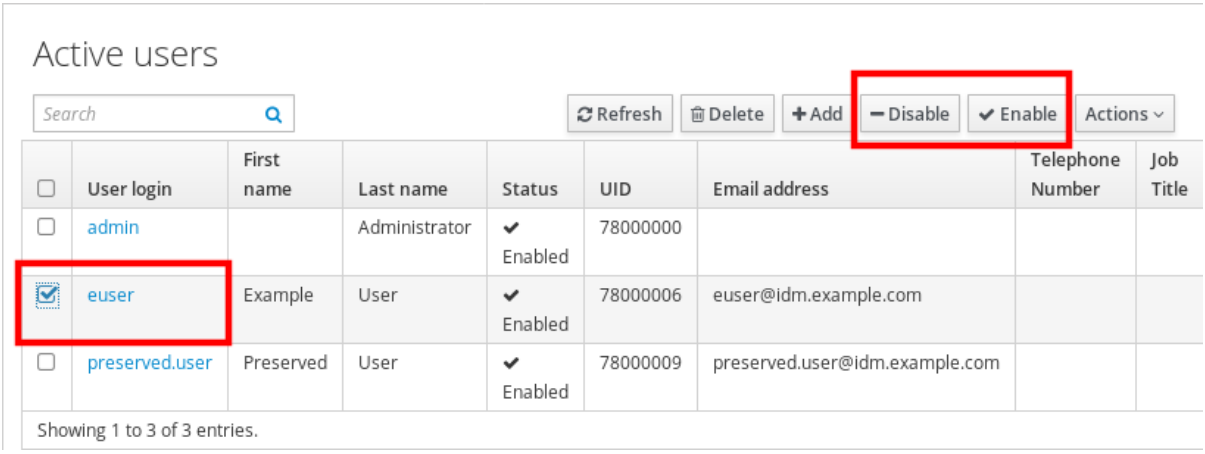
Après la désactivation d'un compte d'utilisateur, les connexions existantes restent valables jusqu'à l'expiration du TGT Kerberos et des autres tickets de l'utilisateur. Après l'expiration du ticket, l'utilisateur ne pourra pas le renouveler.

Conditions préalables

- Privilèges d'administrateur pour la gestion de l'interface Web IdM ou rôle d'administrateur des utilisateurs.

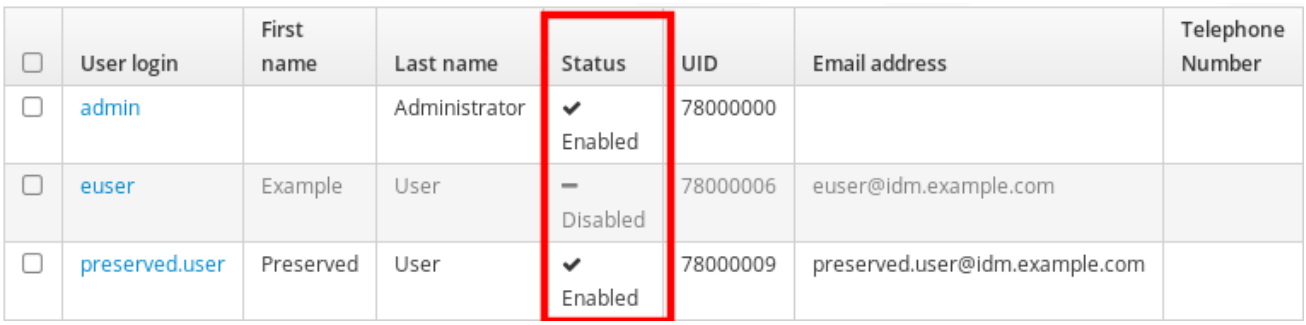
Procédure

1. Connectez-vous à l'interface Web IdM.
2. Allez sur l'onglet **Users → Active users**.
3. Cliquez sur la case à cocher des comptes d'utilisateurs que vous souhaitez désactiver.
4. Cliquez sur le bouton **Disable**.



5. Dans la boîte de dialogue **Confirmation**, cliquez sur le bouton **OK**.

Si la procédure de désactivation a réussi, vous pouvez le vérifier dans la colonne Statut du tableau **Active users**.



3.5. ACTIVATION DES COMPTES D'UTILISATEURS DANS L'INTERFACE WEB

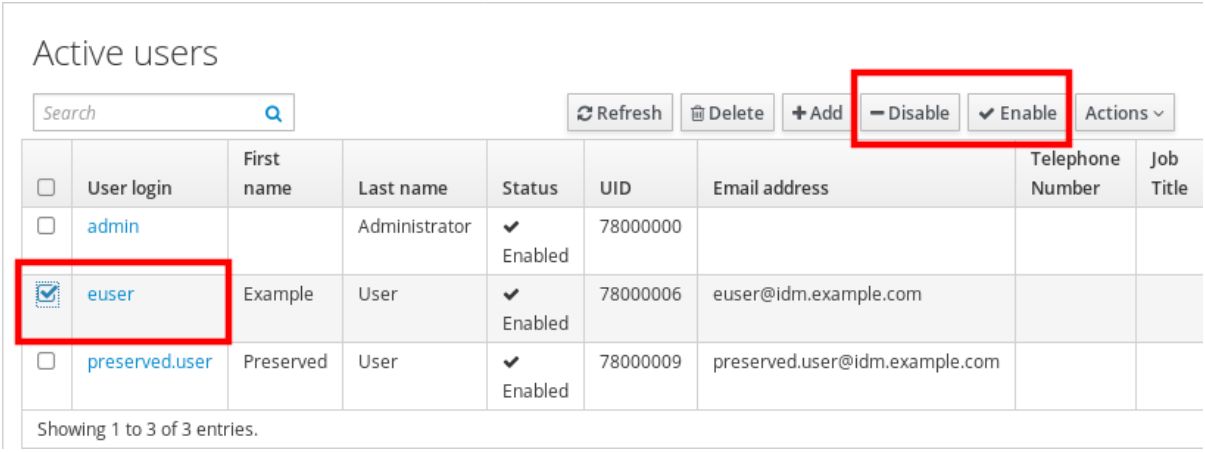
L'IdM permet d'activer des comptes d'utilisateurs actifs désactivés. L'activation d'un compte d'utilisateur active le compte désactivé.

Conditions préalables

- Privilèges d'administrateur pour la gestion de l'interface Web IdM ou rôle d'administrateur des utilisateurs.

Procédure

1. Connectez-vous à l'interface Web IdM.
2. Allez sur l'onglet **Users → Active users**.
3. Cliquez sur la case à cocher des comptes d'utilisateurs que vous souhaitez activer.
4. Cliquez sur le bouton **Enable**.



5. Dans la boîte de dialogue **Confirmation**, cliquez sur le bouton **OK**.

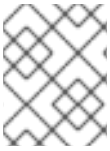
Si la modification a été effectuée avec succès, vous pouvez le vérifier dans la colonne Statut du tableau **Active users**.

3.6. PRÉSERVATION DES UTILISATEURS ACTIFS DANS L'INTERFACE WEB IDM

La préservation des comptes d'utilisateurs vous permet de supprimer des comptes de l'onglet **Active users**, tout en conservant ces comptes dans l'IdM.

Conservez le compte utilisateur si l'employé quitte l'entreprise. Si vous souhaitez désactiver les comptes d'utilisateur pendant quelques semaines ou quelques mois (congé parental, par exemple), désactivez le compte. Pour plus d'informations, voir Désactivation des comptes d'utilisateurs dans l'interface Web. Les comptes conservés ne sont pas actifs et les utilisateurs ne peuvent pas les utiliser pour accéder à votre réseau interne, mais le compte reste dans la base de données avec toutes les données.

Vous pouvez remettre les comptes restaurés en mode actif.



NOTE

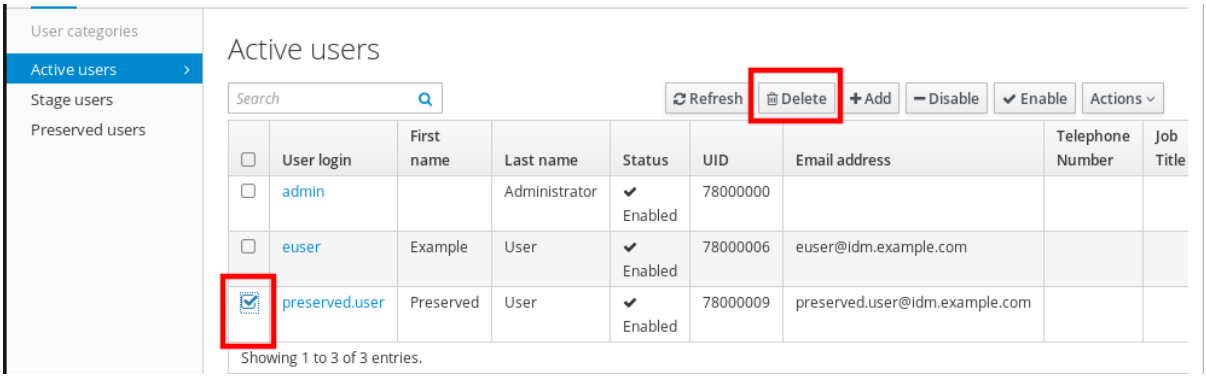
La liste des utilisateurs dans l'état préservé peut fournir un historique des anciens comptes d'utilisateurs.

Conditions préalables

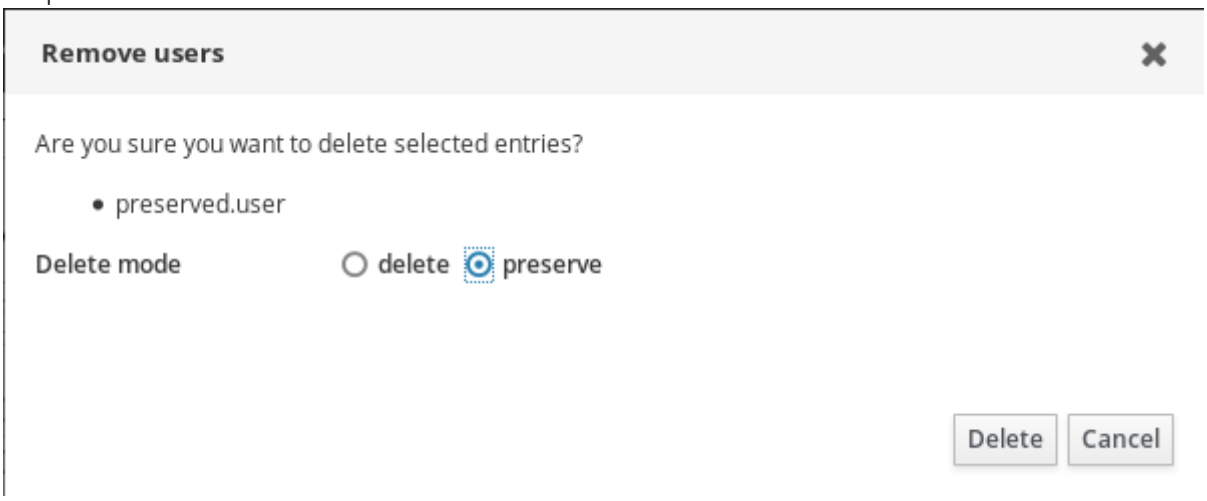
- Privilèges d'administrateur pour la gestion de l'interface Web IdM (Identity Management) ou rôle d'administrateur des utilisateurs.

Procédure

1. Connectez-vous à l'interface Web IdM.
2. Allez sur l'onglet **Users → Active users**.
3. Cliquez sur la case à cocher des comptes d'utilisateurs que vous souhaitez conserver.
4. Cliquez sur le bouton **Delete**.



5. Dans la boîte de dialogue **Remove users**, remplacez le bouton radio **Delete mode** par **preserve**.
6. Cliquez sur le bouton **Delete**.



En conséquence, le compte d'utilisateur est déplacé vers **Preserved users**.

Si vous devez restaurer les utilisateurs conservés, consultez la section Restauration des utilisateurs dans l'interface Web IdM.

3.7. RESTAURATION DES UTILISATEURS DANS L'INTERFACE WEB IDM

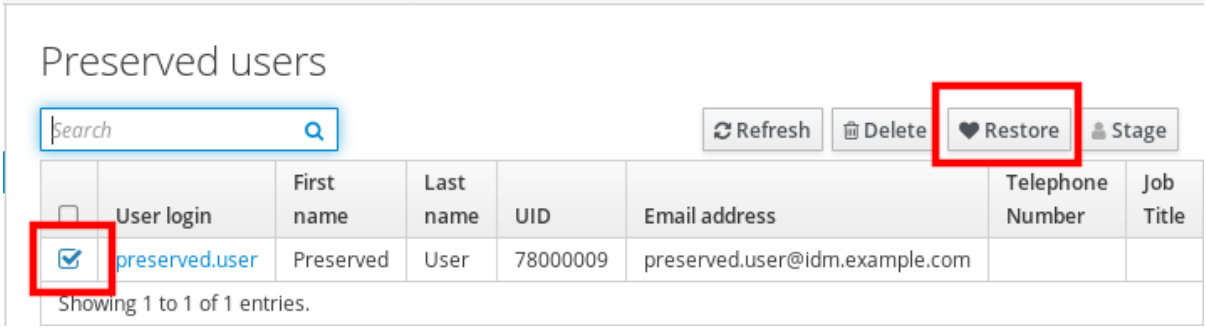
IdM (Identity Management) vous permet de restaurer les comptes d'utilisateurs préservés à l'état actif. Vous pouvez restaurer un utilisateur préservé en tant qu'utilisateur actif ou en tant qu'utilisateur d'étape.

Conditions préalables

- Privilèges d'administrateur pour la gestion de l'interface Web IdM ou rôle d'administrateur des utilisateurs.

Procédure

1. Connectez-vous à l'interface Web IdM.
2. Allez sur l'onglet **Users → Preserved users**.
3. Cliquez sur la case à cocher des comptes d'utilisateurs que vous souhaitez restaurer.
4. Cliquez sur le bouton **Restore**.



5. Dans la boîte de dialogue **Confirmation**, cliquez sur le bouton **OK**.

L'interface Web IdM affiche une confirmation verte et déplace les comptes d'utilisateurs vers l'onglet **Active users**.

3.8. SUPPRESSION D'UTILISATEURS DANS L'INTERFACE WEB IDM

La suppression d'utilisateurs est une opération irréversible, qui entraîne la suppression permanente des comptes d'utilisateurs de la base de données de l'IdM, y compris les appartenances à des groupes et les mots de passe. Toute configuration externe de l'utilisateur, telle que le compte système et le répertoire personnel, n'est pas supprimée, mais n'est plus accessible par l'intermédiaire de l'IdM.

Vous pouvez supprimer :

- Utilisateurs actifs - l'interface Web de l'IdM vous offre les options suivantes :
  - Préservation temporaire des utilisateurs
    Pour plus de détails, voir la section Préserver les utilisateurs actifs dans l'interface Web IdM .
  - Suppression définitive
- Utilisateurs de l'étape - vous pouvez supprimer définitivement les utilisateurs de l'étape.
- Utilisateurs conservés - vous pouvez supprimer définitivement les utilisateurs conservés.

La procédure suivante décrit la suppression des utilisateurs actifs. De même, vous pouvez supprimer des comptes d'utilisateurs sur :

- L'onglet **Stage users**
- L'onglet **Preserved users**

Conditions préalables

- Privilèges d'administrateur pour la gestion de l'interface Web IdM ou rôle d'administrateur des utilisateurs.

Procédure

1. Connectez-vous à l'interface Web IdM.
2. Allez sur l'onglet **Users → Active users**.
   Vous pouvez également supprimer le compte d'utilisateur sur le site **Users → Stage users** ou **Users → Preserved users**.
3. Cliquez sur l'icône **Delete**.
4. Dans la boîte de dialogue **Remove users**, remplacez le bouton radio **Delete mode** par **delete**.
5. Cliquez sur le bouton **Delete**.

Les comptes des utilisateurs ont été définitivement supprimés de l'IdM.

CHAPITRE 4. GÉRER LES COMPTES D'UTILISATEURS À L'AIDE DE PLAYBOOKS ANSIBLE

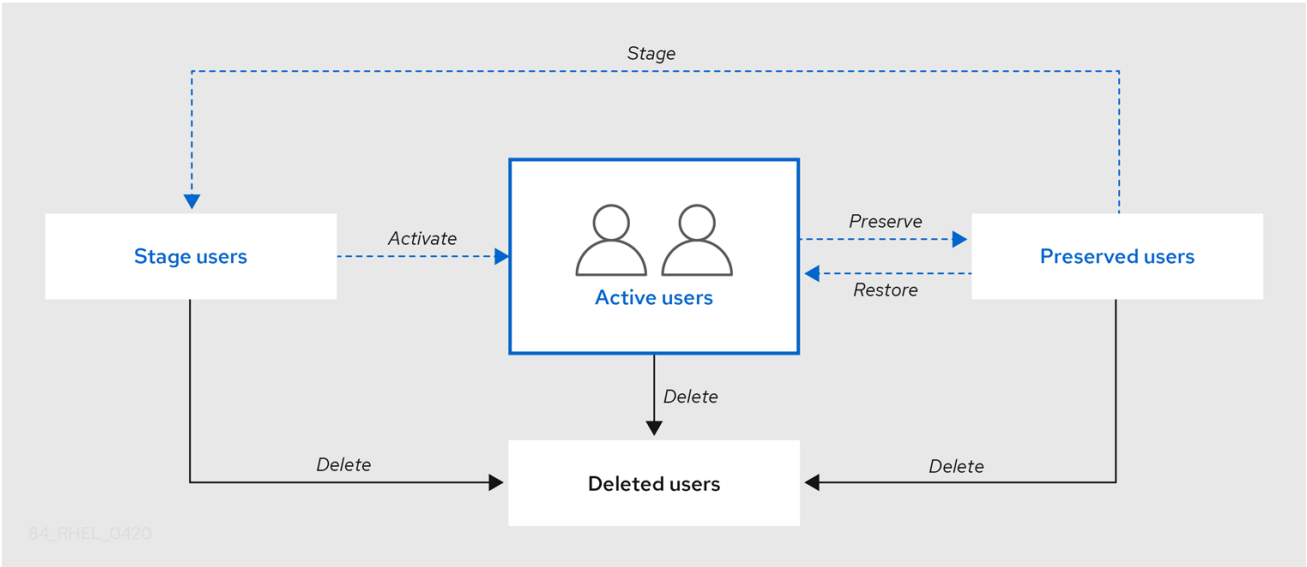
Vous pouvez gérer les utilisateurs dans IdM à l'aide de carnets de commande Ansible. Après avoir présenté le cycle de vie des utilisateurs, ce chapitre décrit comment utiliser les playbooks Ansible pour les opérations suivantes :

- Assurer la présence d'un seul utilisateur répertorié directement dans le fichier **YML**.
- Assurer la présence de plusieurs utilisateurs listés directement dans le fichier **YML**.
- Assurer la présence de plusieurs utilisateurs répertoriés dans un fichier **JSON** référencé à partir du fichier **YML**.
- Garantir l'absence d'utilisateurs listés directement dans le fichier **YML**.

4.1. CYCLE DE VIE DE L'UTILISATEUR

La gestion des identités (IdM) prend en charge trois états de compte utilisateur :

- **Stage** les utilisateurs ne sont pas autorisés à s'authentifier. Il s'agit d'un état initial. Certaines propriétés du compte utilisateur requises pour les utilisateurs actifs ne peuvent pas être définies, par exemple l'appartenance à un groupe.
- **Active** les utilisateurs sont autorisés à s'authentifier. Toutes les propriétés requises du compte utilisateur doivent être définies dans cet état.
- **Preserved** sont d'anciens utilisateurs actifs qui sont considérés comme inactifs et ne peuvent pas s'authentifier auprès de l'IdM. Les utilisateurs préservés conservent la plupart des propriétés du compte qu'ils avaient en tant qu'utilisateurs actifs, mais ils ne font partie d'aucun groupe d'utilisateurs.



Vous pouvez supprimer définitivement les entrées utilisateur de la base de données IdM.

IMPORTANT

Les comptes d'utilisateurs supprimés ne peuvent pas être restaurés. Lorsque vous supprimez un compte d'utilisateur, toutes les informations associées à ce compte sont définitivement perdues.

Un nouvel administrateur ne peut être créé que par un utilisateur disposant de droits d'administrateur, tel que l'utilisateur admin par défaut. Si vous supprimez accidentellement tous les comptes d'administrateur, le gestionnaire de répertoire doit créer manuellement un nouvel administrateur dans le serveur de répertoire.

AVERTISSEMENT

Ne pas supprimer l'utilisateur **admin**. Comme **admin** est un utilisateur prédéfini requis par l'IdM, cette opération pose des problèmes avec certaines commandes. Si vous souhaitez définir et utiliser un autre utilisateur administrateur, désactivez l'utilisateur prédéfini **admin** avec **ipa user-disable admin** après avoir accordé des droits d'administrateur à au moins un autre utilisateur.

AVERTISSEMENT

N'ajoutez pas d'utilisateurs locaux à IdM. Le commutateur de service de noms (NSS) résout toujours les utilisateurs et les groupes IdM avant de résoudre les utilisateurs et les groupes locaux. Cela signifie, par exemple, que l'appartenance à un groupe IdM ne fonctionne pas pour les utilisateurs locaux.

4.2. ASSURER LA PRÉSENCE D'UN UTILISATEUR IDM À L'AIDE D'UN PLAYBOOK ANSIBLE

La procédure suivante décrit comment assurer la présence d'un utilisateur dans IdM à l'aide d'un playbook Ansible.

Conditions préalables

- Vous connaissez le mot de passe de l'IdM **admin**.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.

- L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

Procédure

1. Créez un fichier d'inventaire, par exemple **inventory.file**, et définissez-y **ipaserver**:

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

2. Créez un fichier Ansible playbook avec les données de l'utilisateur dont vous voulez assurer la présence dans IdM. Pour simplifier cette étape, vous pouvez copier et modifier l'exemple dans le fichier **/usr/share/doc/ansible-freeipa/playbooks/user/add-user.yml**. Par exemple, pour créer un utilisateur nommé *idm_user* et ajouter *Password123* comme mot de passe :

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

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Create user idm_user
    ipauser:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: idm_user
      first: Alice
      last: Acme
      uid: 1000111
      gid: 10011
      phone: "+555123457"
      email: idm_user@acme.com
      passwordexpiration: "2023-01-19 23:59:59"
      password: "Password123"
      update_password: on_create
```

Vous devez utiliser les options suivantes pour ajouter un utilisateur :

- **name**le nom d'utilisateur
- **first**: la chaîne de caractères du prénom
- **last**: la chaîne du nom de famille

Pour la liste complète des options disponibles pour l'utilisateur, voir le fichier Markdown de **/usr/share/doc/ansible-freeipa/README-user.md**.

NOTE

Si vous utilisez l'option **update_password: on_create**, Ansible ne crée le mot de passe de l'utilisateur que lorsqu'il crée l'utilisateur. Si l'utilisateur est déjà créé avec un mot de passe, Ansible ne génère pas de nouveau mot de passe.

3. Exécutez le manuel de jeu :

```
$ ansible-playbook --vault-password-file=password_file -v -i
path_to_inventory_directory/inventory.file path_to_playbooks_directory/add-IdM-
user.yml
```

Verification steps

- Vous pouvez vérifier si le nouveau compte d'utilisateur existe dans IdM en utilisant la commande **ipa user-show**:

1. Connectez-vous à **ipaserver** en tant qu'administrateur :

```
$ ssh admin@server.idm.example.com
Password:
[admin@server /]$
```

2. Demander un ticket Kerberos pour l'administrateur :

```
$ kinit admin
Password for admin@IDM.EXAMPLE.COM:
```

3. Demande d'informations sur *idm_user*:

```
$ ipa user-show idm_user
  User login: idm_user
  First name: Alice
  Last name: Acme
  ....
```

L'utilisateur nommé *idm_user* est présent dans IdM.

4.3. ASSURER LA PRÉSENCE DE PLUSIEURS UTILISATEURS IDM À L'AIDE DE PLAYBOOKS ANSIBLE

La procédure suivante décrit comment assurer la présence de plusieurs utilisateurs dans IdM à l'aide d'un playbook Ansible.

Conditions préalables

- Vous connaissez le mot de passe de l'IdM **admin**.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

Procédure

1. Créez un fichier d'inventaire, par exemple **inventory.file**, et définissez-y **ipaserver**:

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

2. Créez un fichier Ansible playbook avec les données des utilisateurs dont vous voulez assurer la présence dans IdM. Pour simplifier cette étape, vous pouvez copier et modifier l'exemple dans le fichier **/usr/share/doc/ansible-freeipa/playbooks/user/ensure-users-present.yml**. Par exemple, pour créer les utilisateurs *idm_user_1*, *idm_user_2*, et *idm_user_3*, et ajouter *Password123* comme mot de passe de *idm_user_1*:

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

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Create user idm_users
    ipauser:
      ipaadmin_password: "{{ ipaadmin_password }}"
      users:
      - name: idm_user_1
        first: Alice
        last: Acme
        uid: 10001
        gid: 10011
        phone: "+555123457"
        email: idm_user@acme.com
        passwordexpiration: "2023-01-19 23:59:59"
        password: "Password123"
      - name: idm_user_2
        first: Bob
        last: Acme
        uid: 100011
        gid: 10011
      - name: idm_user_3
        first: Eve
        last: Acme
        uid: 1000111
        gid: 10011
```

NOTE

Si vous ne spécifiez pas l'option **update_password: on_create**, Ansible réinitialise le mot de passe de l'utilisateur à chaque fois que le livre de jeu est exécuté : si l'utilisateur a modifié le mot de passe depuis la dernière fois que le livre de jeu a été exécuté, Ansible réinitialise le mot de passe.

3. Exécutez le manuel de jeu :

```
$ ansible-playbook --vault-password-file=password_file -v -i
path_to_inventory_directory/inventory.file path_to_playbooks_directory/add-
users.yml
```

Verification steps

- Vous pouvez vérifier si le compte d'utilisateur existe dans IdM en utilisant la commande **ipa user-show**:

1. Connectez-vous à **ipaserver** en tant qu'administrateur :

```
$ ssh administrator@server.idm.example.com
Password:
[admin@server /]$
```

2. Afficher des informations sur *idm_user_1*:

```
$ ipa user-show idm_user_1
  User login: idm_user_1
  First name: Alice
  Last name: Acme
  Password: True
  ....
```

L'utilisateur nommé *idm_user_1* est présent dans IdM.

4.4. ASSURER LA PRÉSENCE DE PLUSIEURS UTILISATEURS IDM À PARTIR D'UN FICHIER JSON EN UTILISANT LES PLAYBOOKS ANSIBLE

La procédure suivante décrit comment assurer la présence de plusieurs utilisateurs dans IdM à l'aide d'un playbook Ansible. Les utilisateurs sont stockés dans un fichier **JSON**.

Conditions préalables

- Vous connaissez le mot de passe de l'IdM **admin**.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

Procédure

1. Créez un fichier d'inventaire, par exemple **inventory.file**, et définissez-y **ipaserver**:

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

2. Créez un fichier playbook Ansible avec les tâches nécessaires. Référencez le fichier **JSON** avec les données des utilisateurs dont vous voulez assurer la présence. Pour simplifier cette étape, vous pouvez copier et modifier l'exemple dans le fichier **/usr/share/doc/ansible-freeipa/ensure-users-present-ymlfile.yml**:

```
---
- name: Ensure users' presence
  hosts: ipaserver

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Include users.json
    include_vars:
      file: users.json

  - name: Users present
    ipauser:
      ipaadmin_password: "{{ ipaadmin_password }}"
      users: "{{ users }}"
```

3. Créez le fichier **users.json** et ajoutez-y les utilisateurs IdM. Pour simplifier cette étape, vous pouvez copier et modifier l'exemple du fichier **/usr/share/doc/ansible-freeipa/playbooks/user/users.json**. Par exemple, pour créer les utilisateurs *idm_user_1*, *idm_user_2*, et *idm_user_3*, et ajouter *Password123* comme mot de passe de *idm_user_1*:

```
{
  "users": [
    {
      "name": "idm_user_1",
      "first": "Alice",
      "last": "Acme",
      "password": "Password123"
    },
    {
      "name": "idm_user_2",
      "first": "Bob",
      "last": "Acme"
    },
    {
      "name": "idm_user_3",
      "first": "Eve",
      "last": "Acme"
    }
  ]
}
```

4. Exécutez le playbook Ansible. Spécifiez le fichier du livre de jeu, le fichier contenant le mot de passe protégeant le fichier **secret.yml** et le fichier d'inventaire :

```
$ ansible-playbook --vault-password-file=password_file -v -i
path_to_inventory_directory/inventory.file path_to_playbooks_directory/ensure-users-
present-jsonfile.yml
```

Verification steps

- Vous pouvez vérifier si les comptes d'utilisateurs sont présents dans IdM à l'aide de la commande **ipa user-show**:

1. Connectez-vous à **ipaserver** en tant qu'administrateur :

```
$ ssh administrator@server.idm.example.com
Password:
[admin@server /]$
```

2. Afficher des informations sur *idm_user_1*:

```
$ ipa user-show idm_user_1
  User login: idm_user_1
  First name: Alice
  Last name: Acme
  Password: True
  ....
```

L'utilisateur nommé *idm_user_1* est présent dans IdM.

4.5. ASSURER L'ABSENCE D'UTILISATEURS UTILISANT DES PLAYBOOKS ANSIBLE

La procédure suivante décrit comment utiliser un playbook Ansible pour s'assurer que des utilisateurs spécifiques sont absents de l'IdM.

Conditions préalables

- Vous connaissez le mot de passe de l'IdM **admin**.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

Procédure

1. Créez un fichier d'inventaire, par exemple **inventory.file**, et définissez-y **ipaserver**:

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

2. Créez un fichier playbook Ansible avec les utilisateurs dont vous voulez garantir l'absence d'IdM. Pour simplifier cette étape, vous pouvez copier et modifier l'exemple dans le fichier **/usr/share/doc/ansible-freeipa/playbooks/user/ensure-users-present.yml**. Par exemple, pour supprimer les utilisateurs *idm_user_1*, *idm_user_2*, et *idm_user_3*:

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

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Delete users idm_user_1, idm_user_2, idm_user_3
    ipauser:
      ipaadmin_password: "{{ ipaadmin_password }}"
      users:
      - name: idm_user_1
      - name: idm_user_2
      - name: idm_user_3
      state: absent
```

3. Exécutez le playbook Ansible. Spécifiez le fichier du livre de jeu, le fichier contenant le mot de passe protégeant le fichier **secret.yml** et le fichier d'inventaire :

```
$ ansible-playbook --vault-password-file=password_file -v -i
path_to_inventory_directory/inventory.file path_to_playbooks_directory/delete-
users.yml
```

Verification steps

Vous pouvez vérifier que les comptes d'utilisateurs n'existent pas dans IdM en utilisant la commande **ipa user-show**:

1. Connectez-vous à **ipaserver** en tant qu'administrateur :

```
$ ssh administrator@server.idm.example.com
Password:
[admin@server /]$
```

2. Demande d'informations sur *idm_user_1*:

```
$ ipa user-show idm_user_1
ipa: ERROR: idm_user_1: user not found
```

L'utilisateur nommé *idm_user_1* n'existe pas dans IdM.

4.6. RESSOURCES SUPPLÉMENTAIRES

- Voir le fichier Markdown de **README-user.md** dans le répertoire **/usr/share/doc/ansible-freeipa/**.

- Voir les exemples de playbooks Ansible dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/user**.

CHAPITRE 5. GESTION DES MOTS DE PASSE DES UTILISATEURS DANS L'IDM

5.1. QUI PEUT MODIFIER LES MOTS DE PASSE DES UTILISATEURS DE L'IDM ET COMMENT ?

Les utilisateurs réguliers qui ne sont pas autorisés à modifier les mots de passe d'autres utilisateurs ne peuvent modifier que leur propre mot de passe. Le nouveau mot de passe doit être conforme aux politiques de mot de passe de l'IdM applicables aux groupes dont l'utilisateur est membre. Pour plus de détails sur la configuration des politiques de mot de passe, voir Définir les politiques de mot de passe de l'IdM.

Les administrateurs et les utilisateurs ayant le droit de modifier les mots de passe peuvent définir des mots de passe initiaux pour les nouveaux utilisateurs et réinitialiser les mots de passe des utilisateurs existants. Ces mots de passe :

- Il n'est pas nécessaire de respecter les politiques de mot de passe de l'IdM.
- Expirent après la première connexion réussie. Dans ce cas, l'IdM invite l'utilisateur à modifier immédiatement le mot de passe expiré. Pour désactiver ce comportement, voir Activer la réinitialisation du mot de passe dans IdM sans inviter l'utilisateur à changer de mot de passe lors de la prochaine connexion.

NOTE

L'utilisateur du LDAP Directory Manager (DM) peut modifier les mots de passe des utilisateurs à l'aide des outils LDAP. Le nouveau mot de passe peut remplacer toute politique de mot de passe de l'IdM. Les mots de passe définis par DM n'expirent pas après la première connexion.

5.2. MODIFICATION DU MOT DE PASSE DE L'UTILISATEUR DANS L'INTERFACE WEB DE L'IDM

En tant qu'utilisateur de la gestion des identités (IdM), vous pouvez modifier votre mot de passe d'utilisateur dans l'interface Web IdM.

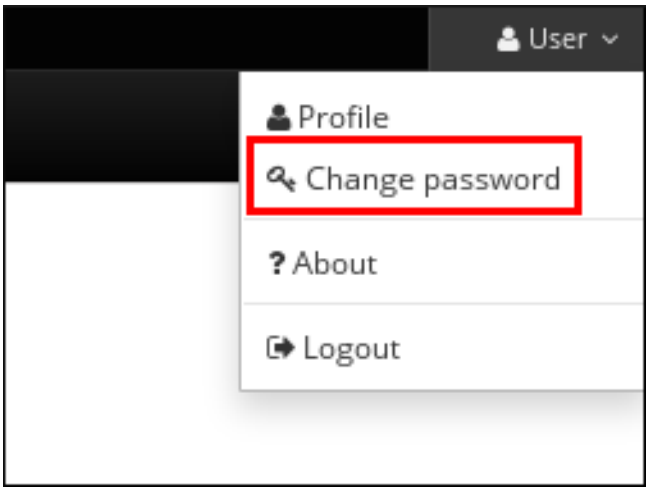
Conditions préalables

- Vous êtes connecté à l'interface Web IdM.

Procédure

1. Dans le coin supérieur droit, cliquez sur **User name → Change password**

Figure 5.1. Réinitialisation du mot de passe



2. Saisissez le mot de passe actuel et le nouveau mot de passe.

5.3. RÉINITIALISATION DU MOT DE PASSE D'UN AUTRE UTILISATEUR DANS L'INTERFACE WEB IDM

En tant qu'utilisateur administratif de Identity Management (IdM), vous pouvez modifier les mots de passe d'autres utilisateurs dans l'interface Web IdM.

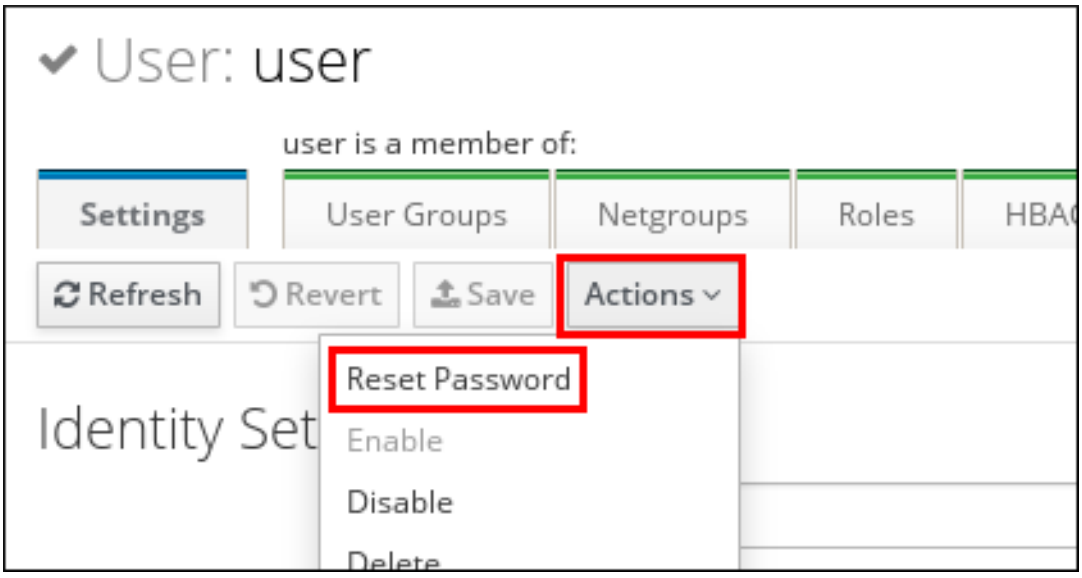
Conditions préalables

- Vous êtes connecté à l'interface Web IdM en tant qu'utilisateur administratif.

Procédure

1. Sélectionner **Identité → Users**.
2. Cliquez sur le nom de l'utilisateur à modifier.
3. Cliquez sur **Actions → Reset password**.

Figure 5.2. Réinitialisation du mot de passe



4. Saisissez le nouveau mot de passe et cliquez sur **Réinitialiser le mot de passe**.

Figure 5.3. Confirmation du nouveau mot de passe

5.4. RÉINITIALISATION DU MOT DE PASSE DE L'UTILISATEUR DU GESTIONNAIRE DE RÉPERTOIRE

Si vous perdez le mot de passe de l'Identity Management (IdM) Directory Manager, vous pouvez le réinitialiser.

Conditions préalables

- Vous avez un accès **root** à un serveur IdM.

Procédure

1. Générez un nouveau hachage de mot de passe à l'aide de la commande **pwdhash**. Par exemple :

```
# pwdhash -D /etc/dirsrv/slapd-IDM-EXAMPLE-COM password
{PBKDF2_SHA256}AAAgABU0bKhyjY53NcxY33ueoPjOUWtl4iyYN5uW...
```

   En spécifiant le chemin d'accès à la configuration du serveur d'annuaire, vous utilisez automatiquement le schéma de stockage du mot de passe défini dans l'attribut **nsslapd-rootpwstoragescheme** pour crypter le nouveau mot de passe.

2. Sur chaque serveur IdM de votre topologie, exécutez les étapes suivantes :

   a. Arrêter tous les services IdM installés sur le serveur :

```
# ipactl stop
```

   b. Modifiez le fichier **/etc/dirsrv/IDM-EXAMPLE-COM/dse.ldif** et attribuez à l'attribut **nsslapd-rootpw** la valeur générée par la commande **pwdhash**:

```
nsslapd-rootpw:
{PBKDF2_SHA256}AAAgABU0bKhyjY53NcxY33ueoPjOUWtl4iyYN5uW...
```

   c. Démarrer tous les services IdM installés sur le serveur :

```
# ipactl start
```

5.5. MODIFICATION DU MOT DE PASSE DE L'UTILISATEUR OU RÉINITIALISATION DU MOT DE PASSE D'UN AUTRE UTILISATEUR DANS L'INTERFACE CLI DE L'IDM

Vous pouvez modifier votre mot de passe utilisateur à l'aide de l'interface de ligne de commande (CLI) de la gestion des identités (IdM). Si vous êtes un utilisateur administratif, vous pouvez utiliser l'interface de ligne de commande pour réinitialiser le mot de passe d'un autre utilisateur.

Conditions préalables

- Vous avez obtenu un ticket d'attribution de ticket (TGT) pour un utilisateur IdM.
- Si vous réinitialisez le mot de passe d'un autre utilisateur, vous devez avoir obtenu un TGT pour un utilisateur administratif dans IdM.

Procédure

- Entrez la commande **ipa user-mod** avec le nom de l'utilisateur et l'option **--password**. La commande vous demandera le nouveau mot de passe.

```
$ ipa user-mod idm_user --password
Password:
Enter Password again to verify:
---------------------
Modified user "idm_user"
---------------------
...
```

NOTE

Vous pouvez également utiliser la commande **ipa passwd** *idm_user* au lieu de **ipa user-mod**.

5.6. PERMETTRE LA RÉINITIALISATION DU MOT DE PASSE DANS L'IDM SANS DEMANDER À L'UTILISATEUR DE CHANGER DE MOT DE PASSE LORS DE LA PROCHAINE CONNEXION

Par défaut, lorsqu'un administrateur réinitialise le mot de passe d'un autre utilisateur, le mot de passe expire après la première connexion réussie. En tant que gestionnaire de l'annuaire IdM, vous pouvez spécifier les privilèges suivants pour les administrateurs IdM individuels :

- Ils peuvent effectuer des opérations de changement de mot de passe sans exiger des utilisateurs qu'ils modifient leur mot de passe lors de leur première connexion.
- Ils peuvent contourner la politique en matière de mots de passe de sorte qu'aucune force ou historique n'est appliquée.

AVERTISSEMENT

Le contournement de la politique de mot de passe peut constituer une menace pour la sécurité. Faites preuve de prudence lorsque vous sélectionnez les utilisateurs auxquels vous accordez ces privilèges supplémentaires.

Conditions préalables

- Vous connaissez le mot de passe du gestionnaire de répertoire.

Procédure

1. Sur chaque serveur de gestion des identités (IdM) du domaine, effectuez les modifications suivantes :
   a. Entrez la commande **ldapmodify** pour modifier les entrées LDAP. Indiquez le nom du serveur IdM et le port 389, puis appuyez sur Entrée :

   ```
   $ ldapmodify -x -D "cn=Directory Manager" -W -h server.idm.example.com -p 389
   Enter LDAP Password:
   ```

   b. Saisissez le mot de passe du gestionnaire de répertoire.
   c. Saisissez le nom distinctif de l'entrée de synchronisation du mot de passe **ipa_pwd_extop** et appuyez sur Entrée :

   ```
   dn : cn=ipa_pwd_extop,cn=plugins,cn=config
   ```

   d. Spécifiez le type de modification **modify** et appuyez sur Entrée :

   ```
   changetype : modify
   ```

   e. Indiquez le type de modification que vous souhaitez que LDAP exécute et à quel attribut. Appuyez sur Entrée :

   ```
   ajouter : passSyncManagersDNs
   ```

   f. Spécifiez les comptes d'utilisateurs administratifs dans l'attribut **passSyncManagersDNs**. Cet attribut a plusieurs valeurs. Par exemple, pour accorder à l'utilisateur **admin** les pouvoirs de réinitialisation du mot de passe de Directory Manager :

   ```
   passSyncManagersDNs: \
   uid=admin,cn=users,cn=accounts,dc=example,dc=com
   ```

   g. Appuyez deux fois sur Enter pour arrêter la modification de l'entrée.

L'ensemble de la procédure se déroule comme suit :

```
$ ldapmodify -x -D "cn=Directory Manager" -W -h server.idm.example.com -p 389
```

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

L'utilisateur **admin**, répertorié sous **passSyncManagerDNs**, dispose désormais de privilèges supplémentaires.

5.7. VÉRIFIER SI LE COMPTE D'UN UTILISATEUR IDM EST VERROUILLÉ

En tant qu'administrateur de la gestion des identités (IdM), vous pouvez vérifier si le compte d'un utilisateur IdM est verrouillé. Pour ce faire, vous devez comparer le nombre maximal autorisé de tentatives de connexion infructueuses d'un utilisateur avec le nombre de connexions infructueuses réelles de l'utilisateur.

Conditions préalables

- Vous avez obtenu le ticket d'attribution de ticket (TGT) d'un utilisateur administratif dans IdM.

Procédure

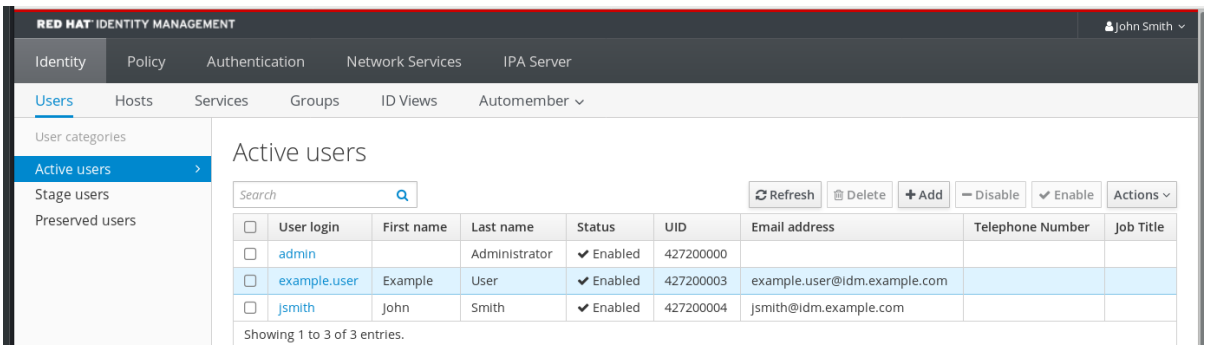
1. Affichez le statut du compte utilisateur pour connaître le nombre d'échecs de connexion :

```
$ ipa user-status example_user
-----------------------
Account disabled: False
-----------------------
  Server: idm.example.com
  Failed logins: 8
  Last successful authentication: N/A
  Last failed authentication: 20220229080317Z
  Time now: 2022-02-29T08:04:46Z
----------------------------
Number of entries returned 1
----------------------------
```

2. Affiche le nombre de tentatives de connexion autorisées pour un utilisateur donné :

   a. Se connecter à l'interface Web IdM en tant qu'administrateur IdM.

   b. Ouvrez l'onglet **Identity → Users → Active users**



a. Cliquez sur le nom de l'utilisateur pour ouvrir les paramètres de l'utilisateur.

b. Dans la section **Password policy**, localisez l'élément **Max failures**.

3. Comparez le nombre d'échecs de connexion affiché dans la sortie de la commande **ipa user-status** avec le nombre de **Max failures** affiché dans l'interface Web de l'IdM. Si le nombre d'échecs de connexion est égal au nombre maximum de tentatives de connexion autorisées, le compte de l'utilisateur est verrouillé.

Ressources supplémentaires

- Déverrouillage des comptes d'utilisateurs en cas d'échec du mot de passe dans l'IdM

5.8. DÉVERROUILLAGE DES COMPTES D'UTILISATEURS EN CAS D'ÉCHEC DU MOT DE PASSE DANS L'IDM

Si un utilisateur tente de se connecter en utilisant un mot de passe incorrect un certain nombre de fois, la gestion des identités (IdM) verrouille le compte de l'utilisateur, ce qui l'empêche de se connecter. Pour des raisons de sécurité, l'IdM n'affiche aucun message d'avertissement indiquant que le compte de l'utilisateur a été verrouillé. Au lieu de cela, l'invite CLI peut continuer à demander à l'utilisateur un mot de passe encore et encore.

L'IdM déverrouille automatiquement le compte d'utilisateur après un certain temps. Vous pouvez également déverrouiller le compte d'utilisateur manuellement en suivant la procédure suivante.

Conditions préalables

- Vous avez obtenu le ticket d'attribution de ticket d'un utilisateur administratif de l'IdM.

Procédure

- Pour déverrouiller un compte d'utilisateur, utilisez la commande **ipa user-unlock**.

```
$ ipa user-unlock idm_user
-----------------------
Unlocked account "idm_user"
-----------------------
```

L'utilisateur peut ensuite se reconnecter.

Ressources supplémentaires

- Vérifier si le compte d'un utilisateur IdM est verrouillé

5.9. ACTIVATION DU SUIVI DE LA DERNIÈRE AUTHENTIFICATION KERBEROS RÉUSSIE POUR LES UTILISATEURS DANS IDM

Pour des raisons de performance, Identity Management (IdM) fonctionnant dans Red Hat Enterprise Linux 8 ne stocke pas l'horodatage de la dernière authentification Kerberos réussie d'un utilisateur. Par conséquent, certaines commandes, telles que **ipa user-status**, n'affichent pas l'horodatage.

Conditions préalables

- Vous avez obtenu le ticket d'attribution de ticket (TGT) d'un utilisateur administratif dans IdM.
- Vous avez un accès **root** au serveur IdM sur lequel vous exécutez la procédure.

Procédure

1. Affiche les fonctions du plug-in de mot de passe actuellement activées :

```
# ipa config-show | grep "Password plugin features"
  Password plugin features: AllowNThash, KDC:Disable Last Success
```

   La sortie montre que le plug-in **KDC:Disable Last Success** est activé. Le plug-in masque la dernière tentative d'authentification Kerberos réussie dans la sortie ipa user-status.

2. Ajouter le paramètre **--ipaconfigstring=*feature*** le paramètre pour chaque fonctionnalité de la commande **ipa config-mod** actuellement activée, à l'exception de **KDC:Disable Last Success**:

```
# ipa config-mod --ipaconfigstring='AllowNThash'
```

   Cette commande n'active que le plug-in **AllowNThash**. Pour activer plusieurs fonctionnalités, spécifiez le paramètre **--ipaconfigstring=*feature*** séparément pour chaque fonctionnalité.

3. Redémarrer IdM :

```
# ipactl restart
```

CHAPITRE 6. DÉFINITION DES POLITIQUES DE MOT DE PASSE DE L'IDM

Ce chapitre décrit les politiques de mot de passe de la gestion des identités (IdM) et explique comment ajouter une nouvelle politique de mot de passe dans l'IdM à l'aide d'un playbook Ansible.

6.1. QU'EST-CE QU'UNE POLITIQUE DE MOT DE PASSE ?

Une politique de mot de passe est un ensemble de règles auxquelles les mots de passe doivent répondre. Par exemple, une politique de mot de passe peut définir la longueur minimale et la durée maximale d'un mot de passe. Tous les utilisateurs concernés par cette politique sont tenus de définir un mot de passe suffisamment long et de le modifier assez fréquemment pour satisfaire aux conditions spécifiées. De cette manière, les politiques de mot de passe contribuent à réduire le risque que quelqu'un découvre et utilise à mauvais escient le mot de passe d'un utilisateur.

6.2. POLITIQUES EN MATIÈRE DE MOTS DE PASSE DANS L'IDM

Les mots de passe sont le moyen le plus courant pour les utilisateurs de la gestion de l'identité (IdM) de s'authentifier auprès du domaine Kerberos IdM. Les stratégies de mot de passe définissent les exigences auxquelles doivent répondre les mots de passe des utilisateurs IdM.

NOTE

La politique de mot de passe de l'IdM est définie dans l'annuaire LDAP sous-jacent, mais c'est le centre de distribution de clés Kerberos (KDC) qui applique la politique de mot de passe.

Les attributs de la politique de mot de passe énumèrent les attributs que vous pouvez utiliser pour définir une politique de mot de passe dans IdM.

Tableau 6.1. Attributs de la politique relative aux mots de passe

Attribut	Explication	Exemple :
Durée de vie maximale	Durée maximale en jours pendant laquelle un mot de passe est valide avant que l'utilisateur ne doive le réinitialiser.	Durée de vie maximale = 90 Les mots de passe des utilisateurs ne sont valables que pendant 90 jours. Après cette période, l'IdM invite les utilisateurs à les modifier.
Durée de vie minimale	Le temps minimum en heures qui doit s'écouler entre deux opérations de changement de mot de passe.	Durée de vie minimale = 1 Après avoir modifié leur mot de passe, les utilisateurs doivent attendre au moins une heure avant de le modifier à nouveau.

Attribut	Explication	Exemple :
Taille de l'histoire	Le nombre de mots de passe précédents qui sont stockés. Un utilisateur ne peut pas réutiliser un mot de passe de son historique de mots de passe, mais peut réutiliser d'anciens mots de passe qui ne sont pas stockés.	Taille de l'historique = 0 Dans ce cas, l'historique des mots de passe est vide et les utilisateurs peuvent réutiliser n'importe lequel de leurs mots de passe précédents.
Classes de personnages	Le nombre de classes de caractères différentes que l'utilisateur doit utiliser dans le mot de passe. Les classes de caractères sont les suivantes : * Caractères majuscules * Caractères minuscules * Chiffres * Caractères spéciaux, tels que virgule (,), point (.), astérisque (*) * Autres caractères UTF-8 L'utilisation d'un caractère trois fois ou plus à la suite diminue la classe du caractère d'une unité. Par exemple : * **Secret1** a 3 classes de caractères : majuscules, minuscules, chiffres * **Secret111** a 2 classes de caractères : majuscules, minuscules, chiffres, et une pénalité de -1 pour l'utilisation répétée de **1**	Classes de caractères = 0 Le nombre de classes requis par défaut est de 0. Pour configurer ce nombre, exécutez la commande **ipa pwpolicy-mod** avec l'option **--minclasses**. Voir également la note importante sous ce tableau.
Longueur minimale	Nombre minimum de caractères dans un mot de passe. Si l'une des options de politique de mot de passe supplémentaires est définie, la longueur minimale des mots de passe est de 6, quelle que soit la valeur de l'option Longueur min.	Longueur minimale = 8 Les utilisateurs ne peuvent pas utiliser de mots de passe de moins de 8 caractères.
Défaillances maximales	Nombre maximal de tentatives de connexion échouées avant que l'IdM ne verrouille le compte de l'utilisateur.	Nombre maximal d'échecs = 6 L'IdM verrouille le compte de l'utilisateur lorsque celui-ci saisit un mot de passe erroné sept fois de suite.

Attribut	Explication	Exemple :
Intervalle de réinitialisation des défaillances	Délai en secondes après lequel l'IdM réinitialise le nombre actuel de tentatives de connexion infructueuses.	Intervalle de réinitialisation des défaillances = 60 Si l'utilisateur attend plus d'une minute après le nombre d'échecs de connexion défini à l'adresse **Max failures**, il peut tenter de se connecter à nouveau sans risquer de voir son compte bloqué.
Durée du verrouillage	Durée en secondes pendant laquelle le compte de l'utilisateur est verrouillé après le nombre de tentatives de connexion infructueuses défini à l'adresse **Max failures**.	Durée du verrouillage = 600 Les utilisateurs dont le compte est verrouillé ne peuvent pas se connecter pendant 10 minutes.

IMPORTANT

Utilisez l'alphabet anglais et les symboles courants pour les classes de caractères requises si vous disposez d'un ensemble diversifié de matériel qui peut ne pas avoir accès aux caractères et symboles internationaux. Pour plus d'informations sur les politiques de classes de caractères dans les mots de passe, voir Quels sont les caractères valides dans un mot de passe ? dans la base de connaissances de Red Hat.

6.3. ASSURER LA PRÉSENCE D'UNE POLITIQUE DE MOT DE PASSE DANS IDM À L'AIDE D'UN PLAYBOOK ANSIBLE

Cette section décrit comment assurer la présence d'une politique de mot de passe dans la gestion des identités (IdM) à l'aide d'un playbook Ansible.

Dans la politique de mot de passe **global_policy** par défaut dans IdM, le nombre de classes de caractères différents dans le mot de passe est fixé à 0. La taille de l'historique est également fixée à 0.

Effectuez cette procédure pour appliquer une politique de mot de passe plus stricte pour un groupe IdM à l'aide d'un playbook Ansible.

NOTE

Vous ne pouvez définir une politique de mot de passe que pour un groupe IdM. Vous ne pouvez pas définir une politique de mot de passe pour un utilisateur individuel.

Conditions préalables

- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.

- L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
- L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

- Vous connaissez le mot de passe de l'administrateur IdM.
- Le groupe pour lequel vous vous assurez de la présence d'une politique de mot de passe existe dans IdM.

Procédure

1. Créez un fichier d'inventaire, par exemple **inventory.file**, et définissez le **FQDN** de votre serveur IdM dans la section **[ipaserver]**:

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

2. Créez votre fichier playbook Ansible qui définit la politique de mot de passe dont vous voulez assurer la présence. Pour simplifier cette étape, copiez et modifiez l'exemple dans le fichier **/usr/share/doc/ansible-freeipa/playbooks/pwpolicy/pwpolicy_present.yml**:

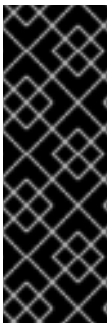
```
---
- name: Tests
  hosts: ipaserver

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Ensure presence of pwpolicy for group ops
    ipapwpolicy:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: ops
      minlife: 7
      maxlife: 49
      history: 5
      priority: 1
      lockouttime: 300
      minlength: 8
      minclasses: 4
      maxfail: 3
      failinterval: 5
```

   Pour plus de détails sur la signification des différentes variables, voir Attributs de la politique de mot de passe.

3. Exécutez le manuel de jeu :

```
$ ansible-playbook --vault-password-file=password_file -v -i
path_to_inventory_directory/inventory.file
path_to_playbooks_directory_/new_pwpolicy_present.yml
```

Vous avez utilisé avec succès un playbook Ansible pour vous assurer qu'une politique de mot de passe pour le groupe **ops** est présente dans IdM.



IMPORTANT

La priorité de la politique de mot de passe **ops** est fixée à *1*, alors que la politique de mot de passe **global_policy** n'a pas de priorité définie. Pour cette raison, la politique **ops** remplace automatiquement **global_policy** pour le groupe **ops** et est appliquée immédiatement.

global_policy sert de stratégie de repli lorsqu'aucune stratégie de groupe n'est définie pour un utilisateur, et ne peut jamais avoir la priorité sur une stratégie de groupe.

Ressources supplémentaires

- Voir le fichier **README-pwpolicy.md** dans le répertoire **/usr/share/doc/ansible-freeipa/**.
- Voir les priorités politiques du mot de passe .

6.4. OPTIONS SUPPLÉMENTAIRES DE POLITIQUE DE MOT DE PASSE DANS IDM

En tant qu'administrateur Identity Management (IdM), vous pouvez renforcer les exigences en matière de mot de passe par défaut en activant des options de politique de mot de passe supplémentaires basées sur l'ensemble des fonctionnalités de **libpwquality**. Les options supplémentaires de stratégie de mot de passe sont les suivantes :

--maxrepeat

Spécifie le nombre maximal acceptable de caractères consécutifs identiques dans le nouveau mot de passe.

--maxsequence

Spécifie la longueur maximale des séquences de caractères monotones dans le nouveau mot de passe. Des exemples d'une telle séquence sont **12345** ou **fedcb**. La plupart de ces mots de passe ne passeront pas le contrôle de simplicité.

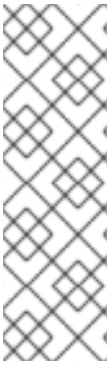
--dictcheck

Si non nul, vérifie si le mot de passe, avec d'éventuelles modifications, correspond à un mot du dictionnaire. Actuellement, **libpwquality** effectue la vérification du dictionnaire à l'aide de la bibliothèque **cracklib**.

--usercheck

S'il est différent de zéro, il vérifie si le mot de passe, avec d'éventuelles modifications, contient le nom de l'utilisateur sous une forme ou une autre. Cette vérification n'est pas effectuée pour les noms d'utilisateur de moins de 3 caractères.

Vous ne pouvez pas appliquer les options supplémentaires de politique de mot de passe aux mots de passe existants. Si vous appliquez l'une des options supplémentaires, l'IdM définit automatiquement l'option **--minlength**, le nombre minimum de caractères dans un mot de passe, à **6** caractères.



NOTE

Dans un environnement mixte avec des serveurs RHEL 7, RHEL 8 et RHEL 9, vous pouvez appliquer les paramètres supplémentaires de la stratégie de mot de passe uniquement sur les serveurs fonctionnant sous RHEL 8.4 ou une version ultérieure. Si un utilisateur est connecté à un client IdM et que ce dernier communique avec un serveur IdM fonctionnant sous RHEL 8.3 ou une version antérieure, les nouvelles exigences en matière de stratégie de mot de passe définies par l'administrateur système ne seront pas appliquées. Pour garantir un comportement cohérent, mettez à niveau ou mettez à jour tous les serveurs vers RHEL 8.4 ou une version ultérieure.

Ressources complémentaires :

- Appliquer des politiques de mot de passe supplémentaires à un groupe IdM
- **pwquality(3)** page de manuel

6.5. APPLIQUER DES OPTIONS SUPPLÉMENTAIRES DE POLITIQUE DE MOT DE PASSE À UN GROUPE IDM

Cette section décrit comment appliquer des options supplémentaires de politique de mot de passe dans la gestion des identités (IdM). L'exemple décrit comment renforcer la politique de mot de passe pour le groupe **managers** en s'assurant que les nouveaux mots de passe ne contiennent pas les noms d'utilisateur respectifs des utilisateurs et que les mots de passe ne contiennent pas plus de deux caractères identiques à la suite.

Conditions préalables

- Vous êtes connecté en tant qu'administrateur IdM.
- Le groupe **managers** existe dans IdM.
- La politique de mot de passe **managers** existe dans IdM.

Procédure

1. Appliquer le contrôle du nom d'utilisateur à tous les nouveaux mots de passe proposés par les utilisateurs du groupe **managers**:

```
$ ipa pwpolicy-mod --usercheck=True managers
```

NOTE

Si vous ne spécifiez pas le nom de la politique de mot de passe, la valeur par défaut **global_policy** est modifiée.

2. Définissez le nombre maximum de caractères identiques consécutifs à 2 dans la politique de mot de passe **managers**:

```
$ ipa pwpolicy-mod --maxrepeat=2 managers
```

Un mot de passe ne sera pas accepté s'il contient plus de 2 caractères identiques consécutifs. Par exemple, la combinaison **eR873mUi111YJQ** est inacceptable parce qu'elle contient trois **1**consécutifs.

Vérification

1. Ajouter un utilisateur test nommé **test_user**:

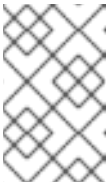
```
$ ipa user-add test_user
First name: test
Last name: user
----------------------------
Added user "test_user"
----------------------------
```

2. Ajoutez l'utilisateur test au groupe **managers**:
   a. Dans l'interface Web IdM, cliquez sur **Identité → Groupes → Groupes d'utilisateurs**.
   b. Cliquez sur **managers**.
   c. Cliquez sur **Add**.
   d. Dans la page **Add users into user group 'managers'**, vérifiez **test_user**.
   e. Cliquez sur la flèche **>** pour déplacer l'utilisateur dans la colonne **Prospective**.
   f. Cliquez sur **Add**.
3. Réinitialiser le mot de passe de l'utilisateur test :
   a. Aller à **Identité → Utilisateurs**.
   b. Cliquez sur **test_user**.
   c. Dans le menu **Actions**, cliquez sur **Reset Password**.
   d. Entrez un mot de passe temporaire pour l'utilisateur.
4. Sur la ligne de commande, essayez d'obtenir un ticket Kerberos (TGT) pour l'adresse **test_user**:

```
$ kinit test_user
```

   a. Saisissez le mot de passe temporaire.
   b. Le système vous informe que vous devez modifier votre mot de passe. Saisissez un mot de passe qui contient le nom d'utilisateur **test_user**:

```
Password expired. You must change it now.
Enter new password:
Enter it again:
Password change rejected: Password not changed.
Unspecified password quality failure while trying to change password.
Please try again.
```



NOTE

Kerberos ne dispose pas d'une politique de signalement des erreurs de mot de passe très fine et, dans certains cas, ne fournit pas de raison claire pour laquelle un mot de passe a été rejeté.

c. Le système vous informe que le mot de passe introduit a été rejeté. Introduisez un mot de passe contenant au moins trois caractères identiques successifs :

```
Password change rejected: Password not changed.
Unspecified password quality failure while trying to change password.
Please try again.

Enter new password:
Enter it again:
```

d. Le système vous informe que le mot de passe introduit a été rejeté. Saisissez un mot de passe qui répond aux critères de la politique de mot de passe **managers**:

```
Password change rejected: Password not changed.
Unspecified password quality failure while trying to change password.
Please try again.

Enter new password:
Enter it again:
```

5. Voir le TGT :

```
$ klist
Ticket cache: KCM:0:33945
Default principal: test_user@IDM.EXAMPLE.COM

Valid starting       Expires              Service principal
07/07/2021 12:44:44  07/08/2021 12:44:44
krbtgt@IDM.EXAMPLE.COM@IDM.EXAMPLE.COM
```

La politique de mot de passe **managers** fonctionne désormais correctement pour les utilisateurs du groupe **managers**.

Ressources supplémentaires

- Politiques de mot de passe supplémentaires dans l'IdM

6.6. UTILISATION D'UN PLAYBOOK ANSIBLE POUR APPLIQUER DES OPTIONS DE POLITIQUE DE MOT DE PASSE SUPPLÉMENTAIRES À UN GROUPE IDM

Vous pouvez utiliser un playbook Ansible pour appliquer des options de politique de mot de passe supplémentaires afin de renforcer les exigences de la politique de mot de passe pour un groupe IdM spécifique. Vous pouvez utiliser les options de politique de mot de passe **maxrepeat**, **maxsequence**, **dictcheck** et **usercheck** à cette fin. L'exemple décrit comment définir les exigences suivantes pour le groupe **managers**:

- Les nouveaux mots de passe des utilisateurs ne contiennent pas leurs noms d'utilisateur respectifs.
- Les mots de passe ne contiennent pas plus de deux caractères identiques successifs.
- Les séquences de caractères monotones dans les mots de passe ne doivent pas dépasser 3 caractères. Cela signifie que le système n'accepte pas un mot de passe comportant une séquence telle que **1234** ou **abcd**.

Conditions préalables

- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - Vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM dans le répertoire ***~/MyPlaybooks/*** dans le répertoire
  - Vous avez stocké votre site **ipaadmin_password** dans le coffre-fort **secret.yml** Ansible.
- Le groupe pour lequel vous vous assurez de la présence d'une politique de mot de passe existe dans IdM.

Procédure

1. Créez votre fichier Ansible playbook **manager_pwpolicy_present.yml** qui définit la politique de mot de passe dont vous voulez assurer la présence. Pour simplifier cette étape, copiez et modifiez l'exemple suivant :

```
---
- name: Tests
  hosts: ipaserver

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Ensure presence of usercheck and maxrepeat pwpolicy for group managers
    ipapwpolicy:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: managers
      usercheck: True
      maxrepeat: 2
      maxsequence: 3
```

2. Exécutez le manuel de jeu :

```
$ ansible-playbook --vault-password-file=password_file -v -i
path_to_inventory_directory/inventory.file
path_to_playbooks_directory_/manager_pwpolicy_present.yml
```

Vérification

1. Ajouter un utilisateur test nommé **test_user**:

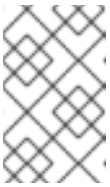
```
$ ipa user-add test_user
First name: test
Last name: user
----------------------------
Added user "test_user"
----------------------------
```

2. Ajoutez l'utilisateur test au groupe **managers**:
   a. Dans l'interface Web IdM, cliquez sur **Identité → Groupes → Groupes d'utilisateurs**.
   b. Cliquez sur **managers**.
   c. Cliquez sur **Add**.
   d. Dans la page **Add users into user group 'managers'**, vérifiez **test_user**.
   e. Cliquez sur la flèche **>** pour déplacer l'utilisateur dans la colonne **Prospective**.
   f. Cliquez sur **Add**.
3. Réinitialiser le mot de passe de l'utilisateur test :
   a. Aller à **Identité → Utilisateurs**.
   b. Cliquez sur **test_user**.
   c. Dans le menu **Actions**, cliquez sur **Reset Password**.
   d. Entrez un mot de passe temporaire pour l'utilisateur.
4. Sur la ligne de commande, essayez d'obtenir un ticket Kerberos (TGT) pour l'adresse **test_user**:

```
$ kinit test_user
```

   a. Saisissez le mot de passe temporaire.
   b. Le système vous informe que vous devez modifier votre mot de passe. Saisissez un mot de passe qui contient le nom d'utilisateur **test_user**:

```
Password expired. You must change it now.
Enter new password:
Enter it again:
Password change rejected: Password not changed.
Unspecified password quality failure while trying to change password.
Please try again.
```



NOTE

Kerberos ne dispose pas d'une politique de signalement des erreurs de mot de passe très fine et, dans certains cas, ne fournit pas de raison claire pour laquelle un mot de passe a été rejeté.

c. Le système vous informe que le mot de passe introduit a été rejeté. Introduisez un mot de passe contenant au moins trois caractères identiques successifs :

```
Password change rejected: Password not changed.
Unspecified password quality failure while trying to change password.
Please try again.

Enter new password:
Enter it again:
```

d. Le système vous informe que le mot de passe saisi a été rejeté. Introduisez un mot de passe qui contient une séquence de caractères monotone de plus de 3 caractères. Des exemples de telles séquences sont **1234** et **fedc**:

```
Password change rejected: Password not changed.
Unspecified password quality failure while trying to change password.
Please try again.

Enter new password:
Enter it again:
```

e. Le système vous informe que le mot de passe introduit a été rejeté. Saisissez un mot de passe qui répond aux critères de la politique de mot de passe **managers**:

```
Password change rejected: Password not changed.
Unspecified password quality failure while trying to change password.
Please try again.

Enter new password:
Enter it again:
```

5. Vérifiez que vous avez obtenu un TGT, ce qui n'est possible qu'après avoir introduit un mot de passe valide :

```
$ klist
Ticket cache: KCM:0:33945
Default principal: test_user@IDM.EXAMPLE.COM

Valid starting       Expires              Service principal
07/07/2021 12:44:44  07/08/2021 12:44:44
krbtgt@IDM.EXAMPLE.COM@IDM.EXAMPLE.COM
```

Ressources supplémentaires

- Politiques de mot de passe supplémentaires dans l'IdM
- **/usr/share/doc/ansible-freeipa/README-pwpolicy.md**
- **/usr/share/doc/ansible-freeipa/playbooks/pwpolicy**

CHAPITRE 7. GESTION DES NOTIFICATIONS D'EXPIRATION DE MOT DE PASSE

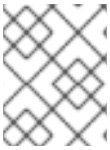
Vous pouvez utiliser l'outil EPN (Expiring Password Notification), fourni par le paquetage **ipa-client-epn**, pour établir une liste des utilisateurs de la gestion des identités (IdM) dont les mots de passe expirent dans un laps de temps configuré. Pour installer, configurer et utiliser l'outil EPN, reportez-vous aux sections correspondantes.

7.1. QU'EST-CE QUE L'OUTIL DE NOTIFICATION D'EXPIRATION DU MOT DE PASSE ?

L'outil EPN (Expiring Password Notification) est un outil autonome qui permet de dresser une liste des utilisateurs de la gestion des identités (IdM) dont les mots de passe expirent dans un laps de temps donné.

Les administrateurs IdM peuvent utiliser l'EPN pour :

- Affiche une liste des utilisateurs concernés au format JSON, qui est créée lors de l'exécution en mode "dry-run".
- Calculez le nombre d'e-mails qui seront envoyés pour un jour ou une période donnés.
- Envoyer aux utilisateurs des notifications par courrier électronique concernant l'expiration du mot de passe.
- Configurez le site **ipa-epn.timer** pour qu'il exécute quotidiennement l'outil EPN et envoie un courrier électronique aux utilisateurs dont les mots de passe expirent dans les plages de dates futures définies.
- Personnaliser la notification par courriel à envoyer aux utilisateurs.



NOTE

Si un compte d'utilisateur est désactivé, aucune notification par courrier électronique n'est envoyée si le mot de passe est sur le point d'expirer.

7.2. INSTALLATION DE L'OUTIL DE NOTIFICATION D'EXPIRATION DE MOT DE PASSE

Cette procédure décrit comment installer l'outil EPN (Expiring Password Notification).

Conditions préalables

.

- Installez l'outil EPN sur un réplica Identity Management (IdM) ou un client IdM avec un serveur Postfix SMTP local configuré en tant qu'hôte intelligent.

Procédure

- Installer l'outil EPN :

```
# dnf install ipa-client-epn
```

7.3. EXÉCUTION DE L'OUTIL EPN POUR ENVOYER DES COURRIELS AUX UTILISATEURS DONT LES MOTS DE PASSE ARRIVENT À EXPIRATION

Cette procédure décrit comment exécuter l'outil EPN (Expiring Password Notification) pour envoyer des courriels aux utilisateurs dont le mot de passe arrive à expiration.

NOTE

L'outil EPN est sans état. Si l'outil EPN n'envoie pas de courrier électronique aux utilisateurs dont les mots de passe expirent un jour donné, il n'enregistre pas la liste de ces utilisateurs.

Conditions préalables

- Le paquetage **ipa-client-epn** est installé. Voir Installation de l'outil de notification d'expiration de mot de passe.
- Personnalisez le modèle d'e-mail **ipa-epn** si nécessaire. Voir Modifier le modèle d'e-mail de notification d'expiration du mot de passe.

Procédure

1. Mettez à jour le fichier de configuration **epn.conf** pour définir les options permettant à l'outil EPN d'avertir les utilisateurs de l'expiration prochaine de leur mot de passe.

```
# vi /etc/ipa/epn.conf
```

2. Mettez à jour le site **notify_ttls** si nécessaire. Par défaut, les utilisateurs dont les mots de passe expirent dans 28, 14, 7, 3 et 1 jour(s) sont avertis.

```
notify_ttls = 28, 14, 7, 3, 1
```

3. Configurez votre serveur SMTP et votre port :

```
smtp_server = localhost
smtp_port = 25
```

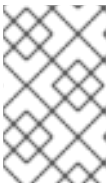
4. Indiquez l'adresse électronique à partir de laquelle la notification d'expiration du courrier électronique est envoyée. Les courriels non délivrés sont renvoyés à cette adresse.

```
mail_from =admin-email@example.com
```

5. Enregistrez le fichier **/etc/ipa/epn.conf**.

6. Exécutez l'outil EPN en mode "dry-run" pour générer une liste des utilisateurs auxquels la notification par e-mail de l'expiration du mot de passe serait envoyée si vous exécutiez l'outil sans l'option **--dry-run**.

```
ipa-epn --dry-run
[
   {
    "uid": "user5",
    "cn": "user 5",
    "krbpasswordexpiration": "2020-04-17 15:51:53",
    "mail": "['user5@ipa.test']"
   }
]
[
   {
    "uid": "user6",
    "cn": "user 6",
    "krbpasswordexpiration": "2020-12-17 15:51:53",
    "mail": "['user5@ipa.test']"
    }
]
The IPA-EPN command was successful
```



NOTE

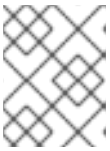
Si la liste des utilisateurs renvoyée est très importante et que vous exécutez l'outil sans l'option **--dry-run**, cela peut entraîner un problème avec votre serveur de messagerie.

7. Exécutez l'outil EPN sans l'option **--dry-run** pour envoyer des courriels d'expiration à la liste de tous les utilisateurs renvoyés lorsque vous avez exécuté l'outil EPN en mode "dry run" :

```
ipa-epn
[
  {
    "uid": "user5",
    "cn": "user 5",
    "krbpasswordexpiration": "2020-10-01 15:51:53",
    "mail": "['user5@ipa.test']"
  }
]
[
  {
    "uid": "user6",
    "cn": "user 6",
    "krbpasswordexpiration": "2020-12-17 15:51:53",
    "mail": "['user5@ipa.test']"
  }
]
The IPA-EPN command was successful
```

8. Vous pouvez ajouter EPN à n'importe quel système de surveillance et l'invoquer avec les options **--from-nbdays** et **--to-nbdays** pour déterminer combien de mots de passe d'utilisateurs vont expirer dans un délai spécifique :

```
# ipa-epn --from-nbdays 8 --to-nbdays 12
```



NOTE

Si vous invoquez l'outil EPN avec les options **--from-nbdays** et **--to-nbdays**, il est automatiquement exécuté en mode "dry-run".

Verification steps

- Exécutez l'outil EPN et vérifiez qu'une notification par courrier électronique est envoyée.

Ressources supplémentaires

- Voir la page de manuel **ipa-epn**.
- Voir la page de manuel **epn.conf**.

7.4. PERMETTRE À IPA-EPN.TIMER D'ENVOYER UN COURRIER ÉLECTRONIQUE À TOUS LES UTILISATEURS DONT LE MOT DE PASSE ARRIVE À EXPIRATION

Cette procédure décrit comment utiliser **ipa-epn.timer** pour exécuter l'outil Expiring Password Notification (EPN) afin d'envoyer des courriels aux utilisateurs dont les mots de passe expirent. Le site **ipa-epn.timer** analyse le fichier **epn.conf** et envoie un courrier électronique aux utilisateurs dont les mots de passe expirent dans les plages de dates futures définies dans ce fichier.

Conditions préalables

- Le paquetage **ipa-client-epn** est installé. Voir Installation de l'outil de notification d'expiration de mot de passe
- Personnalisez le modèle d'e-mail **ipa-epn** si nécessaire. Voir Modifier le modèle d'e-mail de notification d'expiration du mot de passe

Procédure

- Démarrer le site **ipa-epn.timer**:

```
systemctl start ipa-epn.timer
```

Une fois que vous avez lancé la minuterie, l'outil EPN est exécuté par défaut tous les jours à 1 heure du matin.

Ressources supplémentaires

- Voir la page de manuel **ipa-epn**.

7.5. MODIFICATION DU MODÈLE DE COURRIEL DE NOTIFICATION D'EXPIRATION DU MOT DE PASSE

Cette procédure décrit comment personnaliser le modèle de message électronique EPN (Expiring Password Notification).

Conditions préalables

- Le paquet **ipa-client-epn** est installé.

Procédure

1. Ouvrez le modèle de message EPN :

```
# vi /etc/ipa/epn/expire_msg.template
```

2. Mettez à jour le texte du modèle si nécessaire.

```
Hi {{ fullname }},

Your password will expire on {{ expiration }}.

Please change it as soon as possible.
```

   Vous pouvez utiliser les variables suivantes dans le modèle.

   - Identifiant de l'utilisateur : uid
   - Nom complet : nom complet
   - Prénom : prénom
   - Nom de famille : nom
   - Date d'expiration du mot de passe : expiration

3. Sauvegarder le fichier du modèle de message.

Verification steps

- Exécutez l'outil EPN et vérifiez que la notification par courrier électronique contient le texte mis à jour.

Ressources supplémentaires

- Voir la page de manuel **ipa-epn**.

CHAPITRE 8. ACCORDER UN ACCÈS SUDO À UN UTILISATEUR IDM SUR UN CLIENT IDM

Cette section décrit comment accorder l'accès **sudo** aux utilisateurs dans la gestion des identités.

8.1. ACCÈS SUDO SUR UN CLIENT IDM

Les administrateurs système peuvent accorder l'accès **sudo** pour permettre aux utilisateurs non root d'exécuter des commandes administratives qui sont normalement réservées à l'utilisateur **root**. Par conséquent, lorsque les utilisateurs doivent exécuter une commande administrative normalement réservée à l'utilisateur **root**, ils font précéder cette commande de **sudo**. Après avoir introduit son mot de passe, la commande est exécutée comme s'il s'agissait de l'utilisateur **root**. Pour exécuter une commande **sudo** en tant qu'autre utilisateur ou groupe, tel qu'un compte de service de base de données, vous pouvez configurer une règle *RunAs alias* pour **sudo**.

Si un hôte Red Hat Enterprise Linux (RHEL) 8 est inscrit en tant que client Identity Management (IdM), vous pouvez spécifier les règles **sudo** définissant quels utilisateurs IdM peuvent exécuter quelles commandes sur l'hôte de la manière suivante :

- Localement dans le fichier **/etc/sudoers**
- Au niveau central dans l'IdM

Cette section décrit la création d'un **central sudo rule** pour un client IdM à l'aide de l'interface de ligne de commande (CLI) et de l'interface Web IdM.

Vous pouvez également configurer l'authentification sans mot de passe pour **sudo** à l'aide de l'interface GSSAPI (Generic Security Service Application Programming Interface), le moyen natif pour les systèmes d'exploitation basés sur UNIX d'accéder aux services Kerberos et de les authentifier. Vous pouvez utiliser **pam_sss_gss.so** Pluggable Authentication Module (PAM) pour invoquer l'authentification GSSAPI via le service SSSD, ce qui permet aux utilisateurs de s'authentifier auprès de la commande **sudo** à l'aide d'un ticket Kerberos valide.

Ressources supplémentaires

- Voir Gestion de l'accès sudo.

8.2. ACCORDER UN ACCÈS SUDO À UN UTILISATEUR IDM SUR UN CLIENT IDM À L'AIDE DE LA CLI

Dans la gestion des identités (IdM), vous pouvez accorder l'accès **sudo** pour une commande spécifique à un compte d'utilisateur IdM sur un hôte IdM spécifique. Commencez par ajouter une commande **sudo**, puis créez une règle **sudo** pour une ou plusieurs commandes.

Par exemple, suivez cette procédure pour créer la règle **idm_user_reboot sudo** afin d'autoriser le compte **idm_user** à exécuter la commande **/usr/sbin/reboot** sur la machine **idmclient**.

Conditions préalables

- Vous êtes connecté en tant qu'administrateur IdM.
- Vous avez créé un compte utilisateur pour **idm_user** dans IdM et déverrouillé le compte en créant un mot de passe pour l'utilisateur. Pour plus d'informations sur l'ajout d'un nouvel utilisateur IdM à l'aide de la ligne de commande, voir Ajouter des utilisateurs à l'aide de la ligne

de commande.

- Aucun compte local **idm_user** n'est présent sur l'hôte **idmclient**. L'utilisateur **idm_user** n'est pas répertorié dans le fichier local **/etc/passwd**.

Procédure

1. Récupérer un ticket Kerberos en tant qu'IdM **admin**.

```
[root@idmclient ~]# kinit admin
```

2. Ajouter la commande **/usr/sbin/reboot** à la base de données IdM des commandes **sudo**:

```
[root@idmclient ~]# ipa sudocmd-add /usr/sbin/reboot
----------------------------------
Added Sudo Command "/usr/sbin/reboot"
----------------------------------
  Sudo Command: /usr/sbin/reboot
```

3. Créez une règle **sudo** nommée **idm_user_reboot**:

```
[root@idmclient ~]# ipa sudorule-add idm_user_reboot
--------------------------------
Added Sudo Rule "idm_user_reboot"
--------------------------------
  Rule name: idm_user_reboot
  Enabled: TRUE
```

4. Ajoutez la commande **/usr/sbin/reboot** à la règle **idm_user_reboot**:

```
[root@idmclient ~]# ipa sudorule-add-allow-command idm_user_reboot --sudocmds
'/usr/sbin/reboot'
  Rule name: idm_user_reboot
  Enabled: TRUE
  Sudo Allow Commands: /usr/sbin/reboot
-------------------------
Number of members added 1
-------------------------
```

5. Appliquer la règle **idm_user_reboot** à l'hôte IdM **idmclient**:

```
[root@idmclient ~]# ipa sudorule-add-host idm_user_reboot --hosts
idmclient.idm.example.com
Rule name: idm_user_reboot
Enabled: TRUE
Hosts: idmclient.idm.example.com
Sudo Allow Commands: /usr/sbin/reboot
-------------------------
Number of members added 1
-------------------------
```

6. Ajoutez le compte **idm_user** à la règle **idm_user_reboot**:

```
[root@idmclient ~]# ipa sudorule-add-user idm_user_reboot --users idm_user
```

```
Rule name: idm_user_reboot
Enabled: TRUE
Users: idm_user
Hosts: idmclient.idm.example.com
Sudo Allow Commands: /usr/sbin/reboot
-------------------------
Number of members added 1
-------------------------
```

7. Il est possible de définir la validité de la règle **idm_user_reboot**:

   a. Pour définir l'heure à laquelle une règle **sudo** commence à être valide, utilisez la commande **ipa sudorule-mod *sudo_rule_name*** avec l'option **--setattr sudonotbefore=*DATE*** option. La valeur de *DATE* doit suivre le format de **yyyymmddHHMMSSZ**, en spécifiant explicitement les secondes. Par exemple, pour définir le début de la validité de la règle **idm_user_reboot** au 31 décembre 2025 12:34:00, entrez :

   ```
   [root@idmclient ~]# ipa sudorule-mod idm_user_reboot --setattr
   sudonotbefore=20251231123400Z
   ```

   b. Pour définir l'heure à laquelle une règle sudo cesse d'être valide, utilisez l'option **--setattr sudonotafter=DATE**. Par exemple, pour fixer la fin de la validité de la règle **idm_user_reboot** au 31 décembre 2026 12:34:00, entrez :

   ```
   [root@idmclient ~]# ipa sudorule-mod idm_user_reboot --setattr
   sudonotafter=20261231123400Z
   ```

NOTE

La propagation des changements du serveur au client peut prendre quelques minutes.

Verification steps

1. Connectez-vous à l'hôte **idmclient** en tant que compte **idm_user**.

2. Affiche les règles **sudo** que le compte **idm_user** est autorisé à appliquer.

```
[idm_user@idmclient ~]$ sudo -l
Matching Defaults entries for idm_user on idmclient:
    !visiblepw, always_set_home, match_group_by_gid, always_query_group_plugin,
    env_reset, env_keep="COLORS DISPLAY HOSTNAME HISTSIZE KDEDIR
LS_COLORS",
    env_keep+="MAIL PS1 PS2 QTDIR USERNAME LANG LC_ADDRESS LC_CTYPE",
    env_keep+="LC_COLLATE LC_IDENTIFICATION LC_MEASUREMENT
LC_MESSAGES",
    env_keep+="LC_MONETARY LC_NAME LC_NUMERIC LC_PAPER LC_TELEPHONE",
    env_keep+="LC_TIME LC_ALL LANGUAGE LINGUAS _XKB_CHARSET XAUTHORITY
KRB5CCNAME",
    secure_path=/sbin\:/bin\:/usr/sbin\:/usr/bin

User idm_user may run the following commands on idmclient:
    (root) /usr/sbin/reboot
```

3. Redémarrez la machine en utilisant **sudo**. Saisissez le mot de passe de **idm_user** lorsque vous y êtes invité :

```
[idm_user@idmclient ~]$ sudo /usr/sbin/reboot
[sudo] password for idm_user:
```

8.3. ACCORDER UN ACCÈS SUDO À UN UTILISATEUR AD SUR UN CLIENT IDM À L'AIDE DE LA CLI

Les administrateurs du système de gestion des identités (IdM) peuvent utiliser les groupes d'utilisateurs IdM pour définir les autorisations d'accès, le contrôle d'accès basé sur l'hôte, les règles **sudo** et d'autres contrôles sur les utilisateurs IdM. Les groupes d'utilisateurs IdM accordent et restreignent l'accès aux ressources du domaine IdM.

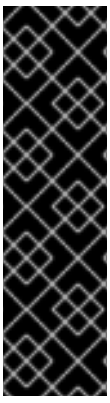
Vous pouvez ajouter à la fois Active Directory (AD) *users* et AD *groups* aux groupes d'utilisateurs IdM. Pour ce faire, procédez comme suit

1. Ajouter les utilisateurs ou groupes AD à un groupe IdM externe *non-POSIX*.
2. Ajouter le groupe IdM externe non-POSIX à un groupe IdM *POSIX*.

Vous pouvez ensuite gérer les privilèges des utilisateurs AD en gérant les privilèges du groupe POSIX. Par exemple, vous pouvez accorder l'accès **sudo** pour une commande spécifique à un groupe d'utilisateurs POSIX IdM sur un hôte IdM spécifique.

NOTE

Il est également possible d'ajouter des groupes d'utilisateurs AD aux groupes externes IdM. Cela peut faciliter la définition de politiques pour les utilisateurs Windows, en maintenant la gestion des utilisateurs et des groupes dans le domaine unique d'AD.



IMPORTANT

Ne **not** utilisez pas les ID overrides des utilisateurs AD pour les règles SUDO dans IdM. Les substitutions d'ID des utilisateurs AD ne représentent que les attributs POSIX des utilisateurs AD, et non les utilisateurs AD eux-mêmes.

Vous pouvez ajouter des dérogations d'ID en tant que membres d'un groupe. Toutefois, vous ne pouvez utiliser cette fonctionnalité que pour gérer les ressources IdM dans l'API IdM. La possibilité d'ajouter des dérogations d'ID en tant que membres d'un groupe n'est pas étendue aux environnements POSIX et vous ne pouvez donc pas l'utiliser pour l'appartenance à **sudo** ou à des règles de contrôle d'accès basées sur l'hôte (HBAC).

Cette procédure décrit comment créer la règle **ad_users_reboot sudo** pour accorder à l'utilisateur **administrator@ad-domain.com** AD la permission d'exécuter la commande **/usr/sbin/reboot** sur l'hôte **idmclient** IdM, qui est normalement réservée à l'utilisateur **root**. **administrator@ad-domain.com** est membre du groupe **ad_users_external** non-POSIX, qui est à son tour membre du groupe **ad_users** POSIX.

Conditions préalables

- Vous avez obtenu l'IdM **admin** Kerberos ticket-granting ticket (TGT).
- Une confiance inter-forêts existe entre le domaine IdM et le domaine AD **ad-domain.com**.

- Aucun compte local **administrator** n'est présent sur l'hôte **idmclient**: l'utilisateur **administrator** n'est pas répertorié dans le fichier local **/etc/passwd**.

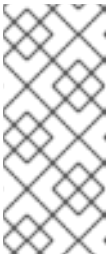
Procédure

1. Créer le groupe ***ad_users*** qui contient le groupe ***ad_users_external*** avec le membre **administrator@ad-domain**:

   a. *Optional*: Créez ou sélectionnez un groupe correspondant dans le domaine AD à utiliser pour gérer les utilisateurs AD dans le domaine IdM. Vous pouvez utiliser plusieurs groupes AD et les ajouter à différents groupes du côté IdM.

   b. Créez le groupe ***ad_users_external*** et indiquer qu'il contient des membres extérieurs au domaine IdM en ajoutant l'option **--external**:

```
[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
```



NOTE

Assurez-vous que le groupe externe que vous spécifiez ici est un groupe de sécurité AD avec une portée de groupe **global** ou **universal** comme défini dans le document sur les groupes de sécurité Active Directory . Par exemple, les groupes de sécurité AD **Domain users** ou **Domain admins** ne peuvent pas être utilisés car leur périmètre de groupe est **domain local**.

   c. Créez le groupe **ad_users**:

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

   d. Ajoutez l'utilisateur AD **administrator@ad-domain.com** à **ad_users_external** en tant que membre externe :

```
[root@ipaserver ~]# ipa group-add-member ad_users_external --external
"administrator@ad-domain.com"
 [member user]:
 [member group]:
  Group name: ad_users_external
  Description: AD users external map
  External member: S-1-5-21-3655990580-1375374850-1633065477-513
------------------------
Number of members added 1
------------------------
```

L'utilisateur AD doit être identifié par un nom complet, tel que **DOMAIN\user_name** ou **user_name@DOMAIN**. L'identité AD est ensuite mappée au SID AD de l'utilisateur. Il en va de même pour l'ajout de groupes AD.

e. Ajoutez **ad_users_external** à **ad_users** en tant que membre :

```
[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
-------------------------
```

2. Accordez aux membres de **ad_users** la permission d'exécuter **/usr/sbin/reboot** sur l'hôte **idmclient**:

a. Ajouter la commande **/usr/sbin/reboot** à la base de données IdM des commandes **sudo**:

```
[root@idmclient ~]# ipa sudocmd-add /usr/sbin/reboot
-------------------------------------
Added Sudo Command "/usr/sbin/reboot"
-------------------------------------
  Sudo Command: /usr/sbin/reboot
```

b. Créez une règle **sudo** nommée **ad_users_reboot**:

```
[root@idmclient ~]# ipa sudorule-add ad_users_reboot
---------------------------------
Added Sudo Rule "ad_users_reboot"
---------------------------------
  Rule name: ad_users_reboot
  Enabled: True
```

c. Ajoutez la commande **/usr/sbin/reboot** à la règle **ad_users_reboot**:

```
[root@idmclient ~]# ipa sudorule-add-allow-command ad_users_reboot --sudocmds
'/usr/sbin/reboot'
  Rule name: ad_users_reboot
  Enabled: True
  Sudo Allow Commands: /usr/sbin/reboot
-------------------------
Number of members added 1
-------------------------
```

d. Appliquer la règle **ad_users_reboot** à l'hôte IdM **idmclient**:

```
[root@idmclient ~]# ipa sudorule-add-host ad_users_reboot --hosts
idmclient.idm.example.com
Rule name: ad_users_reboot
Enabled: True
Hosts: idmclient.idm.example.com
Sudo Allow Commands: /usr/sbin/reboot
```

```
-------------------------
Number of members added 1
-------------------------
```

e. Ajoutez le groupe **ad_users** à la règle **ad_users_reboot**:

```
[root@idmclient ~]# ipa sudorule-add-user ad_users_reboot --groups ad_users
Rule name: ad_users_reboot
Enabled: TRUE
User Groups: ad_users
Hosts: idmclient.idm.example.com
Sudo Allow Commands: /usr/sbin/reboot
-------------------------
Number of members added 1
-------------------------
```

NOTE

La propagation des changements du serveur au client peut prendre quelques minutes.

Verification steps

1. Connectez-vous à l'hôte **idmclient** en tant que **administrator@ad-domain.com**, un membre indirect du groupe **ad_users**:

```
$ ssh administrator@ad-domain.com@ipaclient
Password:
```

2. Optionnellement, afficher les commandes **sudo** que **administrator@ad-domain.com** est autorisé à exécuter :

```
[administrator@ad-domain.com@idmclient ~]$ sudo -l
Matching Defaults entries for administrator@ad-domain.com on idmclient:
    !visiblepw, always_set_home, match_group_by_gid, always_query_group_plugin,
    env_reset, env_keep="COLORS DISPLAY HOSTNAME HISTSIZE KDEDIR
LS_COLORS",
    env_keep+="MAIL PS1 PS2 QTDIR USERNAME LANG LC_ADDRESS LC_CTYPE",
    env_keep+="LC_COLLATE LC_IDENTIFICATION LC_MEASUREMENT
LC_MESSAGES",
    env_keep+="LC_MONETARY LC_NAME LC_NUMERIC LC_PAPER LC_TELEPHONE",
    env_keep+="LC_TIME LC_ALL LANGUAGE LINGUAS _XKB_CHARSET XAUTHORITY
KRB5CCNAME",
    secure_path=/sbin\:/bin\:/usr/sbin\:/usr/bin

User administrator@ad-domain.com may run the following commands on idmclient:
    (root) /usr/sbin/reboot
```

3. Redémarrez la machine en utilisant **sudo**. Saisissez le mot de passe de **administrator@ad-domain.com** lorsque vous y êtes invité :

```
[administrator@ad-domain.com@idmclient ~]$ sudo /usr/sbin/reboot
[sudo] password for administrator@ad-domain.com:
```

Ressources supplémentaires

Ressources supplémentaires

- Utilisateurs d'Active Directory et groupes de gestion des identités
- Inclure les utilisateurs et les groupes d'un domaine Active Directory de confiance dans les règles SUDO

8.4. ACCORDER UN ACCÈS SUDO À UN UTILISATEUR IDM SUR UN CLIENT IDM À L'AIDE DE L'INTERFACE WEB IDM

Dans la gestion des identités (IdM), vous pouvez accorder l'accès **sudo** pour une commande spécifique à un compte d'utilisateur IdM sur un hôte IdM spécifique. Commencez par ajouter une commande **sudo**, puis créez une règle **sudo** pour une ou plusieurs commandes.

Suivez cette procédure pour créer la règle sudo **idm_user_reboot** afin d'autoriser le compte **idm_user** à exécuter la commande **/usr/sbin/reboot** sur la machine **idmclient**.

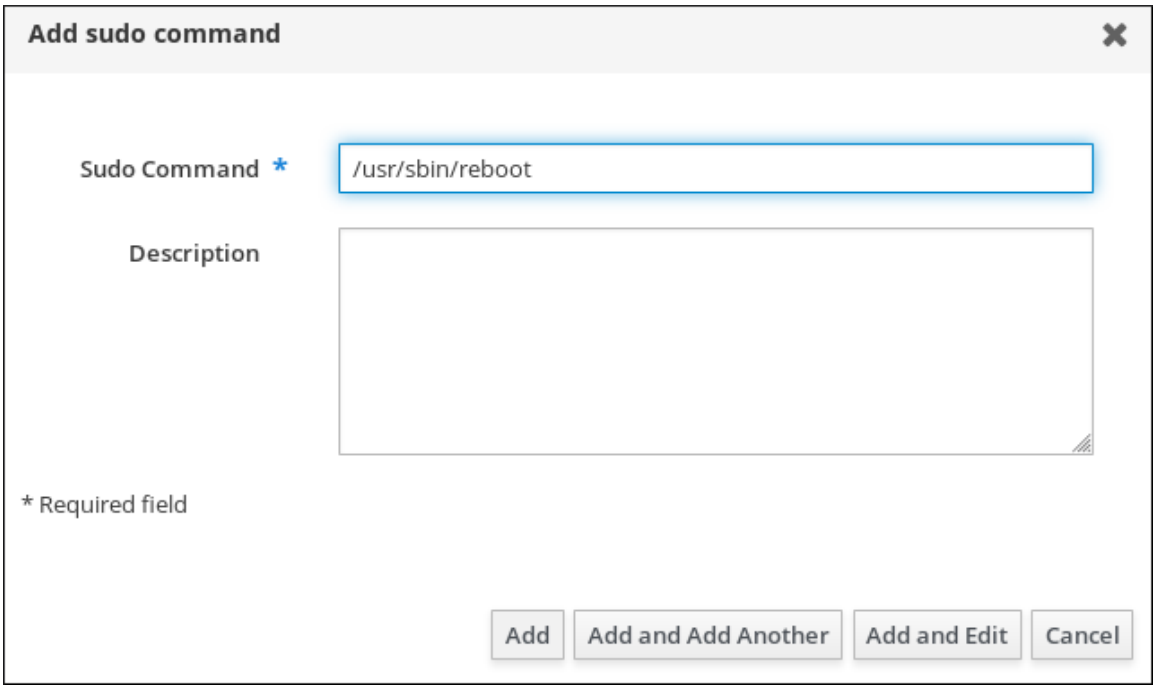
Conditions préalables

- Vous êtes connecté en tant qu'administrateur IdM.
- Vous avez créé un compte utilisateur pour **idm_user** dans IdM et déverrouillé le compte en créant un mot de passe pour l'utilisateur. Pour plus de détails sur l'ajout d'un nouvel utilisateur IdM à l'aide de l'interface de ligne de commande, voir Ajouter des utilisateurs à l'aide de la ligne de commande.
- Aucun compte local **idm_user** n'est présent sur l'hôte **idmclient**. L'utilisateur **idm_user** n'est pas répertorié dans le fichier local **/etc/passwd**.

Procédure

1. Ajouter la commande **/usr/sbin/reboot** à la base de données IdM des commandes **sudo**:
   a. Naviguez vers **Policy → Sudo → Sudo Commands**.
   b. Cliquez sur **Add** dans le coin supérieur droit pour ouvrir la boîte de dialogue **Add sudo command**.
   c. Saisissez la commande que vous souhaitez que l'utilisateur puisse exécuter en utilisant **sudo**: **/usr/sbin/reboot**.

Figure 8.1. Ajout de la commande sudo de l'IdM



d. Cliquez sur **Add**.

2. Utilisez la nouvelle entrée de commande **sudo** pour créer une règle sudo permettant à **idm_user** de redémarrer la machine **idmclient**:

   a. Naviguez vers **Policy → Sudo → Sudo rules**.

   b. Cliquez sur **Add** dans le coin supérieur droit pour ouvrir la boîte de dialogue **Add sudo rule**.

   c. Saisissez le nom de la règle **sudo**: **idm_user_reboot**.

   d. Cliquez sur **Add and Edit**

   e. Spécifiez l'utilisateur :

      i. Dans la section **Who**, cochez le bouton radio **Specified Users and Groups**.

      ii. Dans la sous-section **User category the rule applies to**, cliquez sur **Add** pour ouvrir la boîte de dialogue **Add users into sudo rule "idm_user_reboot"**.

      iii. Dans la boîte de dialogue **Add users into sudo rule "idm_user_reboot"**, dans la colonne **Available**, cochez la case **idm_user** et déplacez-la dans la colonne **Prospective**.

      iv. Cliquez sur **Add**.

   f. Spécifiez l'hôte :

      i. Dans la section **Access this host**, cochez le bouton radio **Specified Hosts and Groups**.

      ii. Dans la sous-section **Host category this rule applies to**, cliquez sur **Add** pour ouvrir la boîte de dialogue **Add hosts into sudo rule "idm_user_reboot"**.

      iii. Dans la boîte de dialogue **Add hosts into sudo rule "idm_user_reboot"**, dans la colonne **Available**, cochez la case **idmclient.idm.example.com** et déplacez-la dans la colonne **Prospective**.

iv. Cliquez sur **Add**.

g. Spécifiez les commandes :

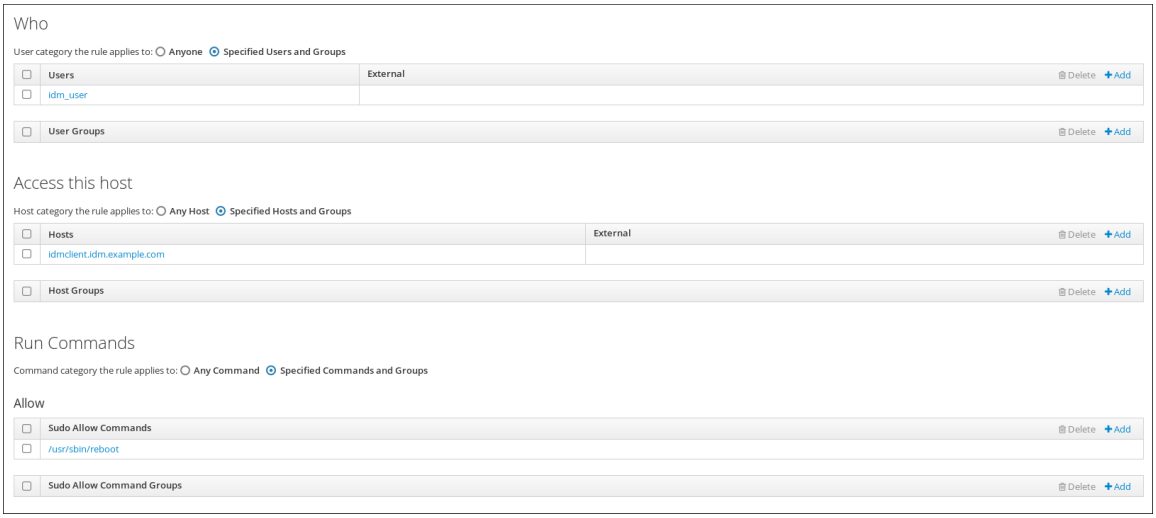
i. Dans la sous-section **Command category the rule applies to** de la section **Run Commands**, cochez le bouton radio **Specified Commands and Groups**.

ii. Dans la sous-section **Sudo Allow Commands**, cliquez sur **Add** pour ouvrir la boîte de dialogue **Add allow sudo commands into sudo rule "idm_user_reboot"**.

iii. Dans la boîte de dialogue **Add allow sudo commands into sudo rule "idm_user_reboot"**, dans la colonne **Available**, cochez la case **/usr/sbin/reboot** et déplacez-la dans la colonne **Prospective**.

iv. Cliquez sur **Add** pour revenir à la page **idm_sudo_reboot**.

Figure 8.2. Ajout d'une règle sudo à l'IdM



h. Cliquez sur **Save** dans le coin supérieur gauche.

La nouvelle règle est activée par défaut.

NOTE

La propagation des changements du serveur au client peut prendre quelques minutes.

Verification steps

1. Connectez-vous à **idmclient** en tant que **idm_user**.

2. Redémarrez la machine en utilisant **sudo**. Saisissez le mot de passe de **idm_user** lorsque vous y êtes invité :

```
$ sudo /usr/sbin/reboot
[sudo] password for idm_user:
```

Si la règle **sudo** est configurée correctement, la machine redémarre.

8.5. CRÉATION D'UNE RÈGLE SUDO SUR LA CLI QUI EXÉCUTE UNE COMMANDE EN TANT QUE COMPTE DE SERVICE SUR UN CLIENT IDM

Dans IdM, vous pouvez configurer une règle **sudo** avec une règle *RunAs alias* pour exécuter une commande **sudo** en tant qu'autre utilisateur ou groupe. Par exemple, vous pouvez avoir un client IdM qui héberge une application de base de données et vous devez exécuter des commandes en tant que compte de service local correspondant à cette application.

Utilisez cet exemple pour créer une règle **sudo** sur la ligne de commande appelée **run_third-party-app_report** afin d'autoriser le compte **idm_user** à exécuter la commande **/opt/third-party-app/bin/report** en tant que compte de service **thirdpartyapp** sur l'hôte **idmclient**.

Conditions préalables

- Vous êtes connecté en tant qu'administrateur IdM.
- Vous avez créé un compte utilisateur pour **idm_user** dans IdM et déverrouillé le compte en créant un mot de passe pour l'utilisateur. Pour plus d'informations sur l'ajout d'un nouvel utilisateur IdM à l'aide de la ligne de commande, voir Ajouter des utilisateurs à l'aide de la ligne de commande.
- Aucun compte local **idm_user** n'est présent sur l'hôte **idmclient**. L'utilisateur **idm_user** n'est pas répertorié dans le fichier local **/etc/passwd**.
- Vous avez une application personnalisée nommée **third-party-app** installée sur l'hôte **idmclient**.
- La commande **report** pour l'application **third-party-app** est installée dans le répertoire **/opt/third-party-app/bin/report**.
- Vous avez créé un compte de service local nommé **thirdpartyapp** pour exécuter les commandes de l'application **third-party-app**.

Procédure

1. Récupérer un ticket Kerberos en tant qu'IdM **admin**.

```
[root@idmclient ~]# kinit admin
```

2. Ajouter la commande **/opt/third-party-app/bin/report** à la base de données IdM des commandes **sudo**:

```
[root@idmclient ~]# ipa sudocmd-add /opt/third-party-app/bin/report
----------------------------------------------------
Added Sudo Command "/opt/third-party-app/bin/report"
----------------------------------------------------
  Sudo Command: /opt/third-party-app/bin/report
```

3. Créez une règle **sudo** nommée **run_third-party-app_report**:

```
[root@idmclient ~]# ipa sudorule-add run_third-party-app_report
-------------------------------------------
Added Sudo Rule "run_third-party-app_report"
-------------------------------------------
  Rule name: run_third-party-app_report
  Enabled: TRUE
```

4. Utilisez l'option **--users=*<user>*** pour spécifier l'utilisateur RunAs pour la commande **sudorule-add-runasuser**:

```
[root@idmclient ~]# ipa sudorule-add-runasuser run_third-party-app_report --users=thirdpartyapp
  Rule name: run_third-party-app_report
  Enabled: TRUE
  RunAs External User: thirdpartyapp
-------------------------
Number of members added 1
-------------------------
```

L'utilisateur (ou le groupe spécifié avec l'option **--groups=***) peut être externe à IdM, comme un compte de service local ou un utilisateur Active Directory. N'ajoutez pas de préfixe **%** pour les noms de groupes.

5. Ajoutez la commande **/opt/third-party-app/bin/report** à la règle **run_third-party-app_report**:

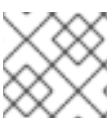
```
[root@idmclient ~]# ipa sudorule-add-allow-command run_third-party-app_report --sudocmds '/opt/third-party-app/bin/report'
Rule name: run_third-party-app_report
Enabled: TRUE
Sudo Allow Commands: /opt/third-party-app/bin/report
RunAs External User: thirdpartyapp
-------------------------
Number of members added 1
-------------------------
```

6. Appliquer la règle **run_third-party-app_report** à l'hôte IdM **idmclient**:

```
[root@idmclient ~]# ipa sudorule-add-host run_third-party-app_report --hosts idmclient.idm.example.com
Rule name: run_third-party-app_report
Enabled: TRUE
Hosts: idmclient.idm.example.com
Sudo Allow Commands: /opt/third-party-app/bin/report
RunAs External User: thirdpartyapp
-------------------------
Number of members added 1
-------------------------
```

7. Ajoutez le compte **idm_user** à la règle **run_third-party-app_report**:

```
[root@idmclient ~]# ipa sudorule-add-user run_third-party-app_report --users idm_user
Rule name: run_third-party-app_report
Enabled: TRUE
Users: idm_user
Hosts: idmclient.idm.example.com
Sudo Allow Commands: /opt/third-party-app/bin/report
RunAs External User: thirdpartyapp
-------------------------
Number of members added 1
```



NOTE

La propagation des changements du serveur au client peut prendre quelques minutes.

Verification steps

1. Connectez-vous à l'hôte **idmclient** en tant que compte **idm_user**.
2. Testez la nouvelle règle sudo :
   a. Affiche les règles **sudo** que le compte **idm_user** est autorisé à appliquer.

```
[idm_user@idmclient ~]$ sudo -l
Matching Defaults entries for idm_user@idm.example.com on idmclient:
    !visiblepw, always_set_home, match_group_by_gid, always_query_group_plugin,
    env_reset, env_keep="COLORS DISPLAY HOSTNAME HISTSIZE KDEDIR
LS_COLORS",
    env_keep+="MAIL PS1 PS2 QTDIR USERNAME LANG LC_ADDRESS LC_CTYPE",
    env_keep+="LC_COLLATE LC_IDENTIFICATION LC_MEASUREMENT
LC_MESSAGES",
    env_keep+="LC_MONETARY LC_NAME LC_NUMERIC LC_PAPER
LC_TELEPHONE",
    env_keep+="LC_TIME LC_ALL LANGUAGE LINGUAS _XKB_CHARSET
XAUTHORITY KRB5CCNAME",
    secure_path=/sbin\:/bin\:/usr/sbin\:/usr/bin

User idm_user@idm.example.com may run the following commands on idmclient:
    (thirdpartyapp) /opt/third-party-app/bin/report
```

   b. Exécutez la commande **report** en tant que compte de service **thirdpartyapp**.

```
[idm_user@idmclient ~]$ sudo -u thirdpartyapp /opt/third-party-app/bin/report
[sudo] password for idm_user@idm.example.com:
Executing report...
Report successful.
```

8.6. CRÉATION D'UNE RÈGLE SUDO DANS L'INTERFACE WEB IDM QUI EXÉCUTE UNE COMMANDE EN TANT QUE COMPTE DE SERVICE SUR UN CLIENT IDM

Dans IdM, vous pouvez configurer une règle **sudo** avec une règle *RunAs alias* pour exécuter une commande **sudo** en tant qu'autre utilisateur ou groupe. Par exemple, vous pouvez avoir un client IdM qui héberge une application de base de données et vous devez exécuter des commandes en tant que compte de service local correspondant à cette application.

Utilisez cet exemple pour créer une règle **sudo** dans l'IdM WebUI appelée **run_third-party-app_report** pour permettre au compte **idm_user** d'exécuter la commande **/opt/third-party-app/bin/report** en tant que compte de service **thirdpartyapp** sur l'hôte **idmclient**.

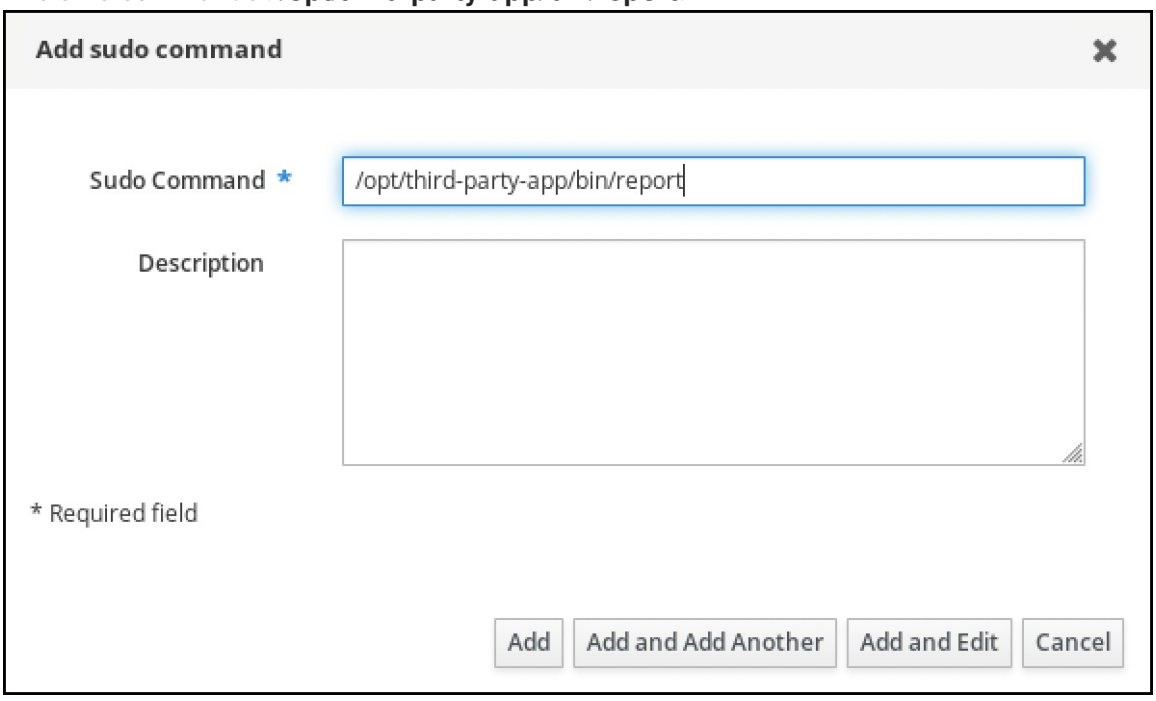
Conditions préalables

- Vous êtes connecté en tant qu'administrateur IdM.
- Vous avez créé un compte utilisateur pour **idm_user** dans IdM et déverrouillé le compte en créant un mot de passe pour l'utilisateur. Pour plus d'informations sur l'ajout d'un nouvel utilisateur IdM à l'aide de la ligne de commande, voir Ajouter des utilisateurs à l'aide de la ligne de commande.

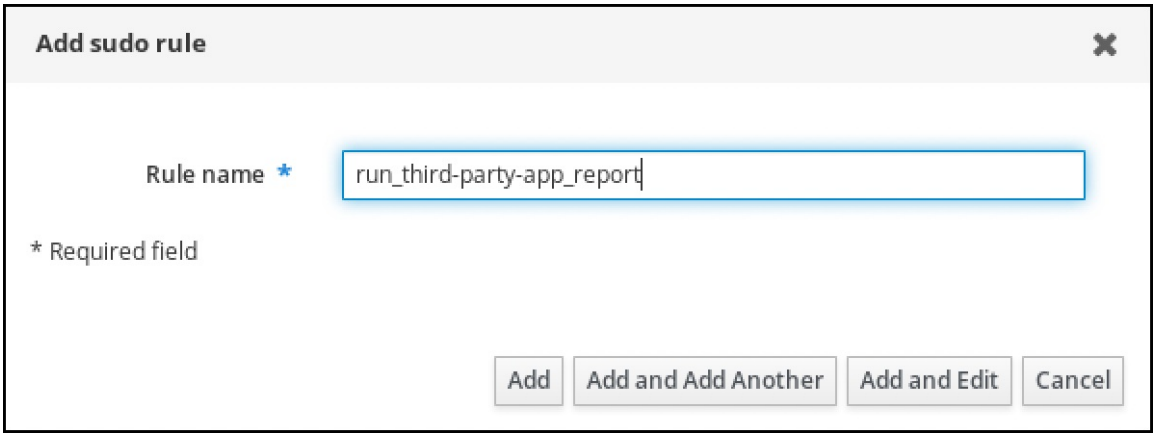
- Aucun compte local **idm_user** n'est présent sur l'hôte **idmclient**. L'utilisateur **idm_user** n'est pas répertorié dans le fichier local **/etc/passwd**.
- Vous avez une application personnalisée nommée **third-party-app** installée sur l'hôte **idmclient**.
- La commande **report** pour l'application **third-party-app** est installée dans le répertoire **/opt/third-party-app/bin/report**.
- Vous avez créé un compte de service local nommé **thirdpartyapp** pour exécuter les commandes de l'application **third-party-app**.

Procédure

1. Ajouter la commande **/opt/third-party-app/bin/report** à la base de données IdM des commandes **sudo**:
   a. Naviguez vers **Policy → Sudo → Sudo Commands**.
   b. Cliquez sur **Add** dans le coin supérieur droit pour ouvrir la boîte de dialogue **Add sudo command**.
   c. Entrez la commande : **/opt/third-party-app/bin/report**.



   d. Cliquez sur **Add**.
2. Utilisez l'entrée de commande new **sudo** pour créer la nouvelle règle **sudo**:
   a. Naviguez vers **Policy → Sudo → Sudo rules**.
   b. Cliquez sur **Add** dans le coin supérieur droit pour ouvrir la boîte de dialogue **Add sudo rule**.
   c. Saisissez le nom de la règle **sudo**: **run_third-party-app_report**.



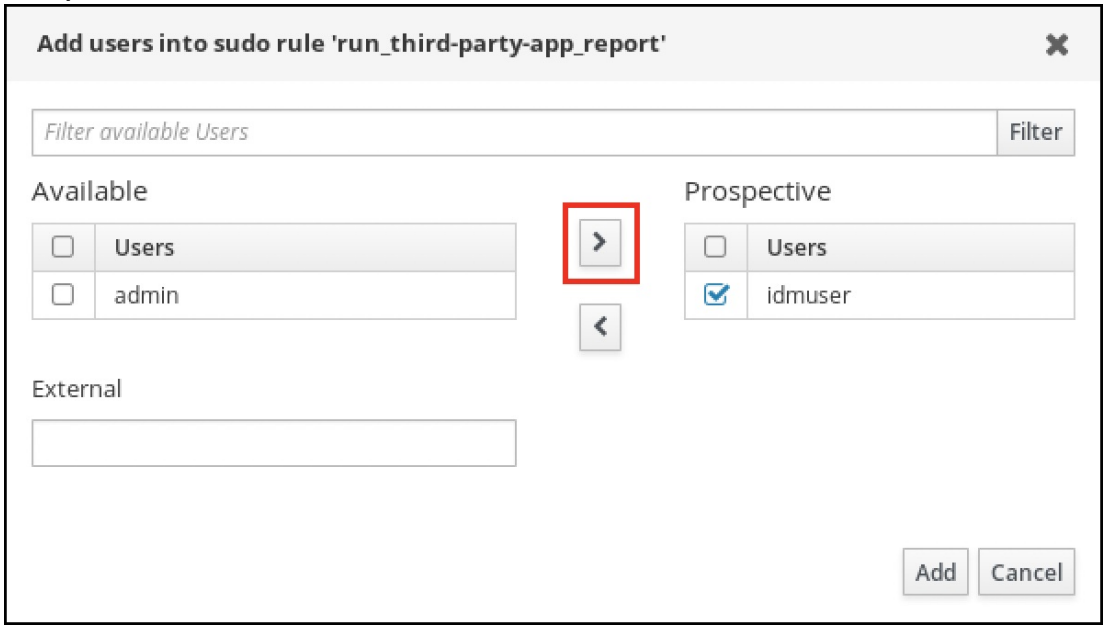
d. Cliquez sur **Add and Edit**

e. Spécifiez l'utilisateur :

   i. Dans la section **Who**, cochez le bouton radio **Specified Users and Groups**.

   ii. Dans la sous-section **User category the rule applies to**, cliquez sur **Add** pour ouvrir la boîte de dialogue **Add users into sudo rule "run_third-party-app_report"**.

   iii. Dans la boîte de dialogue **Add users into sudo rule "run_third-party-app_report"**, dans la colonne **Available**, cochez la case **idm_user** et déplacez-la dans la colonne **Prospective**.



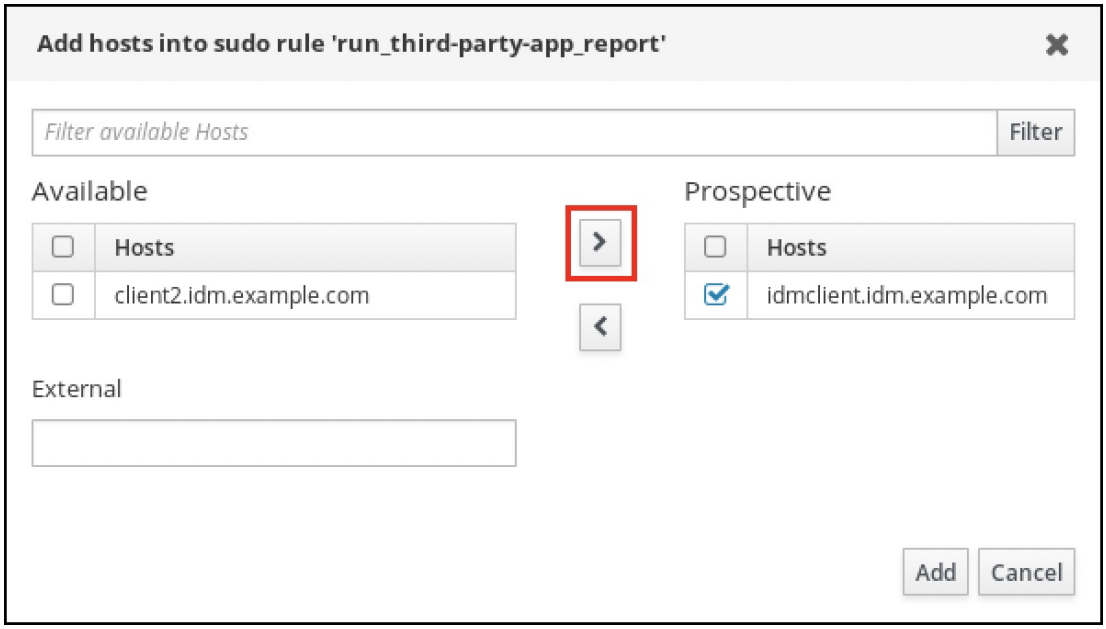
   iv. Cliquez sur **Add**.

f. Spécifiez l'hôte :

   i. Dans la section **Access this host**, cochez le bouton radio **Specified Hosts and Groups**.

   ii. Dans la sous-section **Host category this rule applies to**, cliquez sur **Add** pour ouvrir la boîte de dialogue **Add hosts into sudo rule "run_third-party-app_report"**.

   iii. Dans la boîte de dialogue **Add hosts into sudo rule "run_third-party-app_report"**, dans la colonne **Available**, cochez la case **idmclient.idm.example.com** et déplacez-la dans la colonne **Prospective**.



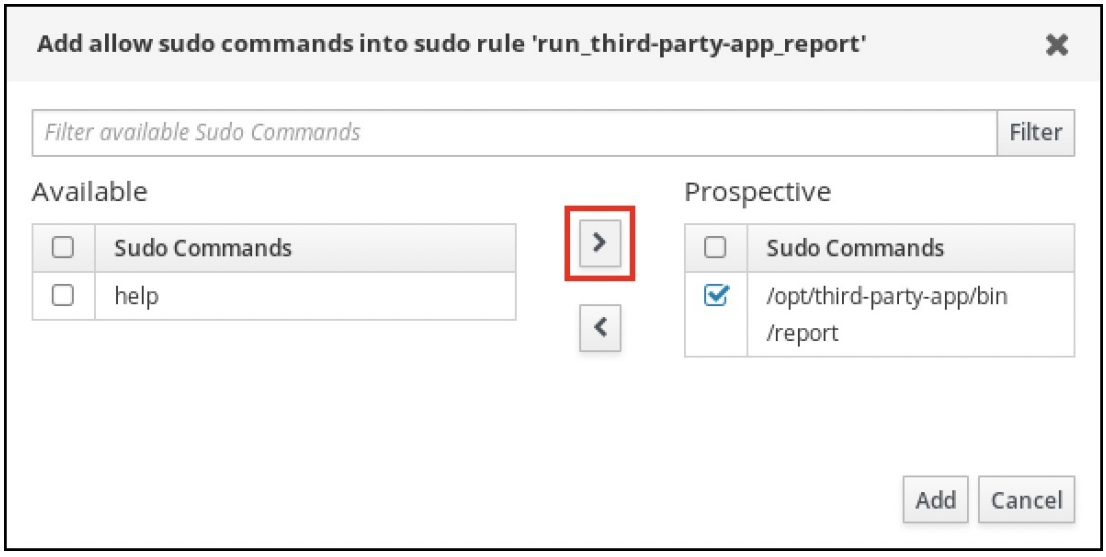
iv. Cliquez sur **Add**.

g. Spécifiez les commandes :

i. Dans la sous-section **Command category the rule applies to** de la section **Run Commands**, cochez le bouton radio **Specified Commands and Groups**.

ii. Dans la sous-section **Sudo Allow Commands**, cliquez sur **Add** pour ouvrir la boîte de dialogue **Add allow sudo commands into sudo rule "run_third-party-app_report"**.

iii. Dans la boîte de dialogue **Add allow sudo commands into sudo rule "run_third-party-app_report"**, dans la colonne **Available**, cochez la case **/opt/third-party-app/bin/report** et déplacez-la dans la colonne **Prospective**.



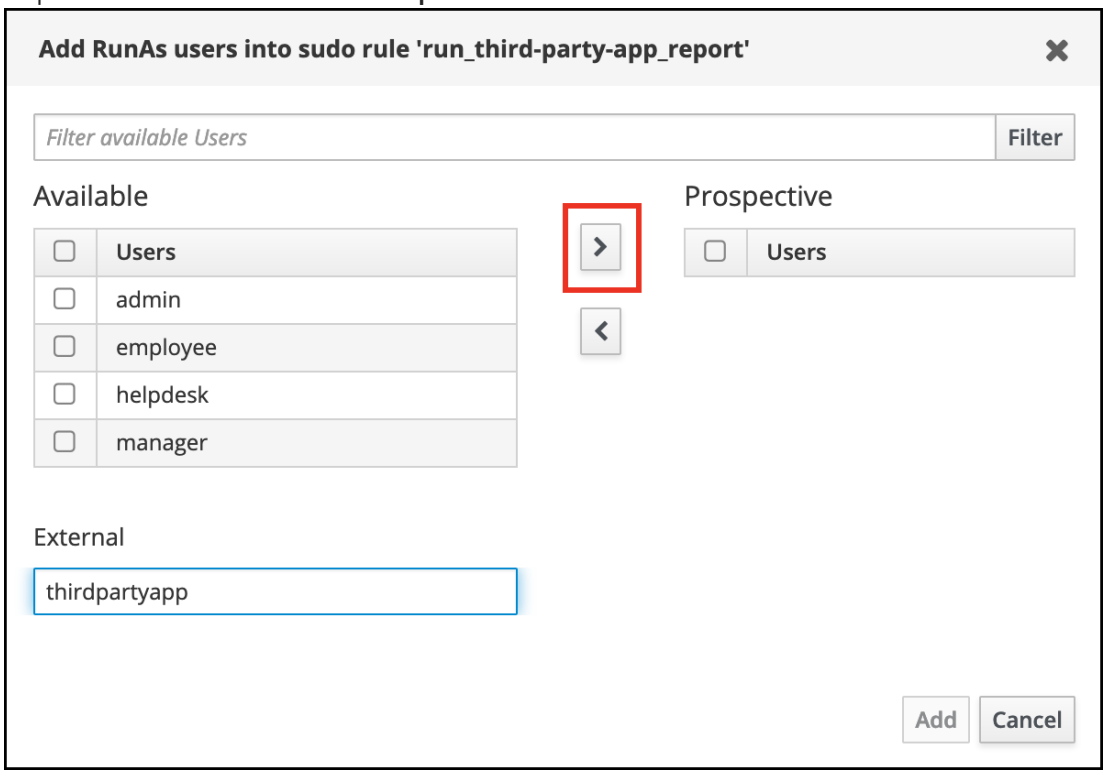
iv. Cliquez sur **Add** pour revenir à la page **run_third-party-app_report**.

h. Spécifiez l'utilisateur RunAs :

i. Dans la section **As Whom**, cochez le bouton radio **Specified Users and Groups**.

ii. Dans la sous-section **RunAs Users**, cliquez sur **Add** pour ouvrir la boîte de dialogue **Add RunAs users into sudo rule "run_third-party-app_report"**.

iii. Dans la boîte de dialogue **Add RunAs users into sudo rule "run_third-party-app_report"**, saisissez le compte de service **thirdpartyapp** dans la case **External** et déplacez-le dans la colonne **Prospective**.

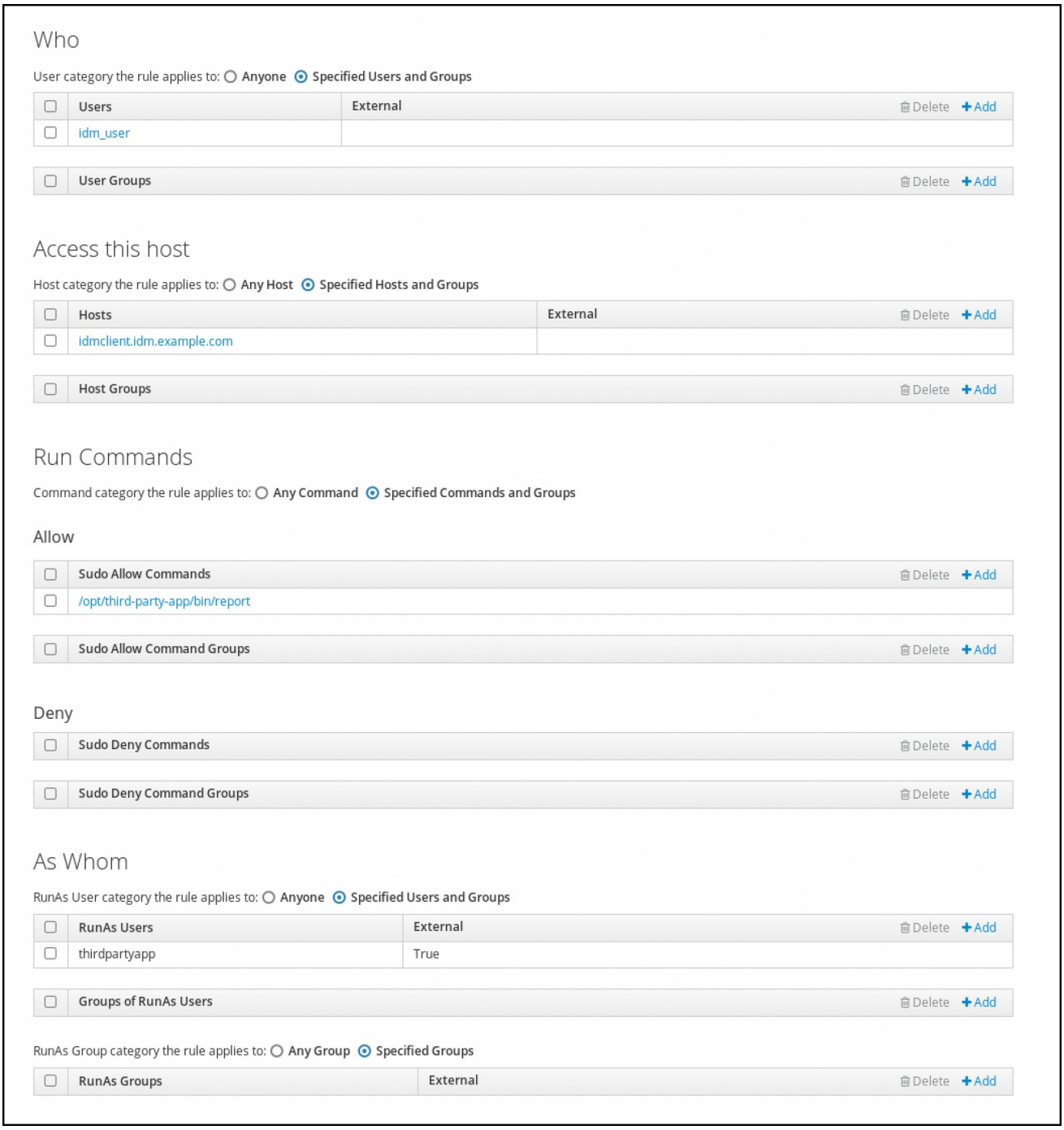


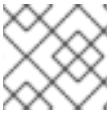
iv. Cliquez sur **Add** pour revenir à la page **run_third-party-app_report**.

i. Cliquez sur **Save** dans le coin supérieur gauche.

La nouvelle règle est activée par défaut.

Figure 8.3. Détails de la règle sudo





NOTE

La propagation des changements du serveur au client peut prendre quelques minutes.

Verification steps

1. Connectez-vous à l'hôte **idmclient** en tant que compte **idm_user**.
2. Testez la nouvelle règle sudo :
   a. Affiche les règles **sudo** que le compte **idm_user** est autorisé à appliquer.

```
[idm_user@idmclient ~]$ sudo -l
Matching Defaults entries for idm_user@idm.example.com on idmclient:
    !visiblepw, always_set_home, match_group_by_gid, always_query_group_plugin,
```

```
    env_reset, env_keep="COLORS DISPLAY HOSTNAME HISTSIZE KDEDIR
LS_COLORS",
    env_keep+="MAIL PS1 PS2 QTDIR USERNAME LANG LC_ADDRESS LC_CTYPE",
    env_keep+="LC_COLLATE LC_IDENTIFICATION LC_MEASUREMENT
LC_MESSAGES",
    env_keep+="LC_MONETARY LC_NAME LC_NUMERIC LC_PAPER
LC_TELEPHONE",
    env_keep+="LC_TIME LC_ALL LANGUAGE LINGUAS _XKB_CHARSET
XAUTHORITY KRB5CCNAME",
    secure_path=/sbin\:/bin\:/usr/sbin\:/usr/bin

User idm_user@idm.example.com may run the following commands on idmclient:
    (thirdpartyapp) /opt/third-party-app/bin/report
```

b. Exécutez la commande **report** en tant que compte de service **thirdpartyapp**.

```
[idm_user@idmclient ~]$ sudo -u thirdpartyapp /opt/third-party-app/bin/report
[sudo] password for idm_user@idm.example.com:
Executing report...
Report successful.
```

8.7. ACTIVATION DE L'AUTHENTIFICATION GSSAPI POUR SUDO SUR UN CLIENT IDM

La procédure suivante décrit l'activation de l'authentification GSSAPI (Generic Security Service Application Program Interface) sur un client IdM pour les commandes **sudo** et **sudo -i** via le module PAM **pam_sss_gss.so**. Avec cette configuration, les utilisateurs IdM peuvent s'authentifier à la commande **sudo** avec leur ticket Kerberos.

Conditions préalables

- Vous avez créé une règle **sudo** pour un utilisateur IdM qui s'applique à un hôte IdM. Pour cet exemple, vous avez créé la règle **idm_user_reboot sudo** pour accorder au compte **idm_user** la permission d'exécuter la commande **/usr/sbin/reboot** sur l'hôte **idmclient**.
- Vous devez disposer des privilèges **root** pour modifier le fichier **/etc/sssd/sssd.conf** et les fichiers PAM dans le répertoire **/etc/pam.d/**.

Procédure

1. Ouvrez le fichier de configuration **/etc/sssd/sssd.conf**.
2. Ajoutez l'entrée suivante à la section **[domain/*<domain_name>*]** l'entrée suivante.

```
[domain/<domain_name>]
pam_gssapi_services = sudo, sudo-i
```

3. Enregistrez et fermez le fichier **/etc/sssd/sssd.conf**.
4. Redémarrez le service SSSD pour charger les modifications de configuration.

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

5. Si vous utilisez RHEL 9.2 ou une version ultérieure :

   a. [Facultatif] Déterminez si vous avez sélectionné le profil **sssd authselect** :

   ```
   # authselect current
   Profile ID: sssd
   ```

   Le résultat indique que le profil **sssd authselect** est sélectionné.

   b. Si le profil **sssd authselect** est sélectionné, activez l'authentification GSSAPI :

   ```
   # authselect enable-feature with-gssapi
   ```

   c. Si le profil **sssd authselect** n'est pas sélectionné, sélectionnez-le et activez l'authentification GSSAPI :

   ```
   # authselect select sssd with-gssapi
   ```

6. Si vous utilisez RHEL 9.1 ou une version antérieure :

   a. Ouvrez le fichier de configuration PAM de **/etc/pam.d/sudo**.

   b. Ajoutez l'entrée suivante comme première ligne de la section **auth** dans le fichier **/etc/pam.d/sudo**.

   ```
   #%PAM-1.0
   auth sufficient pam_sss_gss.so
   auth       include      system-auth
   account    include      system-auth
   password   include      system-auth
   session    include      system-auth
   ```

   c. Enregistrez et fermez le fichier **/etc/pam.d/sudo**.

Verification steps

1. Connectez-vous à l'hôte en tant que compte **idm_user**.

```
[root@idm-client ~]# ssh -l idm_user@idm.example.com localhost
idm_user@idm.example.com's password:
```

2. Vérifiez que vous disposez d'un ticket d'attribution de tickets en tant que compte **idm_user**.

```
[idmuser@idmclient ~]$ klist
Ticket cache: KCM:1366201107
Default principal: idm_user@IDM.EXAMPLE.COM

Valid starting       Expires              Service principal
01/08/2021 09:11:48  01/08/2021 19:11:48
krbtgt/IDM.EXAMPLE.COM@IDM.EXAMPLE.COM
 renew until 01/15/2021 09:11:44
```

3. *(Optional)* Si vous ne disposez pas d'informations d'identification Kerberos pour le compte **idm_user**, détruisez vos informations d'identification Kerberos actuelles et demandez les informations correctes.

```
[idm_user@idmclient ~]$ kdestroy -A

[idm_user@idmclient ~]$ kinit idm_user@IDM.EXAMPLE.COM
Password for idm_user@idm.example.com:
```

4. Redémarrez la machine à l'aide de **sudo**, sans spécifier de mot de passe.

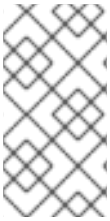
```
[idm_user@idmclient ~]$ sudo /usr/sbin/reboot
```

Ressources supplémentaires

- L'entrée GSSAPI dans la liste terminologique IdM
- Accorder un accès sudo à un utilisateur IdM sur un client IdM à l'aide de l'interface Web IdM
- Accorder un accès sudo à un utilisateur IdM sur un client IdM à l'aide de la CLI
- **pam_sss_gss (8)** page de manuel
- **sssd.conf (5)** page de manuel

8.8. ACTIVATION DE L'AUTHENTIFICATION GSSAPI ET APPLICATION DES INDICATEURS D'AUTHENTIFICATION KERBEROS POUR SUDO SUR UN CLIENT IDM

La procédure suivante décrit l'activation de l'authentification GSSAPI (Generic Security Service Application Program Interface) sur un client IdM pour les commandes **sudo** et **sudo -i** via le module PAM **pam_sss_gss.so**. En outre, seuls les utilisateurs qui se sont connectés avec une carte à puce s'authentifieront à ces commandes avec leur ticket Kerberos.



NOTE

Vous pouvez utiliser cette procédure comme modèle pour configurer l'authentification GSSAPI avec SSSD pour d'autres services compatibles avec PAM, et restreindre davantage l'accès aux seuls utilisateurs dont le ticket Kerberos est associé à un indicateur d'authentification spécifique.

Conditions préalables

- Vous avez créé une règle **sudo** pour un utilisateur IdM qui s'applique à un hôte IdM. Pour cet exemple, vous avez créé la règle **idm_user_reboot sudo** pour accorder au compte **idm_user** la permission d'exécuter la commande **/usr/sbin/reboot** sur l'hôte **idmclient**.
- Vous avez configuré l'authentification par carte à puce pour l'hôte **idmclient**.
- Vous devez disposer des privilèges **root** pour modifier le fichier **/etc/sssd/sssd.conf** et les fichiers PAM dans le répertoire **/etc/pam.d/**.

Procédure

1. Ouvrez le fichier de configuration **/etc/sssd/sssd.conf**.
2. Ajoutez les entrées suivantes à la section **[domain/*<domain_name>*]** les entrées suivantes.

```
[domain/<domain_name>]
pam_gssapi_services = sudo, sudo-i
pam_gssapi_indicators_map = sudo:pkinit, sudo-i:pkinit
```

3. Enregistrez et fermez le fichier **/etc/sssd/sssd.conf**.
4. Redémarrez le service SSSD pour charger les modifications de configuration.

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

5. Ouvrez le fichier de configuration PAM de **/etc/pam.d/sudo**.
6. Ajoutez l'entrée suivante comme première ligne de la section **auth** dans le fichier **/etc/pam.d/sudo**.

```
#%PAM-1.0
auth sufficient pam_sss_gss.so
auth       include      system-auth
account    include      system-auth
password   include      system-auth
session    include      system-auth
```

7. Enregistrez et fermez le fichier **/etc/pam.d/sudo**.
8. Ouvrez le fichier de configuration PAM de **/etc/pam.d/sudo-i**.
9. Ajoutez l'entrée suivante comme première ligne de la section **auth** dans le fichier **/etc/pam.d/sudo-i**.

```
#%PAM-1.0
auth sufficient pam_sss_gss.so
auth       include      sudo
account    include      sudo
password   include      sudo
session    optional     pam_keyinit.so force revoke
session    include      sudo
```

10. Enregistrez et fermez le fichier **/etc/pam.d/sudo-i**.

Verification steps

1. Connectez-vous à l'hôte en tant que compte **idm_user** et authentifiez-vous à l'aide d'une carte à puce.

```
[root@idmclient ~]# ssh -l idm_user@idm.example.com localhost
PIN for smart_card
```

2. Vérifiez que vous disposez d'un ticket d'attribution de billets en tant qu'utilisateur de la carte à puce.

```
[idm_user@idmclient ~]$ klist
Ticket cache: KEYRING:persistent:1358900015:krb_cache_TObtNMd
Default principal: idm_user@IDM.EXAMPLE.COM

Valid starting       Expires              Service principal
02/15/2021 16:29:48  02/16/2021 02:29:48
krbtgt/IDM.EXAMPLE.COM@IDM.EXAMPLE.COM
 renew until 02/22/2021 16:29:44
```

3. Affiche les règles **sudo** que le compte **idm_user** est autorisé à appliquer.

```
[idm_user@idmclient ~]$ sudo -l
Matching Defaults entries for idmuser on idmclient:
    !visiblepw, always_set_home, match_group_by_gid, always_query_group_plugin,
    env_reset, env_keep="COLORS DISPLAY HOSTNAME HISTSIZE KDEDIR
LS_COLORS",
    env_keep+="MAIL PS1 PS2 QTDIR USERNAME LANG LC_ADDRESS LC_CTYPE",
    env_keep+="LC_COLLATE LC_IDENTIFICATION LC_MEASUREMENT
LC_MESSAGES",
    env_keep+="LC_MONETARY LC_NAME LC_NUMERIC LC_PAPER LC_TELEPHONE",
    env_keep+="LC_TIME LC_ALL LANGUAGE LINGUAS _XKB_CHARSET XAUTHORITY
KRB5CCNAME",
    secure_path=/sbin\:/bin\:/usr/sbin\:/usr/bin

User idm_user may run the following commands on idmclient:
    (root) /usr/sbin/reboot
```

4. Redémarrez la machine à l'aide de **sudo**, sans spécifier de mot de passe.

```
[idm_user@idmclient ~]$ sudo /usr/sbin/reboot
```

Ressources supplémentaires

- Options SSSD contrôlant l'authentification GSSAPI pour les services PAM
- L'entrée GSSAPI dans la liste terminologique IdM
- Configuration de la gestion des identités pour l'authentification par carte à puce
- Indicateurs d'authentification Kerberos
- Accorder un accès sudo à un utilisateur IdM sur un client IdM à l'aide de l'interface Web IdM
- Accorder un accès sudo à un utilisateur IdM sur un client IdM en utilisant le CLI .
- **pam_sss_gss (8)** page de manuel
- **sssd.conf (5)** page de manuel

8.9. OPTIONS SSSD CONTRÔLANT L'AUTHENTIFICATION GSSAPI POUR LES SERVICES PAM

Vous pouvez utiliser les options suivantes pour le fichier de configuration **/etc/sssd/sssd.conf** afin d'ajuster la configuration GSSAPI au sein du service SSSD.

pam_gssapi_services

L'authentification GSSAPI avec SSSD est désactivée par défaut. Vous pouvez utiliser cette option pour spécifier une liste de services PAM, séparés par des virgules, qui sont autorisés à essayer l'authentification GSSAPI à l'aide du module PAM **pam_sss_gss.so**. Pour désactiver explicitement l'authentification GSSAPI, définissez cette option sur **-**.

pam_gssapi_indicators_map

Cette option ne s'applique qu'aux domaines de gestion des identités (IdM). Cette option permet de répertorier les indicateurs d'authentification Kerberos requis pour accorder à PAM l'accès à un service. Les paires doivent être au format
***<PAM_service>*:_<required_authentication_indicator>_**.
Les indicateurs d'authentification valides sont les suivants :

- **otp** pour l'authentification à deux facteurs
- **radius** pour l'authentification RADIUS
- **pkinit** pour l'authentification par PKINIT, carte à puce ou certificat
- **hardened** pour des mots de passe renforcés

pam_gssapi_check_upn

Cette option est activée et définie par défaut sur **true**. Si cette option est activée, le service SSSD exige que le nom d'utilisateur corresponde aux informations d'identification Kerberos. Si l'option **false** est activée, le module PAM **pam_sss_gss.so** authentifie chaque utilisateur capable d'obtenir le ticket de service requis.

Examples

Les options suivantes activent l'authentification Kerberos pour les services **sudo** et **sudo-i**, exigent que les utilisateurs de **sudo** s'authentifient avec un mot de passe à usage unique et que les noms d'utilisateur correspondent au principal Kerberos. Ces paramètres se trouvant dans la section **[pam]**, ils s'appliquent à tous les domaines :

```
[pam]
pam_gssapi_services = sudo, sudo-i
pam_gssapi_indicators_map = sudo:otp
pam_gssapi_check_upn = true
```

Vous pouvez également définir ces options dans des sections **[domain]** individuelles afin d'écraser toutes les valeurs globales dans la section **[pam]**. Les options suivantes appliquent des paramètres GSSAPI différents à chaque domaine :

Pour le domaine idm.example.com

- Activez l'authentification GSSAPI pour les services **sudo** et **sudo -i**.
- Exiger des authentificateurs de type certificat ou carte à puce pour la commande **sudo**.
- Exiger des authentificateurs à mot de passe unique pour la commande **sudo -i**.
- Assurer la correspondance entre les noms d'utilisateurs et les principes Kerberos.

Pour le domaine ad.example.com

- Activez l'authentification GSSAPI uniquement pour le service **sudo**.

- Ne pas imposer la correspondance entre les noms d'utilisateurs et les mandants.

```
[domain/idm.example.com]
pam_gssapi_services = sudo, sudo-i
pam_gssapi_indicators_map = sudo:pkinit, sudo-i:otp
pam_gssapi_check_upn = true
...

[domain/ad.example.com]
pam_gssapi_services = sudo
pam_gssapi_check_upn = false
...
```

Ressources supplémentaires

- Indicateurs d'authentification Kerberos

8.10. DÉPANNAGE DE L'AUTHENTIFICATION GSSAPI POUR SUDO

Si vous ne parvenez pas à vous authentifier auprès du service **sudo** à l'aide d'un ticket Kerberos provenant de l'IdM, utilisez les scénarios suivants pour résoudre votre problème.

Conditions préalables

- Vous avez activé l'authentification GSSAPI pour le service **sudo**. Voir Activation de l'authentification GSSAPI pour sudo sur un client IdM.
- Vous devez disposer des privilèges **root** pour modifier le fichier **/etc/sssd/sssd.conf** et les fichiers PAM dans le répertoire **/etc/pam.d/**.

Procédure

- Si l'erreur suivante s'affiche, il se peut que le service Kerberos ne soit pas en mesure de résoudre le domaine correct pour le ticket de service en se basant sur le nom d'hôte :

  ```
  Serveur introuvable dans la base de données Kerberos
  ```

  Dans ce cas, ajoutez le nom d'hôte directement à la section **[domain_realm]** dans le fichier de configuration Kerberos **/etc/krb5.conf**:

  ```
  [idm-user@idm-client ~]$ cat /etc/krb5.conf
  ...

  [domain_realm]
   .example.com = EXAMPLE.COM
   example.com = EXAMPLE.COM
   server.example.com = EXAMPLE.COM
  ```

- Si l'erreur suivante s'affiche, vous ne disposez pas d'informations d'identification Kerberos :

  ```
  Pas d'identifiants Kerberos disponibles
  ```

Dans ce cas, récupérez les informations d'identification Kerberos à l'aide de l'utilitaire **kinit** ou authentifiez-vous à l'aide de SSSD :

```
[idm-user@idm-client ~]$ kinit idm-user@IDM.EXAMPLE.COM
Password for idm-user@idm.example.com:
```

- Si l'une des erreurs suivantes apparaît dans le fichier journal **/var/log/sssd/sssd_pam.log**, les informations d'identification Kerberos ne correspondent pas au nom d'utilisateur de l'utilisateur actuellement connecté :

```
User with UPN [<UPN>] was not found.

UPN [<UPN>] does not match target user [<username>].
```

Dans ce cas, vérifiez que vous vous êtes authentifié avec SSSD, ou envisagez de désactiver l'option **pam_gssapi_check_upn** dans le fichier **/etc/sssd/sssd.conf**:

```
[idm-user@idm-client ~]$ cat /etc/sssd/sssd.conf
...

pam_gssapi_check_upn = false
```

- Pour un dépannage supplémentaire, vous pouvez activer la sortie de débogage pour le module PAM de **pam_sss_gss.so**.
  - Ajoutez l'option **debug** à la fin de toutes les entrées **pam_sss_gss.so** dans les fichiers PAM, telles que **/etc/pam.d/sudo** et **/etc/pam.d/sudo-i**:

```
[root@idm-client ~]# cat /etc/pam.d/sudo
#%PAM-1.0
auth       sufficient   pam_sss_gss.so   debug
auth       include      system-auth
account    include      system-auth
password   include      system-auth
session    include      system-auth
```

```
[root@idm-client ~]# cat /etc/pam.d/sudo-i
#%PAM-1.0
auth       sufficient   pam_sss_gss.so   debug
auth       include      sudo
account    include      sudo
password   include      sudo
session    optional     pam_keyinit.so force revoke
session    include      sudo
```

  - Essayez de vous authentifier avec le module **pam_sss_gss.so** et examinez la sortie de la console. Dans cet exemple, l'utilisateur n'avait pas d'identifiants Kerberos.

```
[idm-user@idm-client ~]$ sudo ls -l /etc/sssd/sssd.conf
pam_sss_gss: Initializing GSSAPI authentication with SSSD
pam_sss_gss: Switching euid from 0 to 1366201107
pam_sss_gss: Trying to establish security context
pam_sss_gss: SSSD User name: idm-user@idm.example.com
pam_sss_gss: User domain: idm.example.com
```

```
pam_sss_gss: User principal:
pam_sss_gss: Target name: host@idm.example.com
pam_sss_gss: Using ccache: KCM:
pam_sss_gss: Acquiring credentials, principal name will be derived
pam_sss_gss: Unable to read credentials from [KCM:] [maj:0xd0000, min:0x96c73ac3]
pam_sss_gss: GSSAPI: Unspecified GSS failure.  Minor code may provide more
information
pam_sss_gss: GSSAPI: No credentials cache found
pam_sss_gss: Switching euid from 1366200907 to 0
pam_sss_gss: System error [5]: Input/output error
```

8.11. UTILISATION D'UN PLAYBOOK ANSIBLE POUR GARANTIR L'ACCÈS SUDO À UN UTILISATEUR IDM SUR UN CLIENT IDM

Dans la gestion des identités (IdM), vous pouvez vous assurer que l'accès **sudo** à une commande spécifique est accordé à un compte d'utilisateur IdM sur un hôte IdM spécifique.

Effectuez cette procédure pour vous assurer qu'une règle **sudo** nommée **idm_user_reboot** existe. La règle accorde à **idm_user** la permission d'exécuter la commande **/usr/sbin/reboot** sur la machine **idmclient**.

Conditions préalables

- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.
- Vous vous êtes assuré de la présence d'un compte utilisateur pour **idm_user** dans IdM et vous avez déverrouillé le compte en créant un mot de passe pour l'utilisateur. Pour plus de détails sur l'ajout d'un nouvel utilisateur IdM à l'aide de l'interface de ligne de commande, voir le lien : Ajouter des utilisateurs à l'aide de la ligne de commande .
- Aucun compte local **idm_user** n'existe sur **idmclient**. L'utilisateur **idm_user** n'est pas listé dans le fichier **/etc/passwd** sur **idmclient**.

Procédure

1. Créez un fichier d'inventaire, par exemple **inventory.file**, et définissez-y **ipaservers**:

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

2. Ajouter une ou plusieurs commandes **sudo**:
   a. Créer un playbook Ansible **ensure-reboot-sudocmd-is-present.yml** qui assure la présence de la commande **/usr/sbin/reboot** dans la base de données IdM des commandes **sudo**.

Pour simplifier cette étape, vous pouvez copier et modifier l'exemple dans le fichier **/usr/share/doc/ansible-freeipa/playbooks/sudocmd/ensure-sudocmd-is-present.yml**:

```
---
- name: Playbook to manage sudo command
  hosts: ipaserver

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  # Ensure sudo command is present
  - ipasudocmd:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: /usr/sbin/reboot
      state: present
```

b. Exécutez le manuel de jeu :

```
$ ansible-playbook --vault-password-file=password_file -v -i
path_to_inventory_directory/inventory.file path_to_playbooks_directory/ensure-
reboot-sudocmd-is-present.yml
```

3. Créez une règle **sudo** qui fait référence aux commandes :

a. Créez un playbook Ansible **ensure-sudorule-for-idmuser-on-idmclient-is-present.yml** qui utilise l'entrée de commande **sudo** pour s'assurer de la présence d'une règle sudo. La règle sudo permet à **idm_user** de redémarrer la machine **idmclient**. Pour simplifier cette étape, vous pouvez copier et modifier l'exemple dans le fichier **/usr/share/doc/ansible-freeipa/playbooks/sudorule/ensure-sudorule-is-present.yml**:

```
---
- name: Tests
  hosts: ipaserver

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  # Ensure a sudorule is present granting idm_user the permission to run /usr/sbin/reboot
on idmclient
  - ipasudorule:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: idm_user_reboot
      description: A test sudo rule.
      allow_sudocmd: /usr/sbin/reboot
      host: idmclient.idm.example.com
      user: idm_user
      state: present
```

b. Exécutez le manuel de jeu :

```
$ ansible-playbook -v -i path_to_inventory_directory/inventory.file
path_to_playbooks_directory/ensure-sudorule-for-idmuser-on-idmclient-is-
present.yml
```

Verification steps

Testez que la règle **sudo** dont vous avez assuré la présence sur le serveur IdM fonctionne sur **idmclient** en vérifiant que **idm_user** peut redémarrer **idmclient** à l'aide de **sudo**. Notez qu'il peut s'écouler quelques minutes avant que les changements effectués sur le serveur ne prennent effet sur le client.

1. Connectez-vous à **idmclient** en tant que **idm_user**.

2. Redémarrez la machine en utilisant **sudo**. Saisissez le mot de passe de **idm_user** lorsque vous y êtes invité :

```
$ sudo /usr/sbin/reboot
[sudo] password for idm_user:
```

Si **sudo** est configuré correctement, la machine redémarre.

Ressources supplémentaires

- Voir les fichiers **README-sudocmd.md**, **README-sudocmdgroup.md**, et **README-sudorule.md** dans le répertoire **/usr/share/doc/ansible-freeipa/**.

CHAPITRE 9. UTILISATION DE LDAPMODIFY POUR GÉRER LES UTILISATEURS IDM EN EXTERNE

En tant qu'administrateur IdM, vous pouvez utiliser les commandes **ipa** pour gérer le contenu de votre annuaire. Vous pouvez également utiliser la commande **ldapmodify** pour atteindre des objectifs similaires. Vous pouvez utiliser cette commande de manière interactive et fournir toutes les données directement dans la ligne de commande. Vous pouvez également fournir les données du fichier au format LDAP Data Interchange Format (LDIF) à la commande **ldapmodify**.

9.1. MODÈLES POUR LA GESTION EXTERNE DES COMPTES D'UTILISATEURS IDM

Cette section décrit des modèles pour diverses opérations de gestion des utilisateurs dans IdM. Les modèles indiquent les attributs que vous devez modifier à l'aide de **ldapmodify** pour atteindre les objectifs suivants :

- Ajout d'un nouvel utilisateur de l'étape
- Modifier l'attribut d'un utilisateur
- Activation d'un utilisateur
- Désactivation d'un utilisateur
- Préserver un utilisateur

Les modèles sont formatés dans le format d'échange de données LDAP (LDIF). LDIF est un format standard d'échange de données en texte clair pour représenter le contenu de l'annuaire LDAP et les demandes de mise à jour.

À l'aide des modèles, vous pouvez configurer le fournisseur LDAP de votre système de provisionnement pour gérer les comptes d'utilisateurs IdM.

Pour des exemples détaillés de procédures, voir les sections suivantes :

- Ajout d'une étape IdM définie par l'utilisateur dans un fichier LDIF
- Ajout d'un utilisateur d'étape IdM directement à partir de l'interface de dialogue en ligne à l'aide de ldapmodify
- Préservation d'un utilisateur IdM avec ldapmodify

Modèles pour l'ajout d'un nouvel utilisateur de l'étape

- Un modèle pour ajouter un utilisateur avec **UID and GID assigned automatically**. Le nom distinctif (DN) de l'entrée créée doit commencer par **uid=user_login**:

```
dn: uid=user_login,cn=staged
users,cn=accounts,cn=provisioning,dc=idm,dc=example,dc=com
changetype: add
objectClass: top
objectClass: inetorgperson
uid: user_login
```

```
sn: surname
givenName: first_name
cn: full_name
```

- Un modèle pour ajouter un utilisateur avec **UID and GID assigned statically**:

```
dn: uid=user_login,cn=staged
users,cn=accounts,cn=provisioning,dc=idm,dc=example,dc=com
changetype: add
objectClass: top
objectClass: person
objectClass: inetorgperson
objectClass: organizationalperson
objectClass: posixaccount
uid: user_login
uidNumber: UID_number
gidNumber: GID_number
sn: surname
givenName: first_name
cn: full_name
homeDirectory: /home/user_login
```

Il n'est pas nécessaire de spécifier des classes d'objets IdM lors de l'ajout d'utilisateurs de scène. L'IdM ajoute ces classes automatiquement après l'activation des utilisateurs.

Modèles pour la modification des utilisateurs existants

- **Modifying a user's attribute**:

```
dn: distinguished_name
changetype: modify
replace: attribute_to_modify
attribute_to_modify: new_value
```

- **Disabling a user**:

```
dn: distinguished_name
changetype: modify
replace: nsAccountLock
nsAccountLock: TRUE
```

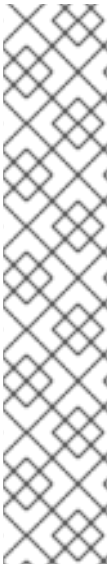
- **Enabling a user**:

```
dn: distinguished_name
changetype: modify
replace: nsAccountLock
nsAccountLock: FALSE
```

La mise à jour de l'attribut **nssAccountLock** n'a aucun effet sur les utilisateurs de l'étape et les utilisateurs préservés. Même si l'opération de mise à jour se termine avec succès, la valeur de l'attribut reste **nssAccountLock: TRUE**.

- **Preserving a user**:

```
dn: distinguished_name
changetype: modrdn
newrdn: uid=user_login
deleteoldrdn: 0
newsuperior: cn=deleted users,cn=accounts,cn=provisioning,dc=idm,dc=example,dc=com
```



NOTE

Avant de modifier un utilisateur, obtenez son nom distinctif (DN) en effectuant une recherche à l'aide de son login. Dans l'exemple suivant, l'utilisateur *user_allowed_to_modify_user_entries* est un utilisateur autorisé à modifier les informations relatives aux utilisateurs et aux groupes, par exemple **activator** ou l'administrateur IdM. Le mot de passe dans l'exemple est le mot de passe de cet utilisateur :

```
[...]
# ldapsearch -LLL -x -D
"uid=user_allowed_to_modify_user_entries,cn=users,cn=accounts,dc=idm,dc=example,dc=com" -w "Secret123" -H ldap://r8server.idm.example.com -b
"cn=users,cn=accounts,dc=idm,dc=example,dc=com" uid=test_user
dn: uid=test_user,cn=users,cn=accounts,dc=idm,dc=example,dc=com
memberOf: cn=ipausers,cn=groups,cn=accounts,dc=idm,dc=example,dc=com
```

9.2. MODÈLES POUR LA GESTION EXTERNE DES COMPTES DE GROUPE IDM

Cette section décrit des modèles pour diverses opérations de gestion des groupes d'utilisateurs dans IdM. Les modèles indiquent les attributs que vous devez modifier à l'aide de **ldapmodify** pour atteindre les objectifs suivants :

- Création d'un nouveau groupe
- Suppression d'un groupe existant
- Ajouter un membre à un groupe
- Supprimer un membre d'un groupe

Les modèles sont formatés dans le format d'échange de données LDAP (LDIF). LDIF est un format standard d'échange de données en texte clair pour représenter le contenu de l'annuaire LDAP et les demandes de mise à jour.

À l'aide des modèles, vous pouvez configurer le fournisseur LDAP de votre système de provisionnement pour gérer les comptes de groupe IdM.

Création d'un nouveau groupe

```
dn: cn=group_name,cn=groups,cn=accounts,dc=idm,dc=example,dc=com
changetype: add
objectClass: top
objectClass: ipaobject
objectClass: ipausergroup
objectClass: groupofnames
objectClass: nestedgroup
objectClass: posixgroup
```

```
uid: group_name
cn: group_name
gidNumber: GID_number
```

Modification des groupes

- **Deleting an existing group**:

```
dn: group_distinguished_name
changetype: delete
```

- **Adding a member to a group**:

```
dn: group_distinguished_name
changetype: modify
add: member
member: uid=user_login,cn=users,cn=accounts,dc=idm,dc=example,dc=com
```

N'ajoutez pas d'utilisateurs en stage ou préservés aux groupes. Même si l'opération de mise à jour se termine avec succès, les utilisateurs ne seront pas mis à jour en tant que membres du groupe. Seuls les utilisateurs actifs peuvent appartenir à des groupes.

- **Removing a member from a group**:

```
dn: distinguished_name
changetype: modify
delete: member
member: uid=user_login,cn=users,cn=accounts,dc=idm,dc=example,dc=com
```

NOTE

Avant de modifier un groupe, obtenez son nom distinctif (DN) en effectuant une recherche à l'aide de son nom.

```
# ldapsearch -YGSSAPI -H ldap://server.idm.example.com -b
"cn=groups,cn=accounts,dc=idm,dc=example,dc=com" "cn=group_name"
dn: cn=group_name,cn=groups,cn=accounts,dc=idm,dc=example,dc=com
ipaNTSecurityIdentifier: S-1-5-21-1650388524-2605035987-2578146103-11017
cn: testgroup
objectClass: top
objectClass: groupofnames
objectClass: nestedgroup
objectClass: ipausergroup
objectClass: ipaobject
objectClass: posixgroup
objectClass: ipantgroupattrs
ipaUniqueID: 569bf864-9d45-11ea-bea3-525400f6f085
gidNumber: 1997010017
```

9.3. UTILISATION INTERACTIVE DE LA COMMANDE LDAPMODIFY

Vous pouvez modifier les entrées LDAP (Lightweight Directory Access Protocol) en mode interactif.

Procédure

1. Dans une ligne de commande, saisissez l'instruction LDAP Data Interchange Format (LDIF) après la commande **ldapmodify**.

   Exemple 9.1. Modifier le numéro de téléphone d'untestuser

   ```
   # ldapmodify -Y GSSAPI -H ldap://server.example.com
   dn: uid=testuser,cn=users,cn=accounts,dc=example,dc=com
   changetype: modify
   replace: telephoneNumber
   telephonenumber: 88888888
   ```

   Notez que vous devez obtenir un ticket Kerberos pour utiliser l'option **-Y**.

2. Appuyez sur **Ctlr D** pour quitter le mode interactif.

3. Vous pouvez également fournir un fichier LDIF après la commande **ldapmodify**:

   Exemple 9.2. La commande ldapmodify permet de lire les données de modification d'un fichier LDIF

   ```
   # ldapmodify -Y GSSAPI -H ldap://server.example.com -f ~/example.ldif
   ```

Ressources supplémentaires

- Pour plus d'informations sur l'utilisation de la commande **ldapmodify**, voir la page de manuel **ldapmodify(1)**.
- Pour plus d'informations sur la structure **LDIF**, voir la page de manuel **ldif(5)**.

9.4. PRÉSERVATION D'UN UTILISATEUR IDM AVEC LDAPMODIFY

Cette section décrit comment utiliser **ldapmodify** pour préserver un utilisateur IdM, c'est-à-dire comment désactiver un compte utilisateur après que l'employé a quitté l'entreprise.

Conditions préalables

- Vous pouvez vous authentifier en tant qu'utilisateur IdM avec un rôle de préservation des utilisateurs.

Procédure

1. Se connecter en tant qu'utilisateur IdM avec un rôle de préservation des utilisateurs :

   ```
   kinit admin
   ```

2. Entrez dans la commande **ldapmodify** et indiquez Generic Security Services API (GSSAPI) comme mécanisme SASL (Simple Authentication and Security Layer) à utiliser pour l'authentification :

   ```
   # ldapmodify -Y GSSAPI
   ```

```
SASL/GSSAPI authentication started
SASL username: admin@IDM.EXAMPLE.COM
SASL SSF: 256
SASL data security layer installed.
```

3. Saisissez l'adresse **dn** de l'utilisateur que vous souhaitez préserver :

```
dn : uid=user1,cn=users,cn=accounts,dc=idm,dc=example,dc=com
```

4. Saisissez **modrdn** comme type de modification à effectuer :

```
changetype : modrdn
```

5. Spécifiez l'adresse **newrdn** pour l'utilisateur :

```
newrdn : uid=user1
```

6. Indiquez que vous souhaitez préserver l'utilisateur :

```
supprimeroldrdn : 0
```

7. Spécifiez le site **new superior DN**:

```
newsuperior : cn=deleted users,cn=accounts,cn=provisioning,dc=idm,dc=example,dc=com
```

La préservation d'un utilisateur déplace l'entrée vers un nouvel emplacement dans l'arborescence des informations de répertoire (DIT). C'est pourquoi vous devez spécifier le DN de la nouvelle entrée parent comme nouveau DN supérieur.

8. Appuyez à nouveau sur **Enter** pour confirmer la fin de l'entrée :

```
[Enter]

modifying rdn of entry "uid=user1,cn=users,cn=accounts,dc=idm,dc=example,dc=com"
```

9. Quittez la connexion en utilisant **Ctrl C** .

Verification steps

- Vérifiez que l'utilisateur a été préservé en dressant la liste de tous les utilisateurs préservés :

```
$ ipa user-find --preserved=true
--------------
1 user matched
--------------
  User login: user1
  First name: First 1
  Last name: Last 1
  Home directory: /home/user1
  Login shell: /bin/sh
  Principal name: user1@IDM.EXAMPLE.COM
  Principal alias: user1@IDM.EXAMPLE.COM
  Email address: user1@idm.example.com
```

```
  UID: 1997010003
  GID: 1997010003
  Account disabled: True
  Preserved user: True
----------------------------
Number of entries returned 1
----------------------------
```

CHAPITRE 10. RECHERCHE D'ENTRÉES IDM À L'AIDE DE LA COMMANDE LDAPSEARCH

Vous pouvez utiliser la commande **ipa find** pour effectuer une recherche dans les entrées de gestion de l'identité. Pour plus d'informations sur la commande **ipa**, voir la section Structure des commandes IPA.

Cette section présente les bases d'une option de recherche alternative à l'aide de la commande en ligne **ldapsearch** par le biais des entrées de gestion de l'identité.

10.1. UTILISATION DE LA COMMANDE LDAPSEARCH

La commande **ldapsearch** a le format suivant :

```
# ldapsearch [-x | -Y mechanism] [options] [search_filter] [list_of_attributes]
```

- Pour configurer la méthode d'authentification, spécifiez l'option **-x** pour utiliser des liaisons simples ou l'option **-Y** pour définir le mécanisme SASL (Simple Authentication and Security Layer). Notez que vous devez obtenir un ticket Kerberos si vous utilisez l'option **-Y GSSAPI**.
- Les options de la commande *options* sont les options de la commande **ldapsearch** décrites dans le tableau ci-dessous.
- Le site *search_filter* est un filtre de recherche LDAP.
- Le site *list_of_attributes* est une liste des attributs que les résultats de la recherche renvoient.

Par exemple, vous souhaitez rechercher le nom d'utilisateur *user01* dans toutes les entrées d'une arborescence LDAP de base :

```
# ldapsearch -x -H ldap://ldap.example.com -s sub "(uid=user01)"
```

- L'option **-x** indique à la commande **ldapsearch** de s'authentifier avec le simple bind. Notez que si vous ne fournissez pas le Distinguish Name (DN) avec l'option **-D**, l'authentification est anonyme.
- L'option **-H** vous connecte au site *ldap://ldap.example.com*.
- L'option **-s sub** indique à la commande **ldapsearch** de rechercher dans toutes les entrées, à partir du DN de base, l'utilisateur portant le nom *user01*. L'option *"(uid=user01)"* est un filtre.

Notez que si vous n'indiquez pas le point de départ de la recherche avec l'option **-b**, la commande recherche dans l'arbre par défaut. Il est spécifié dans le paramètre BASE du fichier **etc/openldap/ldap.conf**.

Tableau 10.1. Les options de la commande ldapsearch

Option	Description

Option	Description
-b	Le point de départ de la recherche. Si vos paramètres de recherche contiennent un astérisque (*) ou un autre caractère que la ligne de commande peut interpréter comme un code, vous devez mettre la valeur entre guillemets simples ou doubles. Par exemple, **-b cn=user,ou=Product Development,dc=example,dc=com**.
-D	Le Distinguished Name (DN) avec lequel vous souhaitez vous authentifier.
-H	Une URL LDAP pour se connecter au serveur. L'option **-H** remplace les options **-h** et **-p**.
-l	Délai en secondes pour attendre la fin d'une demande de recherche.
-s *scope*	L'étendue de la recherche. Vous pouvez choisir l'un des éléments suivants pour le champ d'application : • **base** ne recherche que l'entrée de l'option **-b** ou définie par la variable d'environnement **LDAP_BASEDN**. • **one** recherche uniquement les enfants de l'entrée de l'option **-b**. • **sub** une recherche de sous-arbres à partir du point de départ de l'option -b.
-W	Demande de mot de passe.
-x	Désactive la connexion SASL par défaut pour permettre des liaisons simples.
-Y *SASL_mechanism*	Définit le mécanisme SASL pour l'authentification.
-z *number*	Nombre maximum d'entrées dans le résultat de la recherche.

Remarque : vous devez spécifier l'un des mécanismes d'authentification avec l'option **-x** ou **-Y** dans la commande **ldapsearch**.

Ressources supplémentaires

- Pour plus de détails sur l'utilisation de **ldapsearch**, voir la page de manuel **ldapsearch(1)**.

10.2. UTILISATION DES FILTRES LDAPSEARCH

Les filtres du site **ldapsearch** vous permettent d'affiner les résultats de la recherche.

Par exemple, vous souhaitez que le résultat de la recherche contienne toutes les entrées dont le nom commun est *example*:

```
"(cn=example)"
```

Dans ce cas, *equal sign (=)* est l'opérateur et *example* est la valeur.

Tableau 10.2. Les opérateurs de filtrage ldapsearch

Type de recherche	Opérateur	Description
L'égalité	=	Renvoie les entrées qui correspondent exactement à la valeur. Par exemple, *cn=example*.
Sous-chaîne	*=string* string*	Renvoie toutes les entrées contenant la sous-chaîne correspondante. Par exemple, *cn=exa*l*. L'astérisque (*) indique zéro (0) ou plusieurs caractères.
Supérieur ou égal à	>=	Renvoie toutes les entrées dont les attributs sont supérieurs ou égaux à la valeur. Par exemple, *uidNumber >= 5000*.
Inférieur ou égal à	<=	Renvoie toutes les entrées dont les attributs sont inférieurs ou égaux à la valeur. Par exemple, *uidNumber <= 5000*.
Présence	=*	Renvoie toutes les entrées avec un ou plusieurs attributs. Par exemple, *cn=**.
Approximation	~=	Renvoie toutes les entrées dont les attributs sont similaires à ceux de la valeur. Par exemple, *l~=san fransico* peut renvoyer *l=san francisco*.

Vous pouvez utiliser les opérateurs *boolean* pour combiner plusieurs filtres à la commande **ldapsearch**.

Tableau 10.3. Les opérateurs booléens du filtre ldapsearch

Type de recherche	Opérateur	Description
ET	&	Renvoie toutes les entrées pour lesquelles toutes les affirmations des filtres sont vraies. Par exemple, *(&(filter)(filter)(filter)...)*.
OU	|	Renvoie toutes les entrées pour lesquelles au moins une déclaration dans les filtres est vraie. Par exemple, *(|(filter)(filter)(filter)...)*.
PAS	!	Renvoie toutes les entrées pour lesquelles l'énoncé du filtre n'est pas vrai. Par exemple, *(!(filter))*.

CHAPITRE 11. CONFIGURATION DE L'IDM POUR LE PROVISIONNEMENT EXTERNE DES UTILISATEURS

En tant qu'administrateur système, vous pouvez configurer la gestion des identités (IdM) pour prendre en charge le provisionnement des utilisateurs par une solution externe de gestion des identités.

Plutôt que d'utiliser l'utilitaire **ipa**, l'administrateur du système de provisionnement externe peut accéder au LDAP IdM à l'aide de l'utilitaire **ldapmodify**. L'administrateur peut ajouter des utilisateurs de scène individuels à partir de l'interface de ligne de commande en utilisant ldapmodify ou en utilisant un fichier LDIF.

L'hypothèse est que vous, en tant qu'administrateur IdM, faites entièrement confiance à votre système de provisionnement externe pour n'ajouter que des utilisateurs validés. Toutefois, vous ne souhaitez pas attribuer aux administrateurs du système de provisionnement externe le rôle IdM de **User Administrator** pour leur permettre d'ajouter directement de nouveaux utilisateurs actifs.

Vous pouvez configurer un script pour déplacer automatiquement les utilisateurs en phase créés par le système de provisionnement externe vers des utilisateurs actifs.

Ce chapitre contient les sections suivantes :

1. Préparation de la gestion des identités (IdM) à l'utilisation d'un système de provisionnement externe pour ajouter des utilisateurs d'étape à IdM.
2. Création d'un script pour faire passer les utilisateurs ajoutés par le système de provisionnement externe du statut d'utilisateurs actifs à celui d'utilisateurs actifs.
3. Utilisation d'un système de provisionnement externe pour ajouter un utilisateur de l'étape IdM. Vous pouvez le faire de deux manières :
   - Ajouter un utilisateur de l'étape IdM à l'aide d'un fichier LDIF
   - Ajouter un utilisateur d'étape IdM directement à partir de l'interface de gestion en utilisant ldapmodify

11.1. PRÉPARATION DES COMPTES IDM POUR L'ACTIVATION AUTOMATIQUE DES COMPTES D'UTILISATEURS DE L'ÉTAPE

Cette procédure montre comment configurer deux comptes d'utilisateur IdM pour qu'ils soient utilisés par un système de provisionnement externe. En ajoutant les comptes à un groupe avec une politique de mot de passe appropriée, vous permettez au système de provisionnement externe de gérer le provisionnement des utilisateurs dans IdM. Dans ce qui suit, le compte d'utilisateur à utiliser par le système externe pour ajouter des utilisateurs de stage est nommé **provisionator**. Le compte d'utilisateur à utiliser pour activer automatiquement les utilisateurs de l'étape est nommé **activator**.

Conditions préalables

- L'hôte sur lequel vous effectuez la procédure est enrôlé dans IdM.

Procédure

1. Se connecter en tant qu'administrateur IdM :

```
kinit admin
```

2. Créez un utilisateur nommé **provisionator** avec les privilèges d'ajouter des utilisateurs de scène.

   a. Ajouter le compte utilisateur du provisionneur :

```
ipa user-add provisionator --first=provisioning --last=account --password
```

   a. Accorder à l'utilisateur du provisionneur les privilèges requis.

      i. Créez un rôle personnalisé, **System Provisioning**, pour gérer l'ajout d'utilisateurs de l'étape :

```
$ ipa role-add --desc \N- "Responsable de l'approvisionnement des utilisateurs de l'étape\N" \N- "Provisionnement du système\N"
```

      ii. Ajoutez le privilège **Stage User Provisioning** au rôle. Ce privilège permet d'ajouter des utilisateurs de scène :

```
$ ipa role-add-privilege \N "System Provisioning" --privileges=\N "Stage User Provisioning" $ ipa role-add-privilege \N "System Provisioning" --privileges=\N "Stage User Provisioning"
```

      iii. Ajouter l'utilisateur du provisionneur au rôle :

```
$ ipa role-add-member --users=provisionator "System Provisioning"
```

      iv. Vérifier que le provisionneur existe dans IdM :

```
$ ipa user-find provisionator --all --raw
--------------
1 user matched
--------------
  dn: uid=provisionator,cn=users,cn=accounts,dc=idm,dc=example,dc=com
  uid: provisionator
  [...]
```

3. Créez un utilisateur, **activator**, avec les privilèges nécessaires pour gérer les comptes d'utilisateurs.

   a. Ajouter le compte utilisateur de l'activateur :

```
$ ipa user-add activator --first=activation --last=account --password
```

   b. Accordez à l'utilisateur de l'activateur les privilèges requis en ajoutant l'utilisateur au rôle par défaut **User Administrator**:

```
$ ipa role-add-member --users=activator "User Administrator"
```

4. Créer un groupe d'utilisateurs pour les comptes d'application :

```
ipa group-add application-accounts
```

5. Mettez à jour la politique de mot de passe pour le groupe. La politique suivante empêche l'expiration du mot de passe et le verrouillage du compte, mais compense les risques potentiels en exigeant des mots de passe complexes :

```
$ ipa pwpolicy-add application-accounts --maxlife=10000 --minlife=0 --history=0 --
minclasses=4 --minlength=8 --priority=1 --maxfail=0 --failinterval=1 --lockouttime=0
```

6. (Facultatif) Vérifier que la politique de mot de passe existe dans IdM :

```
$ ipa pwpolicy-show application-accounts
  Group: application-accounts
  Max lifetime (days): 10000
  Min lifetime (hours): 0
  History size: 0
[...]
```

7. Ajoutez les comptes de provisionnement et d'activation au groupe des comptes d'application :

```
$ ipa group-add-member application-accounts --users={provisionator,activator}
```

8. Modifier les mots de passe des comptes utilisateurs :

```
$ kpasswd provisionator
$ kpasswd activator
```

   La modification des mots de passe est nécessaire car les mots de passe des nouveaux utilisateurs de l'IdM expirent immédiatement.

Ressources complémentaires :

- Voir Gestion des comptes d'utilisateurs à l'aide de la ligne de commande .
- Voir Délégation de droits sur les utilisateurs .
- Voir Définition des politiques de mot de passe IdM .

11.2. CONFIGURATION DE L'ACTIVATION AUTOMATIQUE DES COMPTES D'UTILISATEURS DE L'ÉTAPE IDM

Cette procédure montre comment créer un script pour activer les utilisateurs de l'étape. Le système exécute automatiquement le script à des intervalles de temps spécifiés. Cela garantit que les nouveaux comptes d'utilisateurs sont automatiquement activés et disponibles peu de temps après leur création.

IMPORTANT

La procédure suppose que le propriétaire du système de provisionnement externe a déjà validé les utilisateurs et qu'ils n'ont pas besoin d'une validation supplémentaire du côté de l'IdM avant que le script ne les ajoute à l'IdM.

Il suffit d'activer le processus d'activation sur un seul de vos serveurs IdM.

Conditions préalables

- Les comptes **provisionator** et **activator** existent dans IdM. Pour plus de détails, voir Préparation des comptes IdM pour l'activation automatique des comptes d'utilisateurs de stage .
- Vous disposez des droits de root sur le serveur IdM sur lequel vous exécutez la procédure.
- Vous êtes connecté en tant qu'administrateur IdM.
- Vous faites confiance à votre système de provisionnement externe.

Procédure

1. Générer un fichier keytab pour le compte d'activation :

```
# ipa-getkeytab -s server.idm.example.com -p "activator" -k /etc/krb5.ipa-activation.keytab
```

   Si vous souhaitez activer le processus d'activation sur plusieurs serveurs IdM, générez le fichier keytab sur un seul serveur. Copiez ensuite le fichier keytab sur les autres serveurs.

2. Créez un script, **/usr/local/sbin/ipa-activate-all**, avec le contenu suivant pour activer tous les utilisateurs :

```
#!/bin/bash

kinit -k -i activator

ipa stageuser-find --all --raw | grep "  uid:" | cut -d ":" -f 2 | while read uid; do ipa stageuser-activate ${uid}; done
```

3. Modifiez les autorisations et la propriété du script **ipa-activate-all** pour le rendre exécutable :

```
# chmod 755 /usr/local/sbin/ipa-activate-all
# chown root:root /usr/local/sbin/ipa-activate-all
```

4. Créez un fichier d'unité systemd, **/etc/systemd/system/ipa-activate-all.service**, avec le contenu suivant :

```
[Unit]
Description=Scan IdM every minute for any stage users that must be activated

[Service]
Environment=KRB5_CLIENT_KTNAME=/etc/krb5.ipa-activation.keytab
Environment=KRB5CCNAME=FILE:/tmp/krb5cc_ipa-activate-all
ExecStart=/usr/local/sbin/ipa-activate-all
```

5. Créez un timer systemd, **/etc/systemd/system/ipa-activate-all.timer**, avec le contenu suivant :

```
[Unit]
Description=Scan IdM every minute for any stage users that must be activated

[Timer]
OnBootSec=15min
OnUnitActiveSec=1min

[Install]
WantedBy=multi-user.target
```

6. Recharger la nouvelle configuration :

```
# systemctl daemon-reload
```

7. Activer **ipa-activate-all.timer**:

```
# systemctl enable ipa-activate-all.timer
```

8. Démarrer **ipa-activate-all.timer**:

```
# systemctl start ipa-activate-all.timer
```

9. (Facultatif) Vérifiez que le démon **ipa-activate-all.timer** est en cours d'exécution :

```
# systemctl status ipa-activate-all.timer
● ipa-activate-all.timer - Scan IdM every minute for any stage users that must be activated
   Loaded: loaded (/etc/systemd/system/ipa-activate-all.timer; enabled; vendor preset:
disabled)
   Active: active (waiting) since Wed 2020-06-10 16:34:55 CEST; 15s ago
  Trigger: Wed 2020-06-10 16:35:55 CEST; 44s left

Jun 10 16:34:55 server.idm.example.com systemd[1]: Started Scan IdM every minute for any
stage users that must be activated.
```

11.3. AJOUT D'UNE ÉTAPE IDM DÉFINIE PAR L'UTILISATEUR DANS UN FICHIER LDIF

Cette section décrit comment un administrateur d'un système de provisionnement externe peut accéder à IdM LDAP et utiliser un fichier LDIF pour ajouter des utilisateurs d'étape. L'exemple ci-dessous montre l'ajout d'un seul utilisateur, mais plusieurs utilisateurs peuvent être ajoutés dans un fichier en mode groupé.

Conditions préalables

- L'administrateur IdM a créé le compte **provisionator** et un mot de passe. Pour plus de détails, voir Préparation des comptes IdM pour l'activation automatique des comptes d'utilisateurs de scène.
- En tant qu'administrateur externe, vous connaissez le mot de passe du compte **provisionator**.
- Vous pouvez accéder au serveur IdM par SSH à partir de votre serveur LDAP.
- Vous êtes en mesure de fournir l'ensemble minimal d'attributs qu'un utilisateur de la phase IdM doit posséder pour permettre le traitement correct du cycle de vie de l'utilisateur, à savoir :
  - Le site **distinguished name** (dn)
  - The **common name** (cn)
  - The **last name** (sn)
  - Les **uid**

Procédure

1. Sur le serveur externe, créez un fichier LDIF contenant des informations sur le nouvel utilisateur :

```
dn: uid=stageidmuser,cn=staged
users,cn=accounts,cn=provisioning,dc=idm,dc=example,dc=com
changetype: add
objectClass: top
objectClass: inetorgperson
uid: stageidmuser
sn: surname
givenName: first_name
cn: full_name
```

2. Transférer le fichier LDIF du serveur externe vers le serveur IdM :

```
$ scp add-stageidmuser.ldif provisionator@server.idm.example.com:/provisionator/
Password:
add-stageidmuser.ldif                                                    100%  364
217.6KB/s   00:00
```

3. Utilisez le protocole **SSH** pour vous connecter au serveur IdM en tant que **provisionator**:

```
$ ssh provisionator@server.idm.example.com
Password:
[provisionator@server ~]$
```

4. Sur le serveur IdM, obtenez le ticket Kerberos (TGT) pour le compte du provisionneur :

```
[provisionator@server ~]$ kinit provisionator
```

5. Saisir la commande **ldapadd** avec l'option -f et le nom du fichier LDIF. Spécifiez le nom du serveur IdM et le numéro de port :

```
~]$ ldapadd -h server.idm.example.com -p 389 -f  add-stageidmuser.ldif
SASL/GSSAPI authentication started
SASL username: provisionator@IDM.EXAMPLE.COM
SASL SSF: 256
SASL data security layer installed.
adding the entry "uid=stageidmuser,cn=staged
users,cn=accounts,cn=provisioning,dc=idm,dc=example,dc=com"
```

11.4. AJOUT D'UN UTILISATEUR D'ÉTAPE IDM DIRECTEMENT À PARTIR DE L'INTERFACE DE DIALOGUE EN LIGNE À L'AIDE DE LDAPMODIFY

Cette section décrit comment un administrateur d'un système de provisionnement externe peut accéder au LDAP de la gestion des identités (IdM) et utiliser l'utilitaire **ldapmodify** pour ajouter un utilisateur de scène.

Conditions préalables

- L'administrateur IdM a créé le compte **provisionator** et un mot de passe. Pour plus de détails, voir Préparation des comptes IdM pour l'activation automatique des comptes d'utilisateurs de scène.
- En tant qu'administrateur externe, vous connaissez le mot de passe du compte **provisionator**.
- Vous pouvez accéder au serveur IdM par SSH à partir de votre serveur LDAP.
- Vous êtes en mesure de fournir l'ensemble minimal d'attributs qu'un utilisateur de la phase IdM doit posséder pour permettre le traitement correct du cycle de vie de l'utilisateur, à savoir :
  - Le site **distinguished name** (dn)
  - The **common name** (cn)
  - The **last name** (sn)
  - Les **uid**

Procédure

1. Utilisez le protocole **SSH** pour vous connecter au serveur IdM en utilisant votre identité et vos informations d'identification IdM :

```
$ ssh provisionator@server.idm.example.com
Password:
[provisionator@server ~]$
```

2. Obtenir le TGT du compte **provisionator**, un utilisateur IdM ayant le rôle d'ajouter de nouveaux utilisateurs de scène :

```
$ kinit provisionator
```

3. Entrez dans la commande **ldapmodify** et spécifiez Generic Security Services API (GSSAPI) comme mécanisme SASL (Simple Authentication and Security Layer) à utiliser pour l'authentification. Indiquez le nom du serveur IdM et le port :

```
# ldapmodify -h server.idm.example.com -p 389 -Y GSSAPI
SASL/GSSAPI authentication started
SASL username: provisionator@IDM.EXAMPLE.COM
SASL SSF: 56
SASL data security layer installed.
```

4. Saisissez l'adresse **dn** de l'utilisateur que vous ajoutez :

```
dn : uid=stageuser,cn=staged
users,cn=accounts,cn=provisioning,dc=idm,dc=example,dc=com
```

5. Saisissez **add** comme type de modification à effectuer :

```
changetype : add
```

6. Spécifier les catégories de classes d'objets LDAP requises pour permettre le traitement correct du cycle de vie de l'utilisateur :

```
objectClass: top
objectClass: inetorgperson
```

Vous pouvez spécifier des classes d'objets supplémentaires.

7. Saisissez l'adresse **uid** de l'utilisateur :

```
uid : stageuser
```

8. Saisissez l'adresse **cn** de l'utilisateur :

```
cn : Babs Jensen
```

9. Saisissez le nom de famille de l'utilisateur :

```
sn : Jensen
```

10. Appuyez à nouveau sur **Enter** pour confirmer la fin de l'entrée :

```
[Enter]

adding new entry "uid=stageuser,cn=staged
users,cn=accounts,cn=provisioning,dc=idm,dc=example,dc=com"
```

11. Quittez la connexion en utilisant **Ctrl C** .

Verification steps

Vérifiez le contenu de l'entrée de scène pour vous assurer que votre système de provisionnement a ajouté tous les attributs POSIX requis et que l'entrée de scène est prête à être activée.

- Pour afficher les attributs LDAP du nouvel utilisateur de l'étape, entrez la commande **ipa stageuser-show --all --raw**:

```
$ ipa stageuser-show stageuser --all --raw
  dn: uid=stageuser,cn=staged
users,cn=accounts,cn=provisioning,dc=idm,dc=example,dc=com
  uid: stageuser
  sn: Jensen
  cn: Babs Jensen
  has_password: FALSE
  has_keytab: FALSE
  nsaccountlock: TRUE
  objectClass: top
  objectClass: inetorgperson
  objectClass: organizationalPerson
  objectClass: person
```

1. Notez que l'utilisateur est explicitement désactivé par l'attribut **nsaccountlock**.

11.5. RESSOURCES SUPPLÉMENTAIRES

- Voir Utilisation de ldapmodify pour gérer les utilisateurs IdM en externe .

CHAPITRE 12. RENFORCER LA SÉCURITÉ DE KERBEROS AVEC LES INFORMATIONS DU PAC

Les sections suivantes expliquent comment la gestion des identités (IdM) fonctionne avec les informations du certificat d'attributs de privilèges (PAC) par défaut depuis RHEL 8.5. Vous pouvez également activer les identificateurs de sécurité (SID) dans les déploiements IdM installés avant RHEL 8.5.

12.1. UTILISATION DU CERTIFICAT D'ATTRIBUT DE PRIVILÈGE (PAC) DANS L'IDM

Pour renforcer la sécurité, RHEL Identity Management (IdM) émet désormais des tickets Kerberos avec des informations de certificat d'attribut de privilège (PAC) par défaut dans les nouveaux déploiements. Un PAC contient de nombreuses informations sur un principal Kerberos, notamment son identifiant de sécurité (SID), l'appartenance à un groupe et des informations sur le répertoire personnel.

Les SID, que Microsoft Active Directory (AD) utilise par défaut, sont des identifiants uniques au niveau mondial qui ne sont jamais réutilisés. Les SID expriment plusieurs espaces de noms : chaque domaine a un SID, qui est un préfixe dans le SID de chaque objet.

À partir de RHEL 8.5, lorsque vous installez un serveur IdM ou un réplica, le script d'installation génère par défaut des SID pour les utilisateurs et les groupes. Cela permet à IdM de travailler avec des données PAC. Si vous avez installé IdM avant RHEL 8.5 et que vous n'avez pas configuré de confiance avec un domaine AD, il se peut que vous n'ayez pas généré de SID pour vos objets IdM. Pour plus d'informations sur la génération de SID pour vos objets IdM, voir Activation des identifiants de sécurité (SID) dans IdM .

En évaluant les informations PAC dans les tickets Kerberos, vous pouvez contrôler l'accès aux ressources de manière beaucoup plus détaillée. Par exemple, le compte Administrateur d'un domaine a un SID différent de celui du compte Administrateur d'un autre domaine. Dans un environnement IdM avec une confiance dans un domaine AD, vous pouvez définir des contrôles d'accès basés sur des SID uniques au niveau mondial plutôt que sur de simples noms d'utilisateur ou UID qui peuvent se répéter à différents endroits, comme par exemple chaque compte Linux **root** ayant un UID de 0.

12.2. ACTIVATION DES IDENTIFICATEURS DE SÉCURITÉ (SID) DANS L'IDM

Si vous avez installé IdM avant RHEL 8.5 et que vous n'avez pas configuré de confiance avec un domaine AD, il se peut que vous n'ayez pas généré d'identifiants de sécurité (SID) pour vos objets IdM. En effet, auparavant, le seul moyen de générer des SID était d'exécuter la commande **ipa-adtrust-install** pour ajouter le rôle **Trust Controller** à un serveur IdM.

À partir de RHEL 8.6, Kerberos dans IdM exige que vos objets IdM aient des SID, qui sont nécessaires pour la sécurité basée sur les informations du certificat d'accès aux privilèges (PAC).

Conditions préalables

- Vous avez installé IdM avant RHEL 8.5.
- Vous n'avez pas exécuté la tâche **ipa-sidgen**, qui fait partie de la configuration d'une confiance avec un domaine Active Directory.
- Vous pouvez vous authentifier en tant que compte administrateur IdM.

Procédure

- Activez l'utilisation des SID et déclenchez la tâche **SIDgen** pour générer des SID pour les utilisateurs et les groupes existants. Cette tâche peut être gourmande en ressources :

```
[root@server ~]# ipa config-mod --enable-sid --add-sids
```

Vérification

- Vérifiez que l'entrée du compte utilisateur IdM **admin** a un attribut **ipantsecurityidentifier** avec un SID qui se termine par **-500**, le SID réservé à l'administrateur du domaine :

```
[root@server ~]# ipa user-show admin --all | grep ipantsecurityidentifier
  ipantsecurityidentifier: S-1-5-21-2633809701-976279387-419745629-500
```

Ressources supplémentaires

- Utilisation du certificat d'attribut de privilège (PAC) dans l'IdM
- Contrôleurs et agents de confiance
- Intégrer la configuration du SID dans les installateurs de base de l'IPA

CHAPITRE 13. GESTION DES POLITIQUES DE TICKETS KERBEROS

Les politiques de tickets Kerberos dans la gestion des identités (IdM) définissent des restrictions sur l'accès, la durée et le renouvellement des tickets Kerberos. Vous pouvez configurer les politiques de tickets Kerberos pour le Centre de distribution de clés (KDC) fonctionnant sur votre serveur IdM.

Ce chapitre présente les sujets et les tâches suivants relatifs à la gestion des tickets Kerberos :

- Le rôle du KDC IdM
- Types de politiques de ticket IdM Kerberos
- Indicateurs d'authentification Kerberos
- Renforcement des indicateurs d'authentification pour un service IdM
- Configuration de la politique globale de cycle de vie des tickets
- Configuration des politiques de tickets globales par indicateur d'authentification
- Configuration de la politique de billetterie par défaut pour un utilisateur
- Configurer les politiques de tickets des indicateurs d'authentification individuels pour un utilisateur
- Options de l'indicateur d'authentification pour la commande **krbtpolicy-mod**

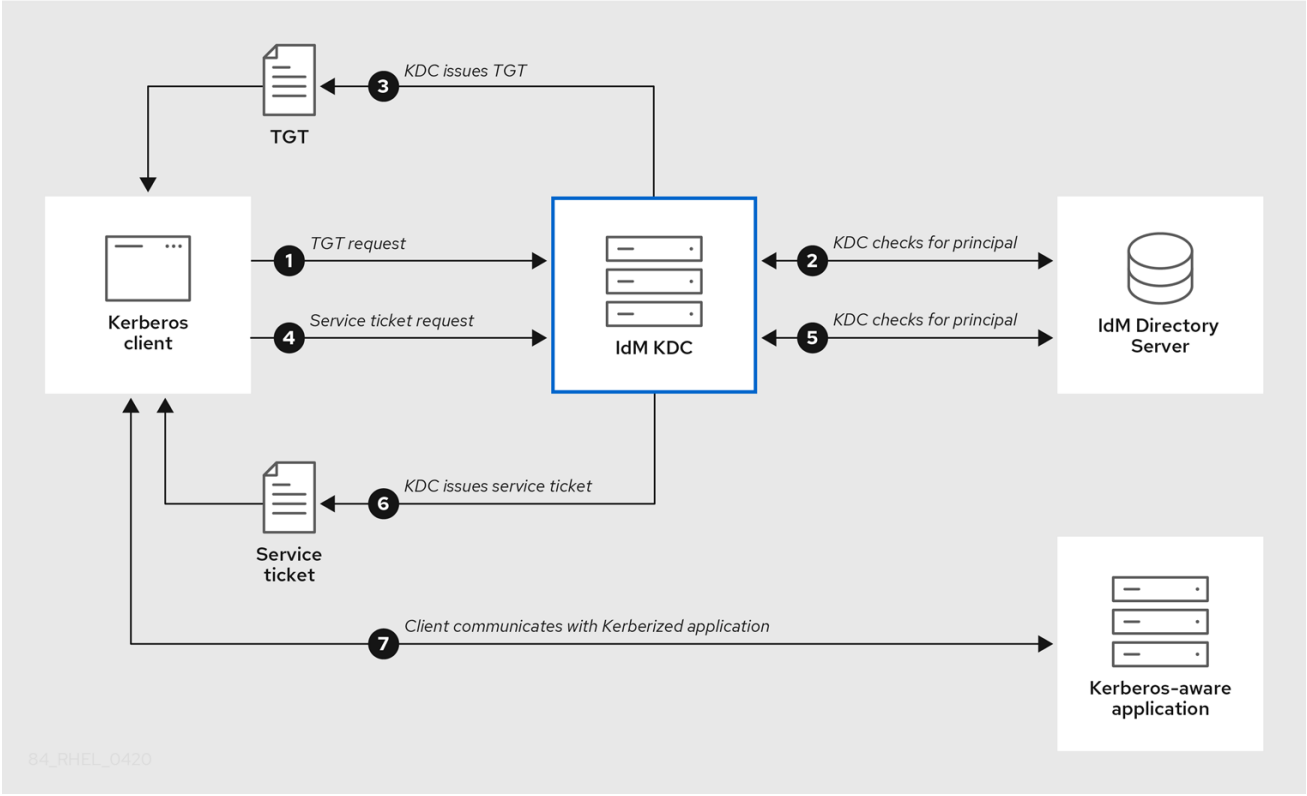
13.1. LE RÔLE DU KDC IDM

Les mécanismes d'authentification de la gestion de l'identité utilisent l'infrastructure Kerberos établie par le centre de distribution de clés (KDC). Le KDC est l'autorité de confiance qui stocke les informations d'identification et garantit l'authenticité des données provenant des entités du réseau IdM.

Chaque utilisateur, service et hôte IdM agit comme un client Kerberos et est identifié par une adresse Kerberos unique *principal*:

- Pour les utilisateurs : **identifier@REALM**, tels que **admin@EXAMPLE.COM**
- Pour les services : **service/fully-qualified-hostname@REALM**, tels que **http/server.example.com@EXAMPLE.COM**
- Pour les hôtes : **host/fully-qualified-hostname@REALM**, tels que **host/client.example.com@EXAMPLE.COM**

L'image suivante est une simplification de la communication entre un client Kerberos, le KDC et une application Kerberisée avec laquelle le client veut communiquer.



1. Un client Kerberos s'identifie auprès du KDC en s'authentifiant en tant que principal Kerberos. Par exemple, un utilisateur IdM effectue **kinit *username*** et fournit son mot de passe.

2. Le KDC vérifie la présence du principal dans sa base de données, authentifie le client et évalue les politiques de tickets Kerberos afin de déterminer s'il convient d'accéder à la demande.

3. Le KDC délivre au client un ticket d'attribution de ticket (TGT) avec un cycle de vie et des indicateurs d'authentification conformément à la politique de ticket appropriée.

4. Avec le TGT, le client demande une adresse *service ticket* au KDC pour communiquer avec un service Kerberisé sur un hôte cible.

5. Le KDC vérifie si le TGT du client est toujours valide et évalue la demande de ticket de service par rapport aux politiques de ticket.

6. Le KDC délivre au client une adresse *service ticket*.

7. Avec le ticket de service, le client peut initier une communication cryptée avec le service sur l'hôte cible.

13.2. TYPES DE POLITIQUES DE TICKET IDM KERBEROS

Les politiques de ticket Kerberos de l'IdM mettent en œuvre les types de politique de ticket suivants :

Politique de connexion

Pour protéger les services Kerberisés avec différents niveaux de sécurité, vous pouvez définir des stratégies de connexion pour appliquer des règles basées sur le mécanisme de préauthentification qu'un client a utilisé pour récupérer un ticket d'attribution de ticket (TGT).
Par exemple, vous pouvez exiger une authentification par carte à puce pour vous connecter à **client1.example.com**, et exiger une authentification à deux facteurs pour accéder à l'application **testservice** sur **client2.example.com**.

Pour appliquer les politiques de connexion, associez *authentication indicators* aux services. Seuls les clients dont les demandes de tickets de service contiennent les indicateurs d'authentification requis peuvent accéder à ces services. Pour plus d'informations, voir Indicateurs d'authentification Kerberos.

Politique relative au cycle de vie des billets

Chaque ticket Kerberos a une *lifetime* et une *renewal age* potentielle : vous pouvez renouveler un ticket avant qu'il n'atteigne sa durée de vie maximale, mais pas après qu'il ait dépassé son âge maximal de renouvellement.
La durée de vie globale par défaut des tickets est d'un jour (86400 secondes) et l'âge maximum de renouvellement global par défaut est d'une semaine (604800 secondes). Pour ajuster ces valeurs globales, voir Configuration de la politique globale de cycle de vie des tickets .

Vous pouvez également définir vos propres politiques de cycle de vie des tickets :

- Pour configurer des valeurs de cycle de vie des tickets globaux différentes pour chaque indicateur d'authentification, voir Configuration des politiques de tickets globaux par indicateur d'authentification.
- Pour définir des valeurs de cycle de vie des tickets pour un seul utilisateur qui s'appliquent quelle que soit la méthode d'authentification utilisée, voir Configuration de la stratégie de ticket par défaut pour un utilisateur.
- Pour définir des valeurs de cycle de vie de ticket individuelles pour chaque indicateur d'authentification qui ne s'appliquent qu'à un seul utilisateur, voir Configuration de politiques de ticket d'indicateur d'authentification individuelles pour un utilisateur.

13.3. INDICATEURS D'AUTHENTIFICATION KERBEROS

Le centre de distribution de clés Kerberos (KDC) associe *authentication indicators* à un ticket d'attribution de ticket (TGT) en fonction du mécanisme de préauthentification utilisé par le client pour prouver son identité :

otp
: authentification à deux facteurs (mot de passe à usage unique)

radius
: Authentification RADIUS (généralement pour l'authentification 802.1x)

pkinit
: Authentification par PKINIT, carte à puce ou certificat

hardened
: mots de passe renforcés (SPAKE ou FAST)[1]

Le KDC attache ensuite les indicateurs d'authentification du TGT à toutes les demandes de tickets de service qui en découlent. Le KDC applique des politiques telles que le contrôle d'accès aux services, la durée de vie maximale des tickets et l'âge maximal de renouvellement sur la base des indicateurs d'authentification.

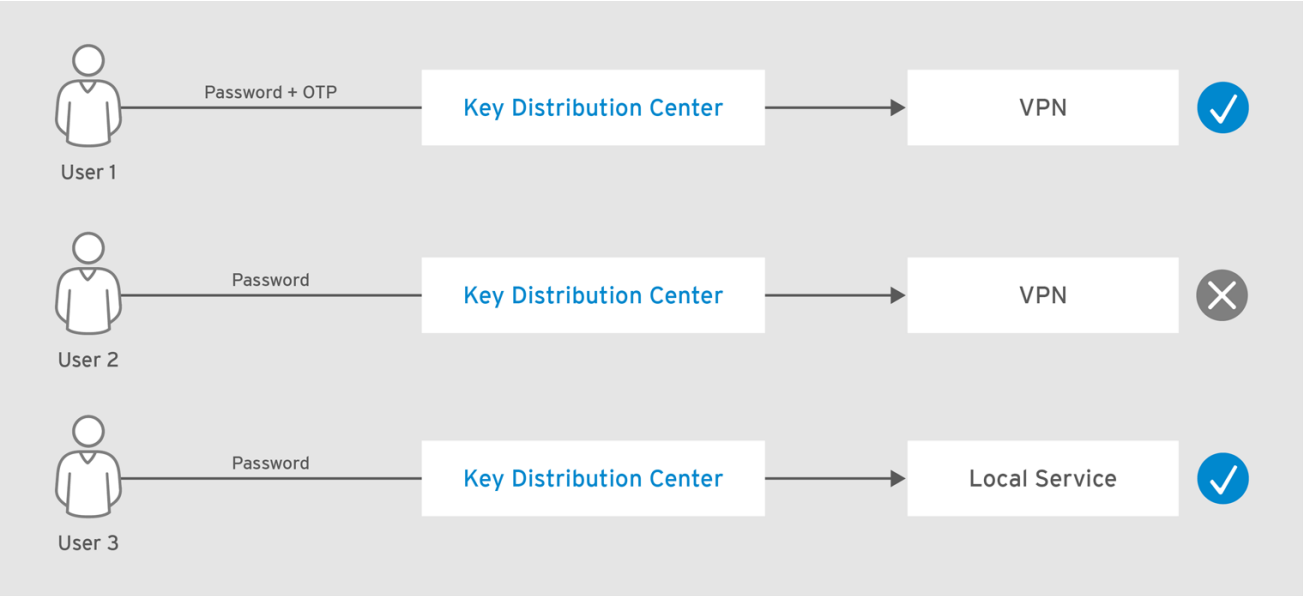
Indicateurs d'authentification et services IdM

Si vous associez un service ou un hôte à un indicateur d'authentification, seuls les clients qui ont utilisé le mécanisme d'authentification correspondant pour obtenir un TGT pourront y accéder. Le KDC, et non l'application ou le service, vérifie la présence d'indicateurs d'authentification dans les demandes de

tickets de service et accorde ou refuse les demandes en fonction des politiques de connexion Kerberos.

Par exemple, pour exiger une authentification à deux facteurs pour se connecter à un réseau privé virtuel (VPN), associez l'indicateur d'authentification **otp** à ce service. Seuls les utilisateurs qui ont utilisé un mot de passe à usage unique pour obtenir leur TGT initial auprès du KDC pourront se connecter au réseau privé virtuel :

Figure 13.1. Exemple de service VPN nécessitant l'indicateur d'authentification otp



RHEL_404973_1016

Si aucun indicateur d'authentification n'est attribué à un service ou à un hôte, celui-ci acceptera les tickets authentifiés par n'importe quel mécanisme.

Ressources supplémentaires

- Renforcement des indicateurs d'authentification pour un service IdM
- Activation de l'authentification GSSAPI et application des indicateurs d'authentification Kerberos pour sudo sur un client IdM

13.4. RENFORCEMENT DES INDICATEURS D'AUTHENTIFICATION POUR UN SERVICE IDM

Les mécanismes d'authentification pris en charge par la gestion des identités (IdM) varient en termes de force d'authentification. Par exemple, l'obtention du ticket initial Kerberos (TGT) à l'aide d'un mot de passe à usage unique (OTP) en combinaison avec un mot de passe standard est considérée comme plus sûre que l'authentification à l'aide d'un mot de passe standard uniquement.

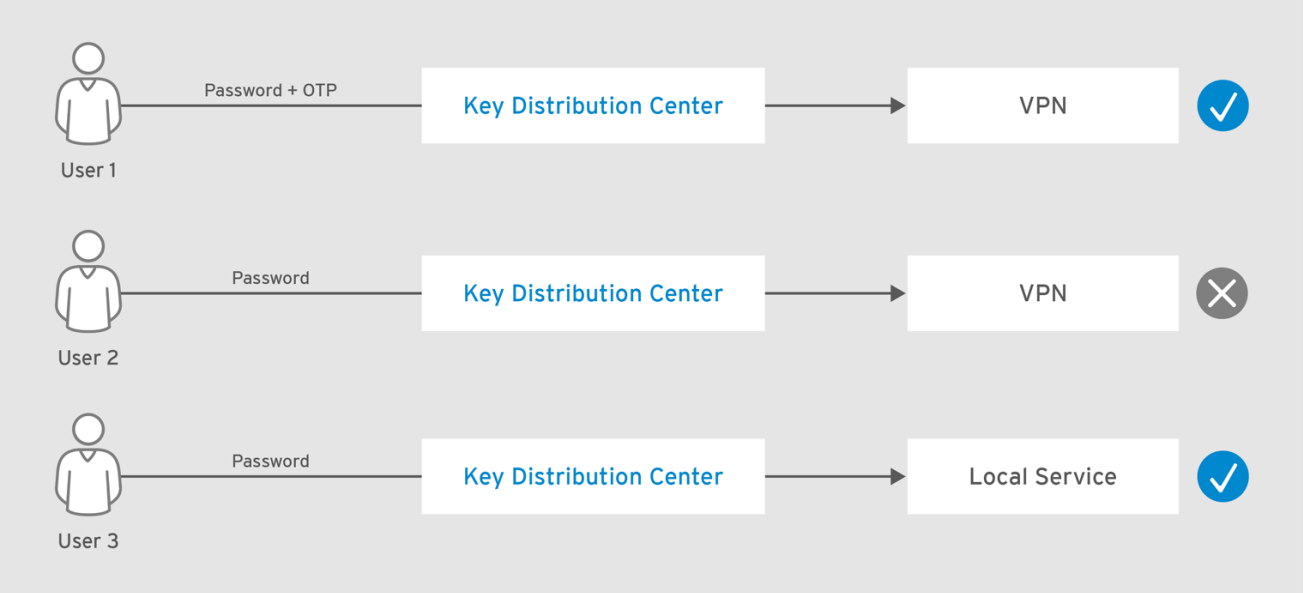
En associant des indicateurs d'authentification à un service IdM particulier, vous pouvez, en tant qu'administrateur IdM, configurer le service de sorte que seuls les utilisateurs qui ont utilisé ces mécanismes de préauthentification spécifiques pour obtenir leur ticket initial (TGT) puissent accéder au service.

De cette façon, vous pouvez configurer différents services IdM de manière à ce que :

- Seuls les utilisateurs qui ont utilisé une méthode d'authentification plus forte pour obtenir leur TGT initial, comme un mot de passe à usage unique (OTP), peuvent accéder à des services essentiels à la sécurité, comme un VPN.

- Les utilisateurs qui ont utilisé des méthodes d'authentification plus simples pour obtenir leur TGT initial, comme un mot de passe, ne peuvent accéder qu'à des services non critiques, comme les connexions locales.

Figure 13.2. Exemple d'authentification à l'aide de différentes technologies



RHEL_404973_1016

Cette procédure décrit la création d'un service IdM et sa configuration pour exiger des indicateurs d'authentification Kerberos particuliers pour les demandes de tickets de service entrantes.

13.4.1. Création d'une entrée de service IdM et de son keytab Kerberos

L'ajout d'une entrée *IdM service* à IdM pour un service fonctionnant sur un hôte IdM crée un principal Kerberos correspondant et permet au service de demander un certificat SSL, un keytab Kerberos ou les deux.

La procédure suivante décrit la création d'une entrée de service IdM et la génération d'un keytab Kerberos associé pour chiffrer la communication avec ce service.

Conditions préalables

- Votre service peut stocker un principal Kerberos, un certificat SSL ou les deux.

Procédure

1. Ajoutez un service IdM avec la commande **ipa service-add** pour créer un principal Kerberos associé à ce service. Par exemple, pour créer l'entrée du service IdM pour l'application **testservice** qui s'exécute sur l'hôte **client.example.com**:

```
[root@client ~]# ipa service-add testservice/client.example.com
---------------------------------------------------------------
Modified service "testservice/client.example.com@EXAMPLE.COM"
---------------------------------------------------------------
  Principal name: testservice/client.example.com@EXAMPLE.COM
  Principal alias: testservice/client.example.com@EXAMPLE.COM
  Managed by: client.example.com
```

2. Générer et stocker un keytab Kerberos pour le service sur le client.

```
[root@client ~]# ipa-getkeytab -k /etc/testservice.keytab -p
testservice/client.example.com
Keytab successfully retrieved and stored in: /etc/testservice.keytab
```

Verification steps

1. Affichez des informations sur un service IdM à l'aide de la commande **ipa service-show**.

```
[root@server ~]# ipa service-show testservice/client.example.com
  Principal name: testservice/client.example.com@EXAMPLE.COM
  Principal alias: testservice/client.example.com@EXAMPLE.COM
  Keytab: True
  Managed by: client.example.com
```

2. Affichez le contenu de la base de données Kerberos du service avec la commande **klist**.

```
[root@server etc]# klist -ekt /etc/testservice.keytab
Keytab name: FILE:/etc/testservice.keytab
KVNO Timestamp           Principal
---- ------------------- ------------------------------------------------------
   2 04/01/2020 17:52:55 testservice/client.example.com@EXAMPLE.COM (aes256-cts-
hmac-sha1-96)
   2 04/01/2020 17:52:55 testservice/client.example.com@EXAMPLE.COM (aes128-cts-
hmac-sha1-96)
   2 04/01/2020 17:52:55 testservice/client.example.com@EXAMPLE.COM (camellia128-cts-
cmac)
   2 04/01/2020 17:52:55 testservice/client.example.com@EXAMPLE.COM (camellia256-cts-
cmac)
```

13.4.2. Associer des indicateurs d'authentification à un service IdM à l'aide de la CLI IdM

En tant qu'administrateur Identity Management (IdM), vous pouvez configurer un hôte ou un service pour exiger qu'un ticket de service présenté par l'application cliente contienne un indicateur d'authentification spécifique. Par exemple, vous pouvez vous assurer que seuls les utilisateurs qui ont utilisé un jeton d'authentification à deux facteurs IdM valide avec leur mot de passe lors de l'obtention d'un ticket Kerberos (TGT) pourront accéder à cet hôte ou à ce service.

Cette procédure décrit la configuration d'un service pour qu'il exige des indicateurs d'authentification Kerberos particuliers dans les demandes de tickets de service entrantes.

Conditions préalables

- Vous avez créé une entrée de service IdM pour un service qui s'exécute sur un hôte IdM. Voir Création d'une entrée de service IdM et de son keytab Kerberos .
- Vous avez obtenu le ticket d'attribution de ticket d'un utilisateur administratif dans IdM.

AVERTISSEMENT

Ne **not** assigner des indicateurs d'authentification aux services IdM internes. Les services IdM suivants ne peuvent pas effectuer les étapes d'authentification interactive requises par PKINIT et les méthodes d'authentification multifactorielle :

> **host**/server.example.com@EXAMPLE.COM
> **HTTP**/server.example.com@EXAMPLE.COM
> **ldap**/server.example.com@EXAMPLE.COM
> **DNS**/server.example.com@EXAMPLE.COM
> **cifs**/server.example.com@EXAMPLE.COM

Procédure

- Utilisez la commande **ipa service-mod** pour spécifier un ou plusieurs indicateurs d'authentification requis pour un service, identifié par l'argument **--auth-ind**.

Méthode d'authentification	**--auth-ind** valeur
Authentification à deux facteurs	**otp**
Authentification RADIUS	**radius**
Authentification par PKINIT, carte à puce ou certificat	**pkinit**
Mots de passe renforcés (SPAKE ou FAST)	**hardened**

Par exemple, pour exiger qu'un utilisateur ait été authentifié par carte à puce ou par OTP pour récupérer un ticket de service pour le principal **testservice** sur l'hôte **client.example.com**:

```
[root@server ~]# ipa service-mod testservice/client.example.com@EXAMPLE.COM --auth-ind otp --auth-ind pkinit
---------------------------------------------------------------
Modified service "testservice/client.example.com@EXAMPLE.COM"
---------------------------------------------------------------
  Principal name: testservice/client.example.com@EXAMPLE.COM
  Principal alias: testservice/client.example.com@EXAMPLE.COM
  Authentication Indicators: otp, pkinit
  Managed by: client.example.com
```

NOTE

Pour supprimer tous les indicateurs d'authentification d'un service, il faut fournir une liste vide d'indicateurs :

```
[root@server ~]# ipa service-mod
testservice/client.example.com@EXAMPLE.COM --auth-ind ''
-----------------------------------------------------
Modified service "testservice/client.example.com@EXAMPLE.COM"
-----------------------------------------------------
  Principal name: testservice/client.example.com@EXAMPLE.COM
  Principal alias: testservice/client.example.com@EXAMPLE.COM
  Managed by: client.example.com
```

Verification steps

- La commande **ipa service-show** permet d'afficher des informations sur un service IdM, y compris les indicateurs d'authentification qu'il requiert.

```
[root@server ~]# ipa service-show testservice/client.example.com
  Principal name: testservice/client.example.com@EXAMPLE.COM
  Principal alias: testservice/client.example.com@EXAMPLE.COM
  Authentication Indicators: otp, pkinit
  Keytab: True
  Managed by: client.example.com
```

Ressources supplémentaires

- Récupération d'un ticket de service Kerberos pour un service IdM
- Activation de l'authentification GSSAPI et application des indicateurs d'authentification Kerberos pour sudo sur un client IdM

13.4.3. Associer des indicateurs d'authentification à un service IdM à l'aide de l'interface Web IdM

En tant qu'administrateur Identity Management (IdM), vous pouvez configurer un hôte ou un service pour qu'un ticket de service présenté par l'application cliente contienne un indicateur d'authentification spécifique. Par exemple, vous pouvez vous assurer que seuls les utilisateurs qui ont utilisé un jeton d'authentification à deux facteurs IdM valide avec leur mot de passe lors de l'obtention d'un ticket Kerberos (TGT) pourront accéder à cet hôte ou à ce service.

Cette procédure décrit comment utiliser l'interface Web IdM pour configurer un hôte ou un service afin qu'il exige des indicateurs d'authentification Kerberos particuliers pour les demandes de tickets entrantes.

Conditions préalables

- Vous vous êtes connecté à l'interface Web IdM en tant qu'utilisateur administratif.

Procédure

1. Sélectionner **Identité → Hôtes** ou **D'identité → Services**.

2. Cliquez sur le nom de l'hôte ou du service requis.

3. Sous **Authentication indicators**, sélectionnez la méthode d'authentification requise.

   - Par exemple, la sélection de **OTP** garantit que seuls les utilisateurs qui ont utilisé un jeton d'authentification à deux facteurs IdM valide avec leur mot de passe lors de l'obtention d'un TGT Kerberos pourront accéder à l'hôte ou au service.

   - Si vous sélectionnez à la fois **OTP** et **RADIUS**, les utilisateurs qui ont utilisé un jeton d'authentification à deux facteurs IdM valide avec leur mot de passe lors de l'obtention d'un TGT Kerberos **and** qui ont utilisé le serveur RADIUS pour obtenir leur TGT Kerberos seront autorisés à accéder à l'application.

4. Cliquez sur **Enregistrer** en haut de la page.

Ressources supplémentaires

- Récupération d'un ticket de service Kerberos pour un service IdM

- Activation de l'authentification GSSAPI et application des indicateurs d'authentification Kerberos pour sudo sur un client IdM

13.4.4. Récupération d'un ticket de service Kerberos pour un service IdM

La procédure suivante décrit la récupération d'un ticket de service Kerberos pour un service IdM. Vous pouvez utiliser cette procédure pour tester les politiques relatives aux tickets Kerberos, par exemple en imposant la présence de certains indicateurs d'authentification Kerberos dans un ticket d'attribution de ticket (TGT).

Conditions préalables

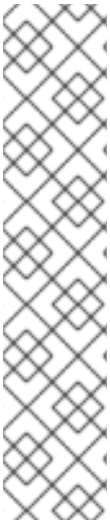
- Si le service avec lequel vous travaillez n'est pas un service IdM interne, vous avez créé une entrée *IdM service* correspondante. Voir Création d'une entrée de service IdM et de son keytab Kerberos.

- Vous disposez d'un ticket Kerberos (TGT).

Procédure

- Utilisez la commande **kvno** avec l'option **-S** pour récupérer un ticket de service et spécifier le nom du service IdM et le nom de domaine complet de l'hôte qui le gère.

```
[root@server ~]# kvno -S testservice client.example.com
testservice/client.example.com@EXAMPLE.COM: kvno = 1
```



NOTE

Si vous devez accéder à un service IdM et que votre ticket d'attribution de ticket (TGT) actuel ne possède pas les indicateurs d'authentification Kerberos requis qui lui sont associés, effacez votre cache d'informations d'identification Kerberos actuel à l'aide de la commande **kdestroy** et récupérez un nouveau TGT :

```
[root@server ~]# kdestroy
```

Par exemple, si vous avez initialement récupéré un TGT en vous authentifiant avec un mot de passe et que vous devez accéder à un service IdM auquel est associé l'indicateur d'authentification **pkinit**, détruisez votre cache d'informations d'identification actuel et authentifiez-vous à nouveau avec une carte à puce. Voir les indicateurs d'authentification Kerberos.

Verification steps

- Utilisez la commande **klist** pour vérifier que le ticket de service se trouve dans le cache d'informations d'identification Kerberos par défaut.

```
[root@server etc]# klist_
Ticket cache: KCM:1000
Default principal: admin@EXAMPLE.COM

Valid starting       Expires              Service principal
04/01/2020 12:52:42  04/02/2020 12:52:39  krbtgt/EXAMPLE.COM@EXAMPLE.COM
04/01/2020 12:54:07 04/02/2020 12:52:39
testservice/client.example.com@EXAMPLE.COM
```

13.4.5. Ressources supplémentaires

- Voir indicateurs d'authentification Kerberos.

13.5. CONFIGURATION DE LA POLITIQUE GLOBALE DE CYCLE DE VIE DES TICKETS

La politique de ticket globale s'applique à tous les tickets de service et aux utilisateurs pour lesquels aucune politique de ticket par utilisateur n'a été définie.

La procédure suivante décrit l'ajustement de la durée de vie maximale des tickets et de l'âge maximal de renouvellement des tickets pour la stratégie globale de tickets Kerberos à l'aide de la commande **ipa krbtpolicy-mod**.

Lors de l'utilisation de la commande **ipa krbtpolicy-mod**, spécifiez au moins l'un des arguments suivants :

- **--maxlife** pour la durée de vie maximale du ticket en secondes
- **--maxrenew** pour l'âge maximum renouvelable en secondes

Procédure

1. Pour modifier la politique globale en matière de tickets :

```
[root@server ~]# ipa krbtpolicy-mod --maxlife=$((8*60*60)) --maxrenew=$((24*60*60))
  Max life: 28800
  Max renew: 86400
```

In this example, the maximum lifetime is set to eight hours (8 * 60 minutes * 60 seconds) and the maximum renewal age is set to one day (24 * 60 minutes * 60 seconds).

2. Facultatif : Pour rétablir les valeurs d'installation par défaut de la politique globale en matière de tickets Kerberos :

```
[root@server ~]# ipa krbtpolicy-reset
  Max life: 86400
  Max renew: 604800
```

Verification steps

- Afficher la politique globale en matière de tickets :

```
[root@server ~]# ipa krbtpolicy-show
  Max life: 28800
  Max renew: 86640
```

Ressources supplémentaires

- Voir Configuration de la politique de billetterie par défaut pour un utilisateur .
- Voir Configuration des politiques de ticket des indicateurs d'authentification individuels pour un utilisateur.

13.6. CONFIGURATION DES POLITIQUES DE TICKETS GLOBALES PAR INDICATEUR D'AUTHENTIFICATION

Cette procédure décrit le réglage de la durée de vie maximale globale du ticket et de l'âge maximal renouvelable pour chaque indicateur d'authentification. Ces paramètres s'appliquent aux utilisateurs pour lesquels aucune politique de ticket par utilisateur n'a été définie.

Utilisez la commande **ipa krbtpolicy-mod** pour spécifier la durée de vie maximale globale ou l'âge maximal renouvelable des tickets Kerberos en fonction des indicateurs d'authentification qui leur sont associés.

Procédure

- Par exemple, pour définir les valeurs globales de durée de vie et d'âge de renouvellement du ticket à deux facteurs à une semaine, et les valeurs globales de durée de vie et d'âge de renouvellement du ticket de carte à puce à deux semaines :

```
[root@server ~]# ipa krbtpolicy-mod --otp-maxlife=604800 --otp-maxrenew=604800 --
pkinit-maxlife=172800 --pkinit-maxrenew=172800
```

Verification steps

- Afficher la politique globale en matière de tickets :

```
[root@server ~]# ipa krbtpolicy-show
  Max life: 86400
  OTP max life: 604800
  PKINIT max life: 172800
  Max renew: 604800
  OTP max renew: 604800
  PKINIT max renew: 172800
```

Notez que les valeurs OTP et PKINIT sont différentes des valeurs globales par défaut **Max life** et **Max renew**.

Ressources supplémentaires

- Voir les options de l'indicateur d'authentification pour la commande **krbtpolicy-mod** .
- Voir Configuration de la politique de billetterie par défaut pour un utilisateur .
- Voir Configuration des politiques de ticket des indicateurs d'authentification individuels pour un utilisateur.

13.7. CONFIGURATION DE LA POLITIQUE DE BILLETTERIE PAR DÉFAUT POUR UN UTILISATEUR

Vous pouvez définir une politique de ticket Kerberos unique qui ne s'applique qu'à un seul utilisateur. Ces paramètres par utilisateur remplacent la politique de ticket globale pour tous les indicateurs d'authentification.

Utilisez la commande **ipa krbtpolicy-mod *username*** et spécifiez au moins l'un des arguments suivants :

- **--maxlife** pour la durée de vie maximale du ticket en secondes
- **--maxrenew** pour l'âge maximum renouvelable en secondes

Procédure

1. Par exemple, pour définir la durée de vie maximale du ticket de l'utilisateur IdM **admin** à deux jours et l'âge maximal de renouvellement à deux semaines :

```
[root@server ~]# ipa krbtpolicy-mod admin --maxlife=172800 --maxrenew=1209600
  Max life: 172800
  Max renew: 1209600
```

2. Optionnel : Pour réinitialiser la politique de ticket d'un utilisateur :

```
[root@server ~]# ipa krbtpolicy-reset admin
```

Verification steps

- Afficher la politique de ticket Kerberos effective qui s'applique à un utilisateur :

```
[root@server ~]# ipa krbtpolicy-show admin
  Max life: 172800
  Max renew: 1209600
```

Ressources supplémentaires

- Voir Configuration de la politique globale de cycle de vie des tickets .
- Voir Configuration des politiques globales de tickets par indicateur d'authentification .

13.8. CONFIGURER DES POLITIQUES DE TICKETS D'INDICATEURS D'AUTHENTIFICATION INDIVIDUELS POUR UN UTILISATEUR

En tant qu'administrateur, vous pouvez définir des politiques de tickets Kerberos pour un utilisateur qui diffèrent selon l'indicateur d'authentification. Par exemple, vous pouvez configurer une politique permettant à l'utilisateur IdM **admin** de renouveler un ticket pendant deux jours s'il a été obtenu par authentification OTP, et pendant une semaine s'il a été obtenu par authentification par carte à puce.

Ces paramètres par indicateur d'authentification remplaceront la politique de ticket par défaut *user's*, la politique de ticket par défaut *global* et toute politique de ticket d'indicateur d'authentification *global*.

Utilisez la commande **ipa krbtpolicy-mod *username*** pour définir des valeurs personnalisées de durée de vie maximale et d'âge maximal renouvelable pour les tickets Kerberos d'un utilisateur en fonction des indicateurs d'authentification qui leur sont associés.

Procédure

1. Par exemple, pour permettre à l'utilisateur d'IdM **admin** de renouveler un ticket Kerberos pendant deux jours s'il a été obtenu avec l'authentification par mot de passe à usage unique, définissez l'option **--otp-maxrenew**:

```
[root@server ~]# ipa krbtpolicy-mod admin --otp-maxrenew=$((2*24*60*60))
  OTP max renew: 172800
```

2. Optionnel : Pour réinitialiser la politique de ticket d'un utilisateur :

```
[root@server ~]# ipa krbtpolicy-reset username
```

Verification steps

- Afficher la politique de ticket Kerberos effective qui s'applique à un utilisateur :

```
[root@server ~]# ipa krbtpolicy-show admin
  Max life: 28800
  Max renew: 86640
```

Ressources supplémentaires

- Voir les options de l'indicateur d'authentification pour la commande **krbtpolicy-mod** .
- Voir Configuration de la politique de billetterie par défaut pour un utilisateur .
- Voir Configuration de la politique globale de cycle de vie des tickets .
- Voir Configuration des politiques globales de tickets par indicateur d'authentification .

13.9. OPTIONS DE L'INDICATEUR D'AUTHENTIFICATION POUR LA COMMANDE KRBTPOLICY-MOD

Spécifiez les valeurs des indicateurs d'authentification avec les arguments suivants.

Tableau 13.1. Options de l'indicateur d'authentification pour la commande krbtpolicy-mod

Indicateur d'authentification	Argument en faveur d'une durée de vie maximale	Argument en faveur d'un âge maximal de renouvellement
otp	**--otp-maxlife**	**--otp-maxrenew**
radius	**--radius-maxlife**	**--radius-maxrenew**
pkinit	**--pkinit-maxlife**	**--pkinit-maxrenew**
hardened	**--hardened-maxlife**	**--hardened-maxrenew**

[1] Un mot de passe renforcé est protégé contre les attaques par force brute du dictionnaire de mots de passe en utilisant la pré-authentification SPAKE (Single-Party Public-Key Authenticated Key Exchange) et/ou l'armure FAST (Flexible Authentication via Secure Tunneling).

CHAPITRE 14. GESTION DES FICHIERS KEYTAB KERBEROS DE L'IDM

La documentation suivante explique ce que sont les fichiers keytab Kerberos et comment la gestion des identités (IdM) les utilise pour permettre aux services de s'authentifier de manière sécurisée avec Kerberos.

Vous pouvez utiliser ces informations pour comprendre pourquoi vous devez protéger ces fichiers sensibles et pour résoudre les problèmes de communication entre les services IdM.

Cette section aborde les sujets suivants :

- Comment la gestion des identités utilise les fichiers keytab de Kerberos
- Vérification de la synchronisation des fichiers keytab Kerberos avec la base de données IdM
- Liste des fichiers keytab Kerberos de l'IdM et de leur contenu

Cette section explique également comment afficher le type de cryptage de votre clé principale IdM .

14.1. COMMENT LA GESTION DES IDENTITÉS UTILISE LES FICHIERS KEYTAB DE KERBEROS

Un keytab Kerberos est un fichier contenant les principaux Kerberos et les clés de chiffrement correspondantes. Les hôtes, les services, les utilisateurs et les scripts peuvent utiliser les keytabs pour s'authentifier auprès du centre de distribution de clés Kerberos (KDC) en toute sécurité, sans nécessiter d'interaction humaine.

Chaque service IdM sur un serveur IdM a un principe Kerberos unique stocké dans la base de données Kerberos. Par exemple, si les serveurs IdM **east.idm.example.com** et **west.idm.example.com** fournissent des services DNS, l'IdM crée 2 principes DNS Kerberos uniques pour identifier ces services, qui suivent la convention de dénomination **<service>/host.domain.com@REALM.COM**:

- **DNS/east.idm.example.com@IDM.EXAMPLE.COM**
- **DNS/west.idm.example.com@IDM.EXAMPLE.COM**

IdM crée un fichier keytab sur le serveur pour chacun de ces services afin de stocker une copie locale des clés Kerberos, ainsi que leur numéro de version de clé (KVNO). Par exemple, le fichier keytab par défaut **/etc/krb5.keytab** stocke le principal **host**, qui représente cette machine dans le domaine Kerberos et est utilisé pour l'authentification de connexion. Le KDC génère des clés de chiffrement pour les différents algorithmes de chiffrement qu'il prend en charge, tels que **aes256-cts-hmac-sha1-96** et **aes128-cts-hmac-sha1-96**.

Vous pouvez afficher le contenu d'un fichier keytab à l'aide de la commande **klist**:

```
[root@idmserver ~]# klist -ekt /etc/krb5.keytab
Keytab name: FILE:/etc/krb5.keytab
KVNO Timestamp           Principal
---- ------------------- ------------------------------------------------------
   2 02/24/2022 20:28:09 host/idmserver.idm.example.com@IDM.EXAMPLE.COM (aes256-cts-hmac-sha1-96)
   2 02/24/2022 20:28:09 host/idmserver.idm.example.com@IDM.EXAMPLE.COM (aes128-cts-hmac-sha1-96)
   2 02/24/2022 20:28:09 host/idmserver.idm.example.com@IDM.EXAMPLE.COM (camellia128-cts-
```

```
cmac)
   2 02/24/2022 20:28:09 host/idmserver.idm.example.com@IDM.EXAMPLE.COM (camellia256-cts-
cmac)
```

Ressources supplémentaires

- Vérification de la synchronisation des fichiers keytab Kerberos avec la base de données IdM
- Liste des fichiers keytab Kerberos de l'IdM et de leur contenu

14.2. VÉRIFICATION DE LA SYNCHRONISATION DES FICHIERS KEYTAB KERBEROS AVEC LA BASE DE DONNÉES IDM

Lorsque vous modifiez un mot de passe Kerberos, IdM génère automatiquement une nouvelle clé Kerberos correspondante et incrémente son numéro de version de clé (KVNO). Si un keytab Kerberos n'est pas mis à jour avec la nouvelle clé et le KVNO, les services qui dépendent de ce keytab pour récupérer une clé valide risquent de ne pas pouvoir s'authentifier auprès du centre de distribution de clés Kerberos (KDC).

Si l'un de vos services IdM ne peut pas communiquer avec un autre service, utilisez la procédure suivante pour vérifier que vos fichiers keytab Kerberos sont synchronisés avec les clés stockées dans la base de données IdM. S'ils ne sont pas synchronisés, récupérez un fichier keytab Kerberos avec une clé et un KVNO mis à jour. Cet exemple compare et récupère un principal **DNS** mis à jour pour un serveur IdM.

Conditions préalables

- Vous devez vous authentifier en tant que compte administrateur IdM pour récupérer les fichiers keytab
- Vous devez vous authentifier en tant que compte **root** pour modifier les fichiers keytab appartenant à d'autres utilisateurs

Procédure

1. Affichez le KVNO des mandants dans le keytab que vous vérifiez. Dans l'exemple suivant, le fichier **/etc/named.keytab** contient la clé du principal **DNS/server1.idm.example.com@EXAMPLE.COM** avec un KVNO de 2.

```
[root@server1 ~]# klist -ekt /etc/named.keytab
Keytab name: FILE:/etc/named.keytab
KVNO Timestamp           Principal
---- ------------------- ------------------------------------------------------
   2 11/26/2021 13:51:11 DNS/server1.idm.example.com@EXAMPLE.COM (aes256-cts-
hmac-sha1-96)
   2 11/26/2021 13:51:11 DNS/server1.idm.example.com@EXAMPLE.COM (aes128-cts-
hmac-sha1-96)
   2 11/26/2021 13:51:11 DNS/server1.idm.example.com@EXAMPLE.COM (camellia128-cts-
cmac)
   2 11/26/2021 13:51:11 DNS/server1.idm.example.com@EXAMPLE.COM (camellia256-cts-
cmac)
```

2. Afficher le KVNO du principal stocké dans la base de données IdM. Dans cet exemple, le KVNO de la clé dans la base de données IdM ne correspond pas au KVNO dans la base de données.

```
[root@server1 ~]# kvno DNS/server1.idm.example.com@EXAMPLE.COM
DNS/server1.idm.example.com@EXAMPLE.COM: kvno = 3
```

3. S'authentifier en tant que compte administrateur IdM.

```
[root@server1 ~]# kinit admin
Password for admin@IDM.EXAMPLE.COM:
```

4. Récupérez une clé Kerberos mise à jour pour le principal et stockez-la dans son keytab. Effectuez cette étape en tant qu'utilisateur **root** afin de pouvoir modifier le fichier **/etc/named.keytab**, qui appartient à l'utilisateur **named**.

```
[root@server1 ~]# ipa-getkeytab -s server1.idm.example.com -p
DNS/server1.idm.example.com -k /etc/named.keytab
```

Vérification

1. Affiche le KVNO mis à jour du principal dans la base de données des clés.

```
[root@server1 ~]# klist -ekt /etc/named.keytab
Keytab name: FILE:/etc/named.keytab
KVNO Timestamp           Principal
---- ------------------- ------------------------------------------------------
   4 08/17/2022 14:42:11 DNS/server1.idm.example.com@EXAMPLE.COM (aes256-cts-
hmac-sha1-96)
   4 08/17/2022 14:42:11 DNS/server1.idm.example.com@EXAMPLE.COM (aes128-cts-
hmac-sha1-96)
   4 08/17/2022 14:42:11 DNS/server1.idm.example.com@EXAMPLE.COM (camellia128-cts-
cmac)
   4 08/17/2022 14:42:11 DNS/server1.idm.example.com@EXAMPLE.COM (camellia256-cts-
cmac)
```

2. Afficher le KVNO du principal stocké dans la base de données IdM et s'assurer qu'il correspond au KVNO de la base de données.

```
[root@server1 ~]# kvno DNS/server1.idm.example.com@EXAMPLE.COM
DNS/server1.idm.example.com@EXAMPLE.COM: kvno = 4
```

Ressources supplémentaires

- Comment la gestion des identités utilise les fichiers keytab de Kerberos
- Liste des fichiers keytab Kerberos de l'IdM et de leur contenu

14.3. LISTE DES FICHIERS KEYTAB KERBEROS DE L'IDM ET DE LEUR CONTENU

Le tableau suivant indique l'emplacement, le contenu et l'objectif des fichiers keytab IdM Kerberos.

Tableau 14.1. Tableau

Emplacement des touches	Contenu	Objectif
/etc/krb5.keytab	**host** principal	Vérification des informations d'identification de l'utilisateur lors de la connexion, utilisée par NFS s'il n'y a pas de principal **nfs**
/etc/dirsrv/ds.keytab	**ldap** principal	Authentification des utilisateurs de la base de données IdM, réplication sécurisée du contenu de la base de données entre les répliques IdM
/var/lib/ipa/gssproxy/http.keytab	**HTTP** principal	Authentification au serveur Apache
/etc/named.keytab	**DNS** principal	Mise à jour sécurisée des enregistrements DNS
/etc/ipa/dnssec/ipa-dnskeysyncd.keytab	**ipa-dnskeysyncd** principal	Synchronisation d'OpenDNSSEC avec LDAP
/etc/pki/pki-tomcat/dogtag.keytab	**dogtag** principal	Communiquer avec l'autorité de certification (AC)
/etc/samba/samba.keytab	**cifs** et **host** directeurs d'école	Communiquer avec le service Samba
/var/lib/sss/keytabs/*ad-domain.com*.keytab	Contrôleurs de domaine (DC) Active Directory (AD) sous la forme suivante ***HOSTNAME$@AD-DOMAIN.COM***	Communication avec les AD DC par l'intermédiaire d'une confiance IdM-AD

Ressources supplémentaires

- Comment la gestion des identités utilise les fichiers keytab de Kerberos
- Vérification de la synchronisation des fichiers keytab Kerberos avec la base de données IdM

14.4. VISUALISATION DU TYPE DE CRYPTAGE DE VOTRE CLÉ MAÎTRESSE IDM

En tant qu'administrateur Identity Management (IdM), vous pouvez afficher le type de cryptage de votre clé principale IdM, qui est la clé que le Centre de distribution Kerberos (KDC) IdM utilise pour crypter tous les autres principaux lorsqu'ils sont stockés au repos. La connaissance du type de cryptage vous aide à déterminer la compatibilité de votre déploiement avec les normes FIPS.

À partir de RHEL 8.7, le type de chiffrement est **aes256-cts-hmac-sha384-192**. Ce type de chiffrement est compatible avec la politique cryptographique FIPS par défaut de RHEL 9 visant à se conformer à la norme FIPS 140-3.

Les types de chiffrement utilisés sur les versions précédentes de RHEL ne sont pas compatibles avec les systèmes RHEL 9 qui adhèrent aux normes FIPS 140-3. Pour rendre les systèmes RHEL 9 en mode FIPS compatibles avec un déploiement RHEL 8 FIPS 140-2, activez la politique cryptographique **FIPS:AD-SUPPORT** sur les systèmes RHEL 9.

NOTE

L'implémentation Active Directory de Microsoft ne prend pas encore en charge les types de chiffrement Kerberos RFC8009 qui utilisent SHA-2 HMAC. Si une confiance IdM-AD est configurée, l'utilisation de la sous-politique cryptographique FIPS:AD-SUPPORT est donc requise même si le type de chiffrement de votre clé principale IdM est **aes256-cts-hmac-sha384-192**.

Conditions préalables

- Vous avez accès à **root** à n'importe laquelle des répliques RHEL 8 dans le déploiement IdM.

Procédure

- Sur le réplica, affichez le type de chiffrement sur l'interface de ligne de commande :

```
# kadmin.local getprinc K/M | grep -E '^Key:'
Key: vno 1, aes256-cts-hmac-sha1-96
```

La clé **aes256-cts-hmac-sha1-96** dans le résultat indique que le déploiement IdM a été installé sur un serveur fonctionnant sous RHEL 8.6 ou une version antérieure. La présence d'une clé **aes256-cts-hmac-sha384-192** dans le résultat indique que le déploiement de l'IdM a été installé sur un serveur exécutant RHEL 8.7 ou une version ultérieure.

CHAPITRE 15. UTILISATION DU PROXY KDC DANS IDM

Certains administrateurs peuvent choisir de rendre les ports Kerberos par défaut inaccessibles dans leur déploiement. Pour permettre aux utilisateurs, aux hôtes et aux services d'obtenir des informations d'identification Kerberos, vous pouvez utiliser le service **HTTPS** comme un proxy qui communique avec Kerberos via le port 443 de **HTTPS**.

Dans le cadre de la gestion de l'identité (IdM), le site **Kerberos Key Distribution Center Proxy** (KKDCP) offre cette fonctionnalité.

Sur un serveur IdM, KKDCP est activé par défaut et disponible à l'adresse suivante **https://*server.idm.example.com*/KdcProxy**. Sur un client IdM, vous devez modifier sa configuration Kerberos pour accéder au KKDCP.

15.1. CONFIGURATION D'UN CLIENT IDM POUR L'UTILISATION DE KKDCP

En tant qu'administrateur du système de gestion des identités (IdM), vous pouvez configurer un client IdM pour qu'il utilise le Kerberos Key Distribution Center Proxy (KKDCP) sur un serveur IdM. Ceci est utile si les ports Kerberos par défaut ne sont pas accessibles sur le serveur IdM et que le port 443 de **HTTPS** est le seul moyen d'accéder au service Kerberos.

Conditions préalables

- Vous avez un accès **root** au client IdM.

Procédure

1. Ouvrez le fichier **/etc/krb5.conf** pour le modifier.
2. Dans la section **[realms]**, entrez l'URL du KKDCP pour les options **kdc**, **admin_server** et **kpasswd_server**:

```
[realms]
EXAMPLE.COM = {
  kdc = https://kdc.example.com/KdcProxy
  admin_server = https://kdc.example.com/KdcProxy
  kpasswd_server = https://kdc.example.com/KdcProxy
  default_domain = example.com
}
```

   Pour des raisons de redondance, vous pouvez ajouter les paramètres **kdc**, **admin_server**, et **kpasswd_server** plusieurs fois pour indiquer différents serveurs KKDCP.

3. Redémarrez le service **sssd** pour que les modifications soient prises en compte :

```
~]# systemctl restart sssd
```

15.2. VÉRIFICATION DE L'ACTIVATION DE KKDCP SUR UN SERVEUR IDM

Sur un serveur Identity Management (IdM), le Kerberos Key Distribution Center Proxy (KKDCP) est automatiquement activé à chaque démarrage du serveur web Apache si la paire d'attributs et de valeurs

ipaConfigString=kdcProxyEnabled existe dans le répertoire. Dans ce cas, le lien symbolique **/etc/httpd/conf.d/ipa-kdc-proxy.conf** est créé.

Vous pouvez vérifier si le KKDCP est activé sur le serveur IdM, même en tant qu'utilisateur non privilégié.

Procédure

- Vérifiez que le lien symbolique existe :

```
$ ls -l /etc/httpd/conf.d/ipa-kdc-proxy.conf
lrwxrwxrwx. 1 root root 36 Jun 21  2020 /etc/httpd/conf.d/ipa-kdc-proxy.conf -> /etc/ipa/kdcproxy/ipa-kdc-proxy.conf
```

La sortie confirme que le KKDCP est activé.

15.3. DÉSACTIVATION DE KKDCP SUR UN SERVEUR IDM

En tant qu'administrateur du système de gestion des identités (IdM), vous pouvez désactiver le Kerberos Key Distribution Center Proxy (KKDCP) sur un serveur IdM.

Conditions préalables

- Vous avez un accès **root** au serveur IdM.

Procédure

1. Supprimer la paire d'attributs et de valeurs **ipaConfigString=kdcProxyEnabled** du répertoire :

```
# ipa-ldap-updater /usr/share/ipa/kdcproxy-disable.uldif
Update complete
The ipa-ldap-updater command was successful
```

2. Redémarrez le service **httpd**:

```
# systemctl restart httpd.service
```

KKDCP est maintenant désactivé sur le serveur IdM actuel.

Verification steps

- Vérifiez que le lien symbolique n'existe pas :

```
$ ls -l /etc/httpd/conf.d/ipa-kdc-proxy.conf
ls: cannot access '/etc/httpd/conf.d/ipa-kdc-proxy.conf': No such file or directory
```

15.4. RÉACTIVATION DE KKDCP SUR UN SERVEUR IDM

Sur un serveur IdM, le Kerberos Key Distribution Center Proxy (KKDCP) est activé par défaut et disponible à l'adresse suivante **https://*server.idm.example.com*/KdcProxy**.

Si KKDCP a été désactivé sur un serveur, vous pouvez le réactiver.

Conditions préalables

- Vous avez un accès **root** au serveur IdM.

Procédure

1. Ajoutez la paire d'attributs et de valeurs **ipaConfigString=kdcProxyEnabled** au répertoire :

```
# ipa-ldap-updater /usr/share/ipa/kdcproxy-enable.uldif
Update complete
The ipa-ldap-updater command was successful
```

2. Redémarrez le service **httpd**:

```
# systemctl restart httpd.service
```

KKDCP est maintenant activé sur le serveur IdM actuel.

Verification steps

- Vérifiez que le lien symbolique existe :

```
$ ls -l /etc/httpd/conf.d/ipa-kdc-proxy.conf
lrwxrwxrwx. 1 root root 36 Jun 21  2020 /etc/httpd/conf.d/ipa-kdc-proxy.conf ->
/etc/ipa/kdcproxy/ipa-kdc-proxy.conf
```

15.5. CONFIGURATION DU SERVEUR KKDCP I

La configuration suivante permet d'utiliser TCP comme protocole de transport entre le KKDCP de l'IdM et le domaine Active Directory (AD), lorsque plusieurs serveurs Kerberos sont utilisés.

Conditions préalables

- Vous avez accès à **root**.

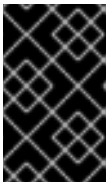
Procédure

1. Dans la section **[global]** du fichier **/etc/ipa/kdcproxy/kdcproxy.conf**, le paramètre **use_dns** doit être réglé sur **false**.

```
[global]
use_dns = false
```

2. Placez les informations relatives à la zone mandataire dans le fichier **/etc/ipa/kdcproxy/kdcproxy.conf**. Par exemple, pour la zone [AD. *EXAMPLE.COM*] avec proxy, listez les paramètres de configuration de la zone comme suit :

```
[AD.EXAMPLE.COM]
kerberos = kerberos+tcp://1.2.3.4:88 kerberos+tcp://5.6.7.8:88
kpasswd = kpasswd+tcp://1.2.3.4:464 kpasswd+tcp://5.6.7.8:464
```



IMPORTANT

Les paramètres de configuration du domaine doivent énumérer plusieurs serveurs séparés par un espace, contrairement à **/etc/krb5.conf** et **kdc.conf**, dans lesquels certaines options peuvent être spécifiées plusieurs fois.

3. Redémarrer les services de gestion des identités (IdM) :

```
# ipactl restart
```

Ressources supplémentaires

- Voir Configurer le serveur IPA en tant que proxy KDC pour la communication AD Kerberos dans la base de connaissances de Red Hat.

15.6. CONFIGURATION DU SERVEUR KKDCP II

La configuration de serveur suivante s'appuie sur les enregistrements du service DNS pour trouver les serveurs Active Directory (AD) avec lesquels communiquer.

Conditions préalables

- Vous avez accès à **root**.

Procédure

1. Dans le fichier **/etc/ipa/kdcproxy/kdcproxy.conf**, à la section **[global]**, le paramètre **use_dns** est remplacé par **true**.

```
[global]
configs = mit
use_dns = true
```

Le paramètre **configs** permet de charger d'autres modules de configuration. Dans ce cas, la configuration est lue à partir de la bibliothèque MIT **libkrb5**.

2. *Optional*: Si vous ne souhaitez pas utiliser les enregistrements de service DNS, ajoutez des serveurs AD explicites à la section **[realms]** du fichier **/etc/krb5.conf**. Si le domaine avec proxy est, par exemple, AD.*EXAMPLE.COM*, vous ajoutez :

```
[realms]
AD.EXAMPLE.COM = {
    kdc = ad-server.ad.example.com
    kpasswd_server = ad-server.ad.example.com
}
```

3. Redémarrer les services de gestion des identités (IdM) :

```
# ipactl restart
```

Ressources supplémentaires

- Voir Configurer le serveur IPA en tant que proxy KDC pour la communication AD Kerberos dans la base de connaissances de Red Hat.

CHAPITRE 16. GÉRER LES RÈGLES DE LIBRE-SERVICE DANS L'IDM À L'AIDE DU CLI

Ce chapitre présente les règles en libre-service dans la gestion des identités (IdM) et décrit comment créer et modifier des règles d'accès en libre-service dans l'interface de ligne de commande (CLI).

16.1. CONTRÔLE D'ACCÈS EN LIBRE-SERVICE DANS L'IDM

Les règles de contrôle d'accès en libre-service définissent les opérations qu'une entité de gestion des identités (IdM) peut effectuer sur son entrée du serveur d'annuaire IdM : par exemple, les utilisateurs IdM ont la possibilité de mettre à jour leurs propres mots de passe.

Cette méthode de contrôle permet à une entité IdM authentifiée de modifier des attributs spécifiques dans son entrée LDAP, mais n'autorise pas les opérations **add** ou **delete** sur l'ensemble de l'entrée.

AVERTISSEMENT

Soyez prudent lorsque vous utilisez des règles de contrôle d'accès en libre-service : une mauvaise configuration des règles de contrôle d'accès peut entraîner une élévation involontaire des privilèges d'une entité.

16.2. CRÉATION DE RÈGLES EN LIBRE-SERVICE À L'AIDE DE L'INTERFACE DE LIGNE DE COMMANDE

Cette procédure décrit la création de règles d'accès en libre-service dans IdM à l'aide de l'interface de ligne de commande (CLI).

Conditions préalables

- Privilèges d'administrateur pour la gestion de l'IdM ou le rôle **User Administrator**.
- Un ticket Kerberos actif. Pour plus de détails, voir Utilisation de kinit pour se connecter manuellement à IdM.

Procédure

- Pour ajouter une règle de libre-service, utilisez la commande **ipa selfservice-add** et spécifiez les deux options suivantes :

  --permissions

  définit les autorisations **read** et **write** accordées par l'instruction de contrôle d'accès (ACI).

  --attrs

  établit la liste complète des attributs auxquels l'ACI accorde une autorisation.

Par exemple, pour créer une règle de libre-service permettant aux utilisateurs de modifier les détails de leur propre nom :

```
$ ipa selfservice-add "Users can manage their own name details" --permissions=write --
attrs=givenname --attrs=displayname --attrs=title --attrs=initials
-----------------------------------------------------------
Added selfservice "Users can manage their own name details"
-----------------------------------------------------------
  Self-service name: Users can manage their own name details
  Permissions: write
  Attributes: givenname, displayname, title, initials
```

16.3. MODIFICATION DES RÈGLES DE LIBRE-SERVICE À L'AIDE DE L'INTERFACE DE LIGNE DE COMMANDE

Cette procédure décrit la modification des règles d'accès en libre-service dans IdM à l'aide de l'interface de ligne de commande (CLI).

Conditions préalables

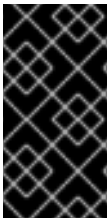
- Privilèges d'administrateur pour la gestion de l'IdM ou le rôle **User Administrator**.
- Un ticket Kerberos actif. Pour plus de détails, voir Utilisation de kinit pour se connecter manuellement à IdM.

Procédure

1. *Optional*: Affichez les règles de libre-service existantes à l'aide de la commande **ipa selfservice-find**.
2. *Optional*: Affiche les détails de la règle de libre-service que vous souhaitez modifier à l'aide de la commande **ipa selfservice-show**.
3. Utilisez la commande **ipa selfservice-mod** pour modifier une règle de libre-service.

Par exemple :

```
$ ipa selfservice-mod "Users can manage their own name details" --attrs=givenname --
attrs=displayname --attrs=title --attrs=initials --attrs=surname
-----------------------------------------------------------
Modified selfservice "Users can manage their own name details"
-----------------------------------------------------------
Self-service name: Users can manage their own name details
Permissions: write
Attributes: givenname, displayname, title, initials
```



IMPORTANT

L'utilisation de la commande **ipa selfservice-mod** écrase les autorisations et les attributs définis précédemment. Il convient donc de toujours inclure la liste complète des autorisations et des attributs existants, ainsi que les nouvelles autorisations et les nouveaux attributs que vous souhaitez définir.

Verification steps

- Utilisez la commande **ipa selfservice-show** pour afficher la règle de libre-service que vous avez modifiée.

```
$ ipa selfservice-show "Users can manage their own name details"
-------------------------------------------------------------
Self-service name: Users can manage their own name details
Permissions: write
Attributes: givenname, displayname, title, initials
```

16.4. SUPPRESSION DES RÈGLES DE LIBRE-SERVICE À L'AIDE DE L'INTERFACE DE LIGNE DE COMMANDE

Cette procédure décrit la suppression des règles d'accès en libre-service dans IdM à l'aide de l'interface de ligne de commande (CLI).

Conditions préalables

- Privilèges d'administrateur pour la gestion de l'IdM ou le rôle **User Administrator**.
- Un ticket Kerberos actif. Pour plus de détails, voir Utilisation de kinit pour se connecter manuellement à IdM.

Procédure

- Utilisez la commande **ipa selfservice-del** pour supprimer une règle de libre-service.

Par exemple :

```
$ ipa selfservice-del "Users can manage their own name details"
-------------------------------------------------------------
Deleted selfservice "Users can manage their own name details"
-------------------------------------------------------------
```

Verification steps

- Utilisez la commande **ipa selfservice-find** pour afficher toutes les règles de libre-service. La règle que vous venez de supprimer devrait être absente.

CHAPITRE 17. GESTION DES RÈGLES DE LIBRE-SERVICE À L'AIDE DE L'INTERFACE WEB IDM

Ce chapitre présente les règles en libre-service dans la gestion des identités (IdM) et décrit comment créer et modifier des règles d'accès en libre-service dans l'interface web (IdM Web UI).

17.1. CONTRÔLE D'ACCÈS EN LIBRE-SERVICE DANS L'IDM

Les règles de contrôle d'accès en libre-service définissent les opérations qu'une entité de gestion des identités (IdM) peut effectuer sur son entrée du serveur d'annuaire IdM : par exemple, les utilisateurs IdM ont la possibilité de mettre à jour leurs propres mots de passe.

Cette méthode de contrôle permet à une entité IdM authentifiée de modifier des attributs spécifiques dans son entrée LDAP, mais n'autorise pas les opérations **add** ou **delete** sur l'ensemble de l'entrée.

AVERTISSEMENT

Soyez prudent lorsque vous utilisez des règles de contrôle d'accès en libre-service : une mauvaise configuration des règles de contrôle d'accès peut entraîner une élévation involontaire des privilèges d'une entité.

17.2. CRÉATION DE RÈGLES EN LIBRE-SERVICE À L'AIDE DE L'INTERFACE WEB IDM

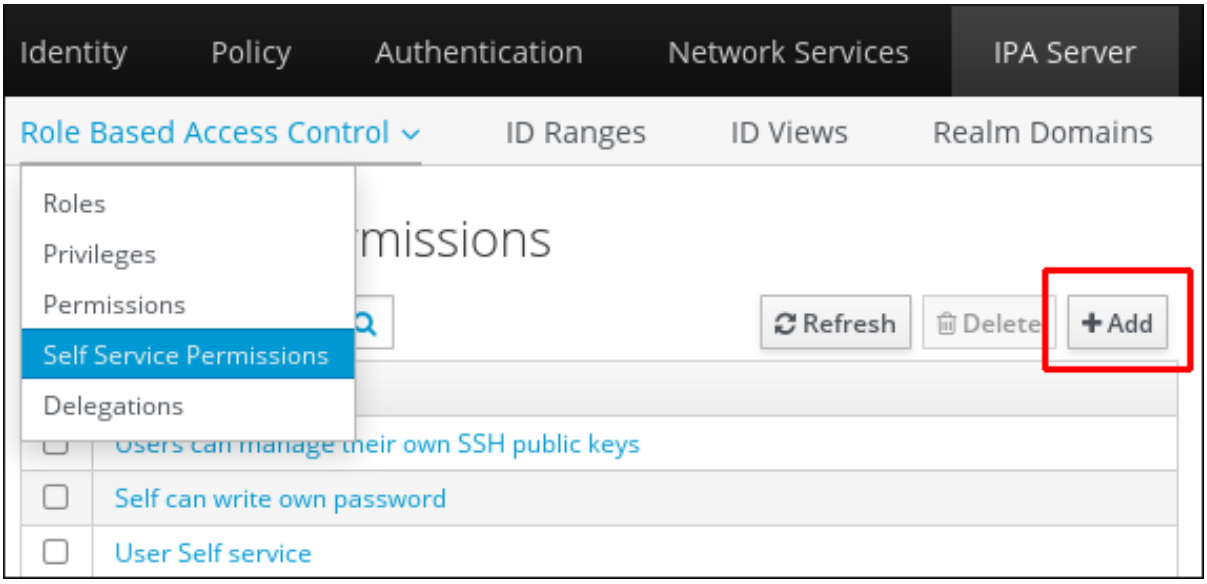
Cette procédure décrit comment créer des règles d'accès en libre-service dans IdM à l'aide de l'interface web (IdM Web UI).

Conditions préalables

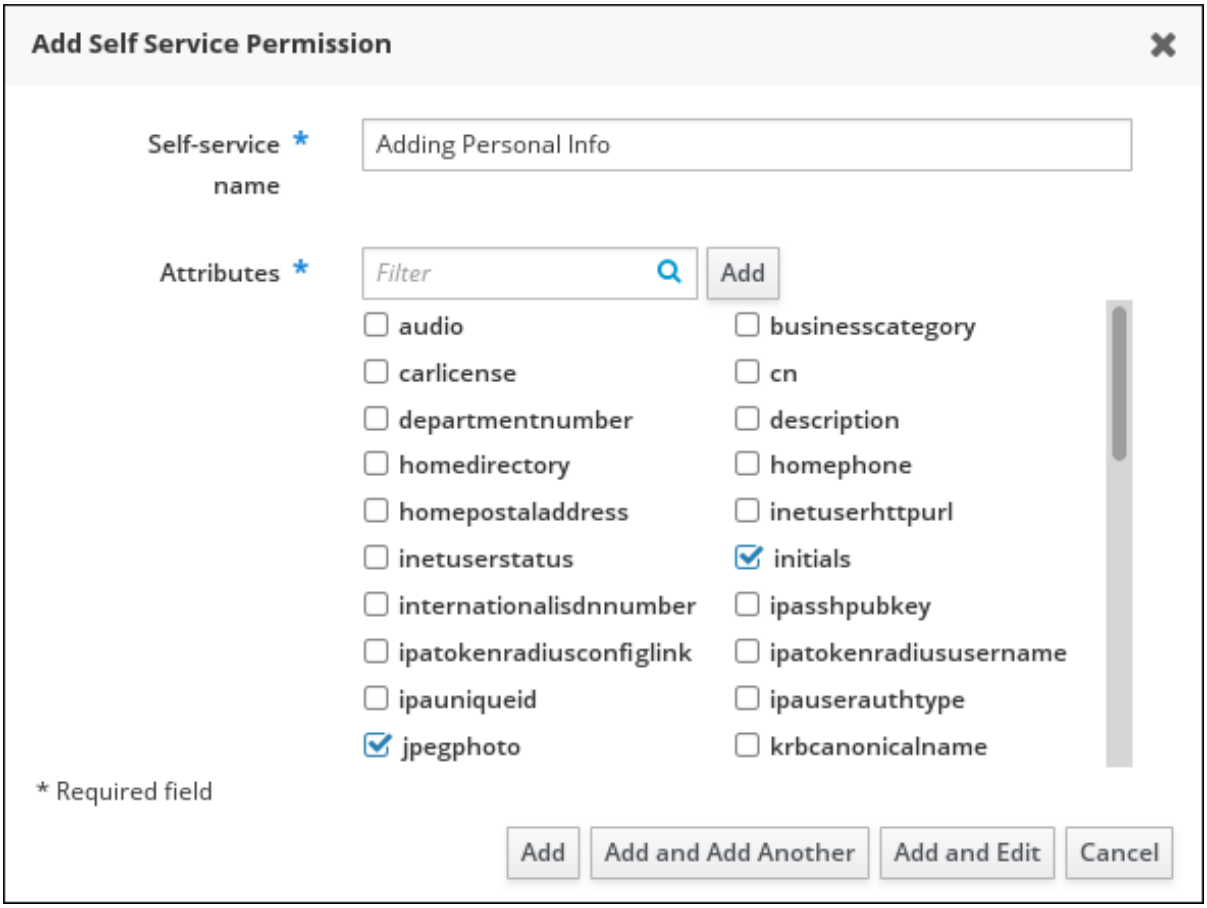
- Privilèges d'administrateur pour la gestion de l'IdM ou le rôle **User Administrator**.
- Vous êtes connecté à l'interface Web IdM. Pour plus de détails, voir Accès à l'IdM Web UI dans un navigateur web.

Procédure

1. Ouvrez le sous-menu **Role-Based Access Control** dans l'onglet **IPA Server** et sélectionnez **Self Service Permissions**.
2. Cliquez sur **Add** en haut à droite de la liste des règles d'accès en libre-service :



3. La fenêtre **Add Self Service Permission** s'ouvre. Saisissez le nom de la nouvelle règle de libre-service dans le champ **Self-service name**. Les espaces sont autorisés :



4. Cochez les cases en regard des attributs que vous souhaitez que les utilisateurs puissent modifier.

5. *Optional*: Si un attribut auquel vous souhaitez donner accès n'est pas répertorié, vous pouvez en ajouter un :

   a. Cliquez sur le bouton **Add**.

   b. Saisissez le nom de l'attribut dans le champ de texte **Attribute** de la fenêtre suivante **Add Custom Attribute**.

c. Cliquez sur le bouton **OK** pour ajouter l'attribut

d. Vérifier que le nouvel attribut est sélectionné

6. Cliquez sur le bouton **Add** au bas du formulaire pour enregistrer la nouvelle règle de libre-service.
Vous pouvez également enregistrer et continuer à modifier la règle de libre-service en cliquant sur le bouton **Add and Edit**, ou enregistrer et ajouter d'autres règles en cliquant sur le bouton **Add and Add another**.

17.3. MODIFICATION DES RÈGLES DE LIBRE-SERVICE À L'AIDE DE L'INTERFACE WEB IDM

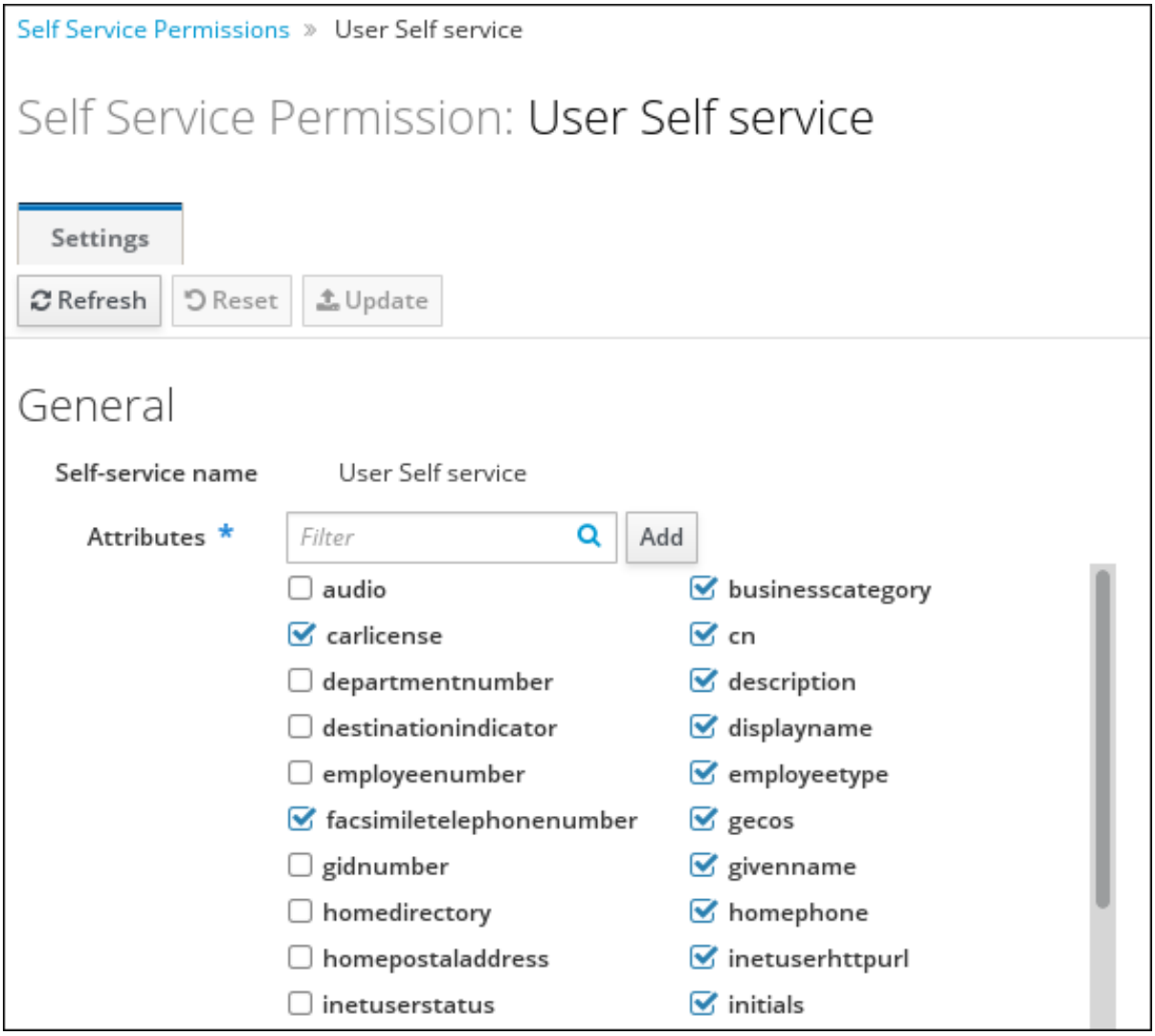
Cette procédure décrit comment modifier les règles d'accès en libre-service dans IdM à l'aide de l'interface web (IdM Web UI).

Conditions préalables

- Privilèges d'administrateur pour la gestion de l'IdM ou le rôle **User Administrator**.
- Vous êtes connecté à l'interface Web IdM. Pour plus de détails, voir Accès à l'IdM Web UI dans un navigateur web.

Procédure

1. Ouvrez le sous-menu **Role-Based Access Control** dans l'onglet **IPA Server** et sélectionnez **Self Service Permissions**.
2. Cliquez sur le nom de la règle de libre-service que vous souhaitez modifier.



3. La page d'édition vous permet uniquement de modifier la liste des attributs que vous souhaitez ajouter ou supprimer de la règle de libre-service. Cochez ou décochez les cases appropriées.

4. Cliquez sur le bouton **Save** pour enregistrer les modifications apportées à la règle de libre-service.

17.4. SUPPRESSION DES RÈGLES DE LIBRE-SERVICE À L'AIDE DE L'INTERFACE WEB IDM

Cette procédure décrit comment supprimer les règles d'accès en libre-service dans IdM à l'aide de l'interface web (IdM Web UI).

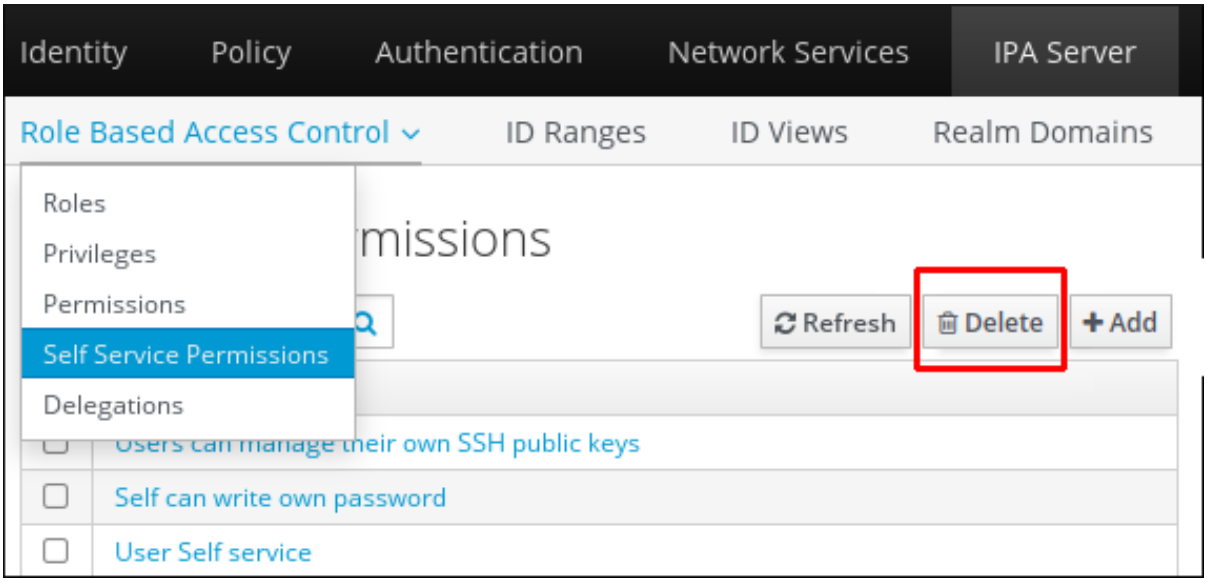
Conditions préalables

- Privilèges d'administrateur pour la gestion de l'IdM ou le rôle **User Administrator**.
- Vous êtes connecté à l'interface Web IdM. Pour plus de détails, voir Accès à l'IdM Web UI dans un navigateur web.

Procédure

1. Ouvrez le sous-menu **Role-Based Access Control** dans l'onglet **IPA Server** et sélectionnez **Self Service Permissions**.

2. Cochez la case en regard de la règle que vous souhaitez supprimer, puis cliquez sur le bouton **Delete** à droite de la liste.



3. Une boîte de dialogue s'ouvre, cliquez sur **Delete** pour confirmer.

CHAPITRE 18. UTILISER LES PLAYBOOKS ANSIBLE POUR GÉRER LES RÈGLES DE SELF-SERVICE DANS L'IDM

Cette section présente les règles en libre-service dans la gestion des identités (IdM) et décrit comment créer et modifier des règles d'accès en libre-service à l'aide des playbooks Ansible. Les règles de contrôle d'accès en libre-service permettent à une entité IdM d'effectuer des opérations spécifiques sur son entrée du serveur d'annuaire IdM.

Cette section couvre les sujets suivants :

- Contrôle d'accès en libre-service dans l'IdM
- Utiliser Ansible pour s'assurer qu'une règle de libre-service est présente
- Utiliser Ansible pour s'assurer qu'une règle de libre-service est absente
- Utiliser Ansible pour s'assurer qu'une règle de libre-service possède des attributs spécifiques
- Utiliser Ansible pour s'assurer qu'une règle de libre-service n'a pas d'attributs spécifiques

18.1. CONTRÔLE D'ACCÈS EN LIBRE-SERVICE DANS L'IDM

Les règles de contrôle d'accès en libre-service définissent les opérations qu'une entité de gestion des identités (IdM) peut effectuer sur son entrée du serveur d'annuaire IdM : par exemple, les utilisateurs IdM ont la possibilité de mettre à jour leurs propres mots de passe.

Cette méthode de contrôle permet à une entité IdM authentifiée de modifier des attributs spécifiques dans son entrée LDAP, mais n'autorise pas les opérations **add** ou **delete** sur l'ensemble de l'entrée.

AVERTISSEMENT

Soyez prudent lorsque vous utilisez des règles de contrôle d'accès en libre-service : une mauvaise configuration des règles de contrôle d'accès peut entraîner une élévation involontaire des privilèges d'une entité.

18.2. UTILISER ANSIBLE POUR S'ASSURER QU'UNE RÈGLE DE LIBRE-SERVICE EST PRÉSENTE

La procédure suivante décrit comment utiliser un playbook Ansible pour définir des règles de libre-service et assurer leur présence sur un serveur de gestion des identités (IdM). Dans cet exemple, la nouvelle règle **Users can manage their own name details**permet aux utilisateurs de modifier leurs propres attributs **givenname**, **displayname**, **title** et **initials**. Cela leur permet, par exemple, de modifier leur nom d'affichage ou leurs initiales s'ils le souhaitent.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.

- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

Procédure

1. Naviguez jusqu'au répertoire ***~/MyPlaybooks/*** répertoire :

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

2. Faites une copie du fichier **selfservice-present.yml** situé dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/selfservice/**:

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/selfservice/selfservice-present.yml
selfservice-present-copy.yml
```

3. Ouvrez le fichier **selfservice-present-copy.yml** Ansible playbook pour l'éditer.
4. Adaptez le fichier en définissant les variables suivantes dans la section **ipaselfservice** task :
   - Définissez la variable **ipaadmin_password** avec le mot de passe de l'administrateur IdM.
   - Définissez la variable **name** avec le nom de la nouvelle règle de libre-service.
   - Attribuez à la variable **permission** une liste de permissions à accorder, séparées par des virgules : **read** et **write**.
   - Définissez la variable **attribute** avec une liste d'attributs que les utilisateurs peuvent gérer eux-mêmes : **givenname**, **displayname**, **title**, et **initials**.

   Il s'agit du fichier playbook Ansible modifié pour l'exemple actuel :

```
---
- name: Self-service present
  hosts: ipaserver

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Ensure self-service rule "Users can manage their own name details" is present
    ipaselfservice:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: "Users can manage their own name details"
      permission: read, write
      attribute:
      - givenname
```

```
    - displayname
    - title
    - initials
```

5. Enregistrer le fichier.

6. Exécutez le playbook Ansible. Spécifiez le fichier du livre de jeu, le fichier contenant le mot de passe protégeant le fichier **secret.yml** et le fichier d'inventaire :

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

Ressources supplémentaires

- Voir Contrôle d'accès en libre-service dans IdM .
- Voir le fichier **README-selfservice.md** dans le répertoire **/usr/share/doc/ansible-freeipa/**.
- Voir le répertoire **/usr/share/doc/ansible-freeipa/playbooks/selfservice**.

18.3. UTILISER ANSIBLE POUR S'ASSURER QU'UNE RÈGLE DE LIBRE-SERVICE EST ABSENTE

La procédure suivante décrit comment utiliser un playbook Ansible pour s'assurer qu'une règle de libre-service spécifiée est absente de votre configuration IdM. L'exemple ci-dessous décrit comment s'assurer que la règle de libre-service **Users can manage their own name details** n'existe pas dans IdM. Cela garantit que les utilisateurs ne peuvent pas, par exemple, modifier leur propre nom d'affichage ou leurs initiales.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

Procédure

1. Naviguez jusqu'au répertoire ***~/MyPlaybooks/*** répertoire :

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

2. Faites une copie du fichier **selfservice-absent.yml** situé dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/selfservice/**:

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/selfservice/selfservice-absent.yml
selfservice-absent-copy.yml
```

3. Ouvrez le fichier **selfservice-absent-copy.yml** Ansible playbook pour l'éditer.
4. Adaptez le fichier en définissant les variables suivantes dans la section **ipaselfservice** task :
   - Définissez la variable **ipaadmin_password** avec le mot de passe de l'administrateur IdM.
   - Définissez la variable **name** avec le nom de la règle de libre-service.
   - Fixer la variable **state** à **absent**.

   Il s'agit du fichier playbook Ansible modifié pour l'exemple actuel :

```
---
- name: Self-service absent
  hosts: ipaserver

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Ensure self-service rule "Users can manage their own name details" is absent
    ipaselfservice:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: "Users can manage their own name details"
      state: absent
```

5. Enregistrer le fichier.
6. Exécutez le playbook Ansible. Spécifiez le fichier du livre de jeu, le fichier contenant le mot de passe protégeant le fichier **secret.yml** et le fichier d'inventaire :

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

Ressources supplémentaires

- Voir Contrôle d'accès en libre-service dans IdM .
- Voir le fichier **README-selfservice.md** dans le répertoire **/usr/share/doc/ansible-freeipa/**.
- Voir les exemples de playbooks dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/selfservice**.

18.4. UTILISER ANSIBLE POUR S'ASSURER QU'UNE RÈGLE DE LIBRE-SERVICE POSSÈDE DES ATTRIBUTS SPÉCIFIQUES

La procédure suivante décrit comment utiliser un playbook Ansible pour s'assurer qu'une règle de libre-service existante possède des paramètres spécifiques. Dans l'exemple, vous vous assurez que la règle de libre-service **Users can manage their own name details** possède également l'attribut de membre **surname**.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.
- La règle de libre-service **Users can manage their own name details** existe dans IdM.

Procédure

1. Naviguez jusqu'au répertoire ***~/MyPlaybooks/*** répertoire :

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

2. Faites une copie du fichier **selfservice-member-present.yml** situé dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/selfservice/**:

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/selfservice/selfservice-member-present.yml selfservice-member-present-copy.yml
```

3. Ouvrez le fichier **selfservice-member-present-copy.yml** Ansible playbook pour l'éditer.
4. Adaptez le fichier en définissant les variables suivantes dans la section **ipaselfservice** task :
   - Définissez la variable **ipaadmin_password** avec le mot de passe de l'administrateur IdM.
   - Définissez la variable **name** avec le nom de la règle de libre-service à modifier.
   - Fixer la variable **attribute** à **surname**.
   - Fixer la variable **action** à **member**.

   Il s'agit du fichier playbook Ansible modifié pour l'exemple actuel :

```
---
- name: Self-service member present
  hosts: ipaserver

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Ensure selfservice "Users can manage their own name details" member attribute surname is present
    ipaselfservice:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: "Users can manage their own name details"
```

```
attribute:
- surname
action: member
```

5. Enregistrer le fichier.

6. Exécutez le playbook Ansible. Spécifiez le fichier du livre de jeu, le fichier contenant le mot de passe protégeant le fichier **secret.yml** et le fichier d'inventaire :

```
$ ansible-playbook --vault-password-file=password_file -v -i inventory selfservice-member-present-copy.yml
```

Ressources supplémentaires

- Voir Contrôle d'accès en libre-service dans IdM .
- Voir le fichier **README-selfservice.md** disponible dans le répertoire **/usr/share/doc/ansible-freeipa/**.
- Voir les exemples de playbooks dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/selfservice**.

18.5. UTILISER ANSIBLE POUR S'ASSURER QU'UNE RÈGLE DE LIBRE-SERVICE N'A PAS D'ATTRIBUTS SPÉCIFIQUES

La procédure suivante décrit comment utiliser une séquence Ansible pour s'assurer qu'une règle de libre-service n'a pas de paramètres spécifiques. Vous pouvez utiliser ce livre de lecture pour vous assurer qu'une règle de libre-service n'accorde pas d'accès indésirable. Dans l'exemple, vous vous assurez que la règle de libre-service **Users can manage their own name details** n'a pas les attributs de membre **givenname** et **surname**.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.
- La règle de libre-service **Users can manage their own name details** existe dans IdM.

Procédure

1. Naviguez jusqu'au répertoire ***~/MyPlaybooks/*** répertoire :

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

2. Faites une copie du fichier **selfservice-member-absent.yml** situé dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/selfservice/**:

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/selfservice/selfservice-member-absent.yml selfservice-member-absent-copy.yml
```

3. Ouvrez le fichier **selfservice-member-absent-copy.yml** Ansible playbook pour l'éditer.

4. Adaptez le fichier en définissant les variables suivantes dans la section **ipaselfservice** task :

   - Définissez la variable **ipaadmin_password** avec le mot de passe de l'administrateur IdM.
   - Définissez la variable **name** avec le nom de la règle de libre-service que vous souhaitez modifier.
   - Fixez la variable **attribute** à **givenname** et **surname**.
   - Fixer la variable **action** à **member**.
   - Fixer la variable **state** à **absent**.

   Il s'agit du fichier playbook Ansible modifié pour l'exemple actuel :

```
---
- name: Self-service member absent
  hosts: ipaserver

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Ensure selfservice "Users can manage their own name details" member attributes givenname and surname are absent
    ipaselfservice:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: "Users can manage their own name details"
      attribute:
      - givenname
      - surname
      action: member
      state: absent
```

5. Enregistrer le fichier.

6. Exécutez le playbook Ansible. Spécifiez le fichier du livre de jeu, le fichier contenant le mot de passe protégeant le fichier **secret.yml** et le fichier d'inventaire :

```
$ ansible-playbook --vault-password-file=password_file -v -i inventory selfservice-member-absent-copy.yml
```

Ressources supplémentaires

- Voir Contrôle d'accès en libre-service dans IdM .

- Voir le fichier **README-selfservice.md** dans le répertoire **/usr/share/doc/ansible-freeipa/**.
- Voir les exemples de playbooks dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/selfservice**.

CHAPITRE 19. GESTION DES GROUPES D'UTILISATEURS DANS L'INTERFACE CLI DE L'IDM

Ce chapitre présente la gestion des groupes d'utilisateurs à l'aide de la CLI IdM.

Un groupe d'utilisateurs est un ensemble d'utilisateurs ayant des privilèges, des politiques de mot de passe et d'autres caractéristiques communes.

Dans le cadre de la gestion des identités (IdM), un groupe d'utilisateurs peut inclure :

- Utilisateurs de l'IdM
- d'autres groupes d'utilisateurs de l'IdM
- les utilisateurs externes, c'est-à-dire les utilisateurs qui existent en dehors de l'IdM

19.1. LES DIFFÉRENTS TYPES DE GROUPES DANS L'IDM

IdM prend en charge les types de groupes suivants :

Groupes POSIX (par défaut)

Les groupes POSIX prennent en charge les attributs Linux POSIX pour leurs membres. Notez que les groupes qui interagissent avec Active Directory ne peuvent pas utiliser les attributs POSIX.
Les attributs POSIX identifient les utilisateurs en tant qu'entités distinctes. Parmi les exemples d'attributs POSIX concernant les utilisateurs, on peut citer **uidNumber**, un numéro d'utilisateur (UID), et **gidNumber**, un numéro de groupe (GID).

Groupes non-POSIX

Les groupes non-POSIX ne prennent pas en charge les attributs POSIX. Par exemple, ces groupes n'ont pas de GID défini.
Tous les membres de ce type de groupe doivent appartenir au domaine IdM.

Groupes externes

Utilisez les groupes externes pour ajouter des membres de groupes qui existent dans un magasin d'identité en dehors du domaine IdM, par exemple :

- Un système local
- Un domaine Active Directory
- Un service d'annuaire

Les groupes externes ne prennent pas en charge les attributs POSIX. Par exemple, ces groupes n'ont pas de GID défini.

Tableau 19.1. Groupes d'utilisateurs créés par défaut

Nom du groupe	Membres du groupe par défaut
ipausers	Tous les utilisateurs de l'IdM

Nom du groupe	Membres du groupe par défaut
admins	Utilisateurs disposant de privilèges administratifs, y compris l'utilisateur par défaut **admin**
editors	Il s'agit d'un groupe ancien qui ne bénéficie plus de privilèges particuliers
trust admins	Utilisateurs ayant des privilèges pour gérer les trusts Active Directory

Lorsque vous ajoutez un utilisateur à un groupe d'utilisateurs, celui-ci bénéficie des privilèges et des règles associés au groupe. Par exemple, pour accorder des privilèges administratifs à un utilisateur, ajoutez-le au groupe **admins**.

AVERTISSEMENT

Ne pas supprimer le groupe **admins**. Comme **admins** est un groupe prédéfini requis par IdM, cette opération pose des problèmes avec certaines commandes.

En outre, IdM crée *user private groups* par défaut chaque fois qu'un nouvel utilisateur est créé dans IdM. Pour plus d'informations sur les groupes privés, voir Ajouter des utilisateurs sans groupe privé.

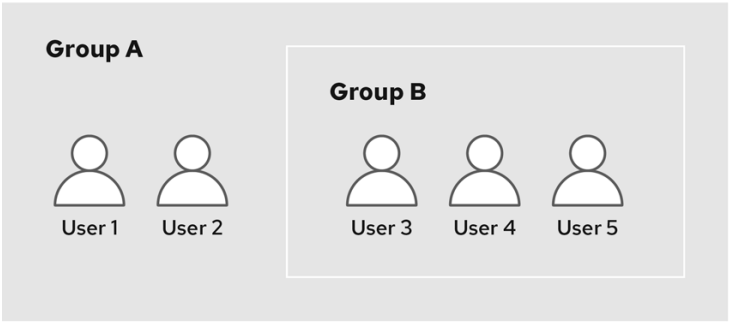
19.2. MEMBRES DIRECTS ET INDIRECTS DU GROUPE

Les attributs des groupes d'utilisateurs dans l'IdM s'appliquent à la fois aux membres directs et indirects : lorsque le groupe B est membre du groupe A, tous les utilisateurs du groupe B sont considérés comme des membres indirects du groupe A.

Par exemple, dans le diagramme suivant :

- L'utilisateur 1 et l'utilisateur 2 sont *direct members* du groupe A.
- L'utilisateur 3, l'utilisateur 4 et l'utilisateur 5 sont *indirect members* du groupe A.

Figure 19.1. Appartenance directe et indirecte à un groupe



Si vous définissez une politique de mot de passe pour le groupe d'utilisateurs A, cette politique s'applique également à tous les utilisateurs du groupe d'utilisateurs B.

19.3. AJOUT D'UN GROUPE D'UTILISATEURS À L'AIDE DE LA CLI IDM

Cette section décrit comment ajouter un groupe d'utilisateurs à l'aide de la CLI IdM.

Conditions préalables

- Vous devez être connecté en tant qu'administrateur. Pour plus de détails, voir Utilisation de kinit pour se connecter manuellement à l'IdM.

Procédure

- Ajoutez un groupe d'utilisateurs à l'aide de la commande **ipa group-add *group_name*** pour ajouter un groupe d'utilisateurs. Par exemple, pour créer le groupe_a :

```
$ ipa group-add group_a
---------------------
Added group "group_a"
---------------------
  Group name: group_a
  GID: 1133400009
```

  Par défaut, **ipa group-add** ajoute un groupe d'utilisateurs POSIX. Pour spécifier un autre type de groupe, ajoutez des options à **ipa group-add**:

  - **--nonposix** pour créer un groupe non-POSIX
  - **--external** pour créer un groupe externe
    Pour plus de détails sur les types de groupes, voir Les différents types de groupes dans IdM .

  Vous pouvez spécifier un GID personnalisé lors de l'ajout d'un groupe d'utilisateurs en utilisant l'option **--gid=*custom_GID*** pour ajouter un groupe d'utilisateurs. Dans ce cas, veillez à éviter les conflits d'identifiants. Si vous ne spécifiez pas de GID personnalisé, IdM attribue automatiquement un GID à partir de la plage d'ID disponible.

AVERTISSEMENT

N'ajoutez pas de groupes locaux à IdM. Le commutateur de service de noms (NSS) résout toujours les utilisateurs et les groupes IdM avant de résoudre les utilisateurs et les groupes locaux. Cela signifie, par exemple, que l'appartenance à un groupe IdM ne fonctionne pas pour les utilisateurs locaux.

19.4. RECHERCHE DE GROUPES D'UTILISATEURS À L'AIDE DE LA CLI IDM

Cette section décrit comment rechercher des groupes d'utilisateurs existants à l'aide de l'interface CLI de l'IdM.

Procédure

- Affichez tous les groupes d'utilisateurs à l'aide de la commande **ipa group-find**. Pour spécifier un type de groupe, ajoutez des options à **ipa group-find**:
  - Affichez tous les groupes POSIX à l'aide de la commande **ipa group-find --posix**.
  - Affichez tous les groupes non-POSIX à l'aide de la commande **ipa group-find --nonposix**.
  - Affichez tous les groupes externes à l'aide de la commande **ipa group-find --external**. Pour plus d'informations sur les différents types de groupes, voir Les différents types de groupes dans IdM.

19.5. SUPPRESSION D'UN GROUPE D'UTILISATEURS À L'AIDE DE LA CLI IDM

Cette section décrit comment supprimer un groupe d'utilisateurs à l'aide de l'interface CLI de l'IdM. Notez que la suppression d'un groupe ne supprime pas les membres du groupe de l'IdM.

Conditions préalables

- Vous devez être connecté en tant qu'administrateur. Pour plus de détails, voir Utilisation de kinit pour se connecter manuellement à l'IdM.

Procédure

- Supprimez un groupe d'utilisateurs à l'aide de la commande **ipa group-del *group_name*** pour supprimer un groupe d'utilisateurs. Par exemple, pour supprimer le groupe_a :

```
$ ipa group-del group_a
-------------------------
Deleted group "group_a"
-------------------------
```

19.6. AJOUT D'UN MEMBRE À UN GROUPE D'UTILISATEURS À L'AIDE DE LA CLI IDM

Cette section décrit comment ajouter un membre à un groupe d'utilisateurs à l'aide de la CLI IdM. Vous pouvez ajouter des utilisateurs et des groupes d'utilisateurs en tant que membres d'un groupe d'utilisateurs. Pour plus d'informations, voir Les différents types de groupes dans IdM et Les membres directs et indirects d'un groupe.

Conditions préalables

- Vous devez être connecté en tant qu'administrateur. Pour plus de détails, voir Utilisation de kinit pour se connecter manuellement à l'IdM.

Procédure

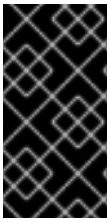
- Ajoutez un membre à un groupe d'utilisateurs à l'aide de la commande **ipa group-add-member**. Spécifiez le type de membre à l'aide de ces options :
  - **--users** ajoute un utilisateur IdM

- **--external** ajoute un utilisateur qui existe en dehors du domaine IdM, au format **DOMAIN\user_name** ou **user_name@domain**
- **--groups** ajoute un groupe d'utilisateurs IdM

Par exemple, pour ajouter le groupe_b comme membre du groupe_a :

```
$ ipa group-add-member group_a --groups=group_b
Group name: group_a
GID: 1133400009
Member users: user_a
Member groups: group_b
Indirect Member users: user_b
-------------------------
Number of members added 1
-------------------------
```

Les membres du groupe_b sont désormais des membres indirects du groupe_a.



IMPORTANT

Lorsque vous ajoutez un groupe en tant que membre d'un autre groupe, ne créez pas de groupes récursifs. Par exemple, si le groupe A est membre du groupe B, n'ajoutez pas le groupe B comme membre du groupe A. Les groupes récursifs peuvent avoir un comportement imprévisible.

NOTE

Après avoir ajouté un membre à un groupe d'utilisateurs, la mise à jour peut prendre un certain temps avant de se propager à tous les clients de votre environnement de gestion des identités. En effet, lorsqu'un hôte donné résout des utilisateurs, des groupes et des groupes nets, le site **System Security Services Daemon** (SSSD) consulte d'abord son cache et n'effectue des recherches sur le serveur qu'en cas d'enregistrements manquants ou périmés.

19.7. AJOUT D'UTILISATEURS SANS GROUPE PRIVÉ D'UTILISATEURS

Par défaut, l'IdM crée des groupes privés d'utilisateurs (UPG) chaque fois qu'un nouvel utilisateur est créé dans l'IdM. Les UPG sont un type de groupe spécifique :

- L'UPG porte le même nom que l'utilisateur nouvellement créé.
- L'utilisateur est le seul membre de l'UPG. L'UPG ne peut pas contenir d'autres membres.
- Le GID du groupe privé correspond à l'UID de l'utilisateur.

Il est toutefois possible d'ajouter des utilisateurs sans créer de GUP.

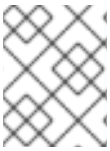
19.7.1. Utilisateurs sans groupe privé d'utilisateurs

Si un groupe NIS ou un autre groupe système utilise déjà le GID qui serait attribué à un groupe privé d'utilisateurs, il faut éviter de créer un UPG.

Vous pouvez le faire de deux manières :

- Ajouter un nouvel utilisateur sans UPG, sans désactiver globalement les groupes privés. Voir Ajouter un utilisateur sans groupe privé lorsque les groupes privés sont activés globalement .
- Désactivez globalement les groupes privés d'utilisateurs pour tous les utilisateurs, puis ajoutez un nouvel utilisateur. Voir Désactiver globalement les groupes privés d'utilisateurs pour tous les utilisateurs et Ajouter un utilisateur lorsque les groupes privés d'utilisateurs sont globalement désactivés.

Dans les deux cas, l'IdM exigera la spécification d'un GID lors de l'ajout de nouveaux utilisateurs, sinon l'opération échouera. En effet, l'IdM a besoin d'un GID pour le nouvel utilisateur, mais le groupe d'utilisateurs par défaut **ipausers** est un groupe non-POSIX et n'a donc pas de GID associé. Le GID que vous spécifiez ne doit pas nécessairement correspondre à un groupe déjà existant.



NOTE

La spécification du GID ne crée pas de nouveau groupe. Elle définit uniquement l'attribut GID pour le nouvel utilisateur, car cet attribut est requis par l'IdM.

19.7.2. Ajout d'un utilisateur sans groupe privé lorsque les groupes privés sont globalement activés

Vous pouvez ajouter un utilisateur sans créer de groupe privé d'utilisateurs (UPG), même si les UPG sont activés sur le système. Pour ce faire, vous devez définir manuellement un GID pour le nouvel utilisateur. Pour plus de détails sur les raisons de cette opération, voir Utilisateurs sans groupe privé d'utilisateurs.

Procédure

- Pour empêcher l'IdM de créer un UPG, ajoutez l'option **--noprivate** à la commande **ipa user-add**.
  Notez que pour que la commande aboutisse, vous devez spécifier un GID personnalisé. Par exemple, pour ajouter un nouvel utilisateur avec le GID 10000 :

```
$ ipa user-add jsmith --first=John --last=Smith --noprivate --gid 10000
```

19.7.3. Désactivation globale des groupes privés d'utilisateurs pour tous les utilisateurs

Vous pouvez désactiver globalement les groupes privés d'utilisateurs (UPG). Cela empêche la création de groupes privés d'utilisateurs pour tous les nouveaux utilisateurs. Les utilisateurs existants ne sont pas affectés par cette modification.

Procédure

1. Obtenir les privilèges d'administrateur :

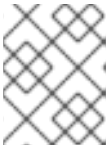
```
kinit admin
```

2. IdM utilise le plug-in Directory Server Managed Entries pour gérer les UPG. Dressez la liste des instances du plug-in :

```
$ ipa-managed-entries --list
```

3. Pour s'assurer que IdM ne crée pas d'UPG, désactivez l'instance de plug-in responsable de la gestion des groupes privés d'utilisateurs :

```
$ ipa-managed-entries -e "UPG Definition" disable
Disabling Plugin
```



NOTE

Pour réactiver l'instance **UPG Definition** ultérieurement, utilisez la commande **ipa-managed-entries -e "UPG Definition" enable**.

4. Redémarrez Directory Server pour charger la nouvelle configuration.

```
sudo systemctl restart dirsrv.target
```

Pour ajouter un utilisateur après la désactivation des UPG, vous devez spécifier un GID. Pour plus d'informations, voir Ajouter un utilisateur lorsque les groupes privés d'utilisateurs sont globalement désactivés

Verification steps

- Pour vérifier si les UPG sont globalement désactivés, utilisez à nouveau la commande disable :

```
$ ipa-managed-entries -e "UPG Definition" disable
Plugin already disabled
```

19.7.4. Ajout d'un utilisateur lorsque les groupes privés d'utilisateurs sont globalement désactivés

Lorsque les groupes privés d'utilisateurs (UPG) sont désactivés globalement, l'IdM n'attribue pas automatiquement un GID à un nouvel utilisateur. Pour ajouter un utilisateur avec succès, vous devez attribuer un GID manuellement ou à l'aide d'une règle automember. Pour plus d'informations, voir Utilisateurs sans groupe privé d'utilisateurs.

Conditions préalables

- Les groupes privés d'utilisateurs doivent être désactivés globalement pour tous les utilisateurs. Pour plus d'informations, voir Désactivation globale des groupes privés d'utilisateurs pour tous les utilisateurs

Procédure

- Pour s'assurer que l'ajout d'un nouvel utilisateur réussit lorsque la création d'UPG est désactivée, choisissez l'une des options suivantes :
  - Spécifiez un GID personnalisé lors de l'ajout d'un nouvel utilisateur. Le GID ne doit pas nécessairement correspondre à un groupe d'utilisateurs existant.
    Par exemple, lorsque vous ajoutez un utilisateur à partir de la ligne de commande, ajoutez l'option **--gid** à la commande **ipa user-add**.
  - Utilisez une règle automember pour ajouter l'utilisateur à un groupe existant avec un GID.

19.8. AJOUT D'UTILISATEURS OU DE GROUPES EN TANT QUE GESTIONNAIRES MEMBRES D'UN GROUPE D'UTILISATEURS IDM À L'AIDE DE LA CLI IDM

Cette section décrit comment ajouter des utilisateurs ou des groupes en tant que gestionnaires membres d'un groupe d'utilisateurs IdM à l'aide de la CLI IdM. Les gestionnaires membres peuvent ajouter des utilisateurs ou des groupes aux groupes d'utilisateurs IdM, mais ne peuvent pas modifier les attributs d'un groupe.

Conditions préalables

- Vous devez être connecté en tant qu'administrateur. Pour plus de détails, voir Utilisation de kinit pour se connecter manuellement à l'IdM.
- Vous devez disposer du nom de l'utilisateur ou du groupe que vous ajoutez en tant que membres gestionnaires et du nom du groupe que vous souhaitez qu'ils gèrent.

Procédure

- Ajouter un utilisateur en tant que gestionnaire membre d'un groupe d'utilisateurs IdM à l'aide de la commande **ipa group-add-member-manager**.
  Par exemple, pour ajouter l'utilisateur **test** en tant que membre gestionnaire de **group_a**:

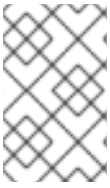
```
$ ipa group-add-member-manager group_a --users=test
Group name: group_a
GID: 1133400009
Membership managed by users: test
-------------------------
Number of members added 1
-------------------------
```

  L'utilisateur **test** peut désormais gérer les membres de **group_a**.

- Ajouter un groupe en tant que gestionnaire membre d'un groupe d'utilisateurs IdM à l'aide de la commande **ipa group-add-member-manager**.
  Par exemple, pour ajouter le groupe **group_admins** en tant que gestionnaire membre de **group_a**:

```
$ ipa group-add-member-manager group_a --groups=group_admins
Group name: group_a
GID: 1133400009
Membership managed by groups: group_admins
Membership managed by users: test
-------------------------
Number of members added 1
-------------------------
```

  Le groupe **group_admins** peut désormais gérer les membres de **group_a**.



NOTE

Après avoir ajouté un gestionnaire membre à un groupe d'utilisateurs, la mise à jour peut prendre un certain temps avant de se propager à tous les clients de votre environnement de gestion des identités.

Verification steps

- Utiliser la commande **ipa group-show** pour vérifier que l'utilisateur et le groupe ont été ajoutés en tant que gestionnaires membres.

```
$ ipa group-show group_a
Group name: group_a
GID: 1133400009
Membership managed by groups: group_admins
Membership managed by users: test
```

Ressources supplémentaires

- Voir **ipa group-add-member-manager --help** pour plus de détails.

19.9. VISUALISATION DES MEMBRES D'UN GROUPE À L'AIDE DE LA CLI IDM

Cette section décrit comment visualiser les membres d'un groupe à l'aide de l'interface CLI de l'IdM. Vous pouvez visualiser les membres directs et indirects d'un groupe. Pour plus d'informations, voir Membres directs et indirects d' un groupe.

Procédure :

- Pour dresser la liste des membres d'un groupe, utilisez la commande **ipa group-show *group_name*** pour obtenir la liste des membres d'un groupe. Par exemple :

```
$ ipa group-show group_a
  ...
  Member users: user_a
  Member groups: group_b
  Indirect Member users: user_b
```

NOTE

La liste des membres indirects ne comprend pas les utilisateurs externes des domaines Active Directory de confiance. Les objets utilisateurs de confiance Active Directory ne sont pas visibles dans l'interface de gestion des identités car ils n'existent pas en tant qu'objets LDAP dans la gestion des identités.

19.10. SUPPRESSION D'UN MEMBRE D'UN GROUPE D'UTILISATEURS À L'AIDE DE LA CLI IDM

Cette section décrit comment supprimer un membre d'un groupe d'utilisateurs à l'aide de l'interface CLI de l'IdM.

Conditions préalables

- Vous devez être connecté en tant qu'administrateur. Pour plus de détails, voir Utilisation de kinit pour se connecter manuellement à l'IdM.

Procédure

1. *Optional.* Utilisez la commande **ipa group-show** pour confirmer que le groupe inclut le membre que vous souhaitez supprimer.

2. Supprimer un membre d'un groupe d'utilisateurs à l'aide de la commande **ipa group-remove-member**.
   Spécifiez les membres à supprimer à l'aide de ces options :

   - **--users** supprime un utilisateur IdM
   - **--external** supprime un utilisateur qui existe en dehors du domaine IdM, sous le format **DOMAIN\user_name** ou **user_name@domain**
   - **--groups** supprime un groupe d'utilisateurs IdM

   Par exemple, pour supprimer *user1*, *user2* et *group1* d'un groupe appelé *group_name*:

```
$ ipa group-remove-member group_name --users=user1 --users=user2 --groups=group1
```

19.11. SUPPRESSION D'UTILISATEURS OU DE GROUPES EN TANT QUE GESTIONNAIRES MEMBRES D'UN GROUPE D'UTILISATEURS IDM À L'AIDE DE LA CLI IDM

Cette section décrit comment supprimer des utilisateurs ou des groupes en tant que gestionnaires membres d'un groupe d'utilisateurs IdM à l'aide de la CLI IdM. Les gestionnaires membres peuvent supprimer des utilisateurs ou des groupes de groupes d'utilisateurs IdM, mais ne peuvent pas modifier les attributs d'un groupe.

Conditions préalables

- Vous devez être connecté en tant qu'administrateur. Pour plus de détails, voir Utilisation de kinit pour se connecter manuellement à l'IdM.
- Vous devez disposer du nom de l'utilisateur ou du groupe existant que vous supprimez et du nom du groupe qu'il gère.

Procédure

- Supprimer un utilisateur en tant que gestionnaire membre d'un groupe d'utilisateurs IdM à l'aide de la commande **ipa group-remove-member-manager**.
  Par exemple, pour supprimer l'utilisateur **test** en tant que gestionnaire membre de **group_a**:

```
$ ipa group-remove-member-manager group_a --users=test
Group name: group_a
GID: 1133400009
Membership managed by groups: group_admins
-------------------------
Number of members removed 1
-------------------------
```

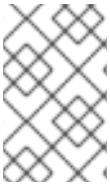
  L'utilisateur **test** ne peut plus gérer les membres de **group_a**.

- Supprimer un groupe en tant que gestionnaire membre d'un groupe d'utilisateurs IdM à l'aide de la commande **ipa group-remove-member-manager**.

Par exemple, pour supprimer le groupe **group_admins** en tant que gestionnaire membre de **group_a**:

```
$ ipa group-remove-member-manager group_a --groups=group_admins
Group name: group_a
GID: 1133400009
----------------------------
Number of members removed 1
----------------------------
```

Le groupe **group_admins** ne peut plus gérer les membres de **group_a**.



NOTE

Après avoir supprimé un gestionnaire membre d'un groupe d'utilisateurs, la mise à jour peut prendre un certain temps avant de se propager à tous les clients de votre environnement de gestion des identités.

Verification steps

- Utiliser la commande **ipa group-show** pour vérifier que l'utilisateur et le groupe ont été supprimés en tant que gestionnaires membres.

```
$ ipa group-show group_a
Group name: group_a
GID: 1133400009
```

Ressources supplémentaires

- Voir **ipa group-remove-member-manager --help** pour plus de détails.

CHAPITRE 20. GESTION DES GROUPES D'UTILISATEURS DANS L'INTERFACE WEB IDM

Ce chapitre présente la gestion des groupes d'utilisateurs à l'aide de l'interface web IdM.

Un groupe d'utilisateurs est un ensemble d'utilisateurs ayant des privilèges, des politiques de mot de passe et d'autres caractéristiques communes.

Dans le cadre de la gestion des identités (IdM), un groupe d'utilisateurs peut inclure :

- Utilisateurs de l'IdM
- d'autres groupes d'utilisateurs de l'IdM
- les utilisateurs externes, c'est-à-dire les utilisateurs qui existent en dehors de l'IdM

20.1. LES DIFFÉRENTS TYPES DE GROUPES DANS L'IDM

IdM prend en charge les types de groupes suivants :

Groupes POSIX (par défaut)

Les groupes POSIX prennent en charge les attributs Linux POSIX pour leurs membres. Notez que les groupes qui interagissent avec Active Directory ne peuvent pas utiliser les attributs POSIX.
Les attributs POSIX identifient les utilisateurs en tant qu'entités distinctes. Parmi les exemples d'attributs POSIX concernant les utilisateurs, on peut citer **uidNumber**, un numéro d'utilisateur (UID), et **gidNumber**, un numéro de groupe (GID).

Groupes non-POSIX

Les groupes non-POSIX ne prennent pas en charge les attributs POSIX. Par exemple, ces groupes n'ont pas de GID défini.
Tous les membres de ce type de groupe doivent appartenir au domaine IdM.

Groupes externes

Utilisez les groupes externes pour ajouter des membres de groupes qui existent dans un magasin d'identité en dehors du domaine IdM, par exemple :

- Un système local
- Un domaine Active Directory
- Un service d'annuaire

Les groupes externes ne prennent pas en charge les attributs POSIX. Par exemple, ces groupes n'ont pas de GID défini.

Tableau 20.1. Groupes d'utilisateurs créés par défaut

Nom du groupe	Membres du groupe par défaut
ipausers	Tous les utilisateurs de l'IdM

Nom du groupe	Membres du groupe par défaut
admins	Utilisateurs disposant de privilèges administratifs, y compris l'utilisateur par défaut **admin**
editors	Il s'agit d'un groupe ancien qui ne bénéficie plus de privilèges particuliers
trust admins	Utilisateurs ayant des privilèges pour gérer les trusts Active Directory

Lorsque vous ajoutez un utilisateur à un groupe d'utilisateurs, celui-ci bénéficie des privilèges et des règles associés au groupe. Par exemple, pour accorder des privilèges administratifs à un utilisateur, ajoutez-le au groupe **admins**.

AVERTISSEMENT

Ne pas supprimer le groupe **admins**. Comme **admins** est un groupe prédéfini requis par IdM, cette opération pose des problèmes avec certaines commandes.

En outre, IdM crée *user private groups* par défaut chaque fois qu'un nouvel utilisateur est créé dans IdM. Pour plus d'informations sur les groupes privés, voir Ajouter des utilisateurs sans groupe privé.

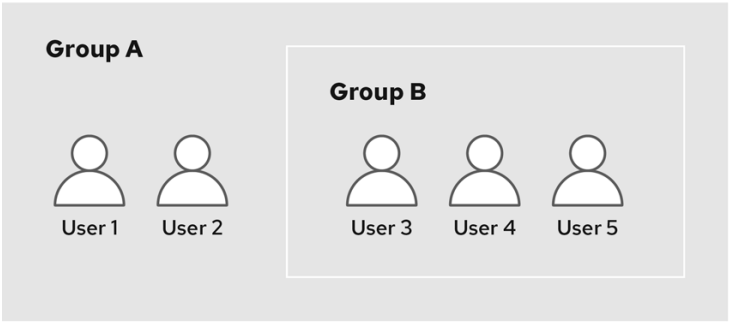
20.2. MEMBRES DIRECTS ET INDIRECTS DU GROUPE

Les attributs des groupes d'utilisateurs dans l'IdM s'appliquent à la fois aux membres directs et indirects : lorsque le groupe B est membre du groupe A, tous les utilisateurs du groupe B sont considérés comme des membres indirects du groupe A.

Par exemple, dans le diagramme suivant :

- L'utilisateur 1 et l'utilisateur 2 sont *direct members* du groupe A.
- L'utilisateur 3, l'utilisateur 4 et l'utilisateur 5 sont *indirect members* du groupe A.

Figure 20.1. Appartenance directe et indirecte à un groupe



Si vous définissez une politique de mot de passe pour le groupe d'utilisateurs A, cette politique s'applique également à tous les utilisateurs du groupe d'utilisateurs B.

20.3. AJOUT D'UN GROUPE D'UTILISATEURS À L'AIDE DE L'INTERFACE WEB IDM

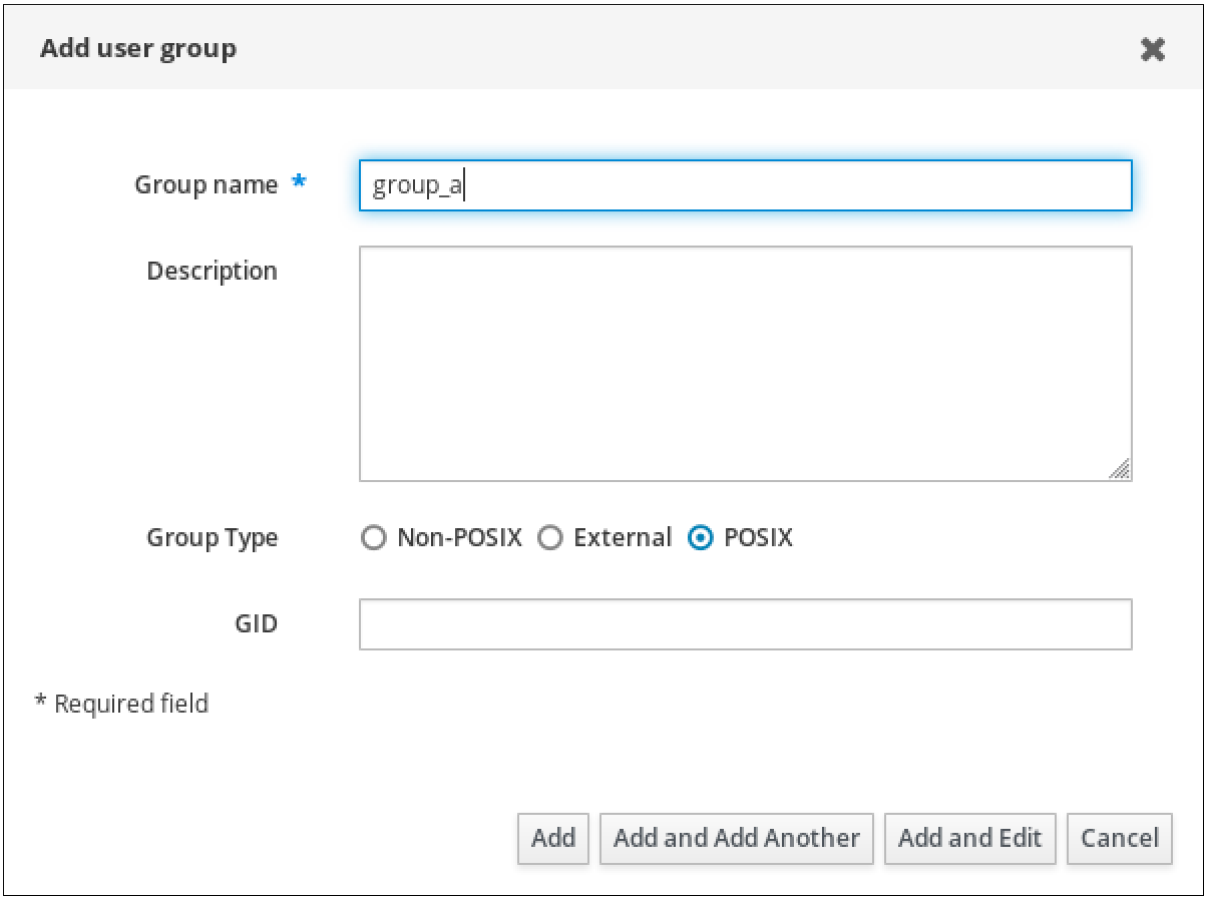
Cette section décrit comment ajouter un groupe d'utilisateurs à l'aide de l'interface Web IdM.

Conditions préalables

- Vous êtes connecté à l'interface Web IdM.

Procédure

1. Cliquez sur **Identity → Groups**, et sélectionnez **User Groups** dans la barre latérale gauche.
2. Cliquez sur **Add** pour commencer à ajouter le groupe.
3. Complétez les informations sur le groupe. Pour plus d'informations sur les types de groupes d'utilisateurs, voir Les différents types de groupes dans IdM .
   Vous pouvez spécifier un GID personnalisé pour le groupe. Dans ce cas, veillez à éviter les conflits d'identifiants. Si vous ne spécifiez pas de GID personnalisé, IdM attribue automatiquement un GID à partir de la plage d'ID disponible.



4. Cliquez sur **Add** pour confirmer.

20.4. SUPPRESSION D'UN GROUPE D'UTILISATEURS À L'AIDE DE L'INTERFACE WEB IDM

Cette section décrit comment supprimer un groupe d'utilisateurs à l'aide de l'interface Web de l'IdM. Notez que la suppression d'un groupe ne supprime pas les membres du groupe de l'IdM.

Conditions préalables

- Vous êtes connecté à l'interface Web IdM.

Procédure

1. Cliquez sur **Identity → Groups** et sélectionnez **User Groups**.
2. Sélectionnez le groupe à supprimer.
3. Cliquez sur **Delete**.
4. Cliquez sur **Delete** pour confirmer.

20.5. AJOUTER UN MEMBRE À UN GROUPE D'UTILISATEURS À L'AIDE DE L'INTERFACE WEB IDM

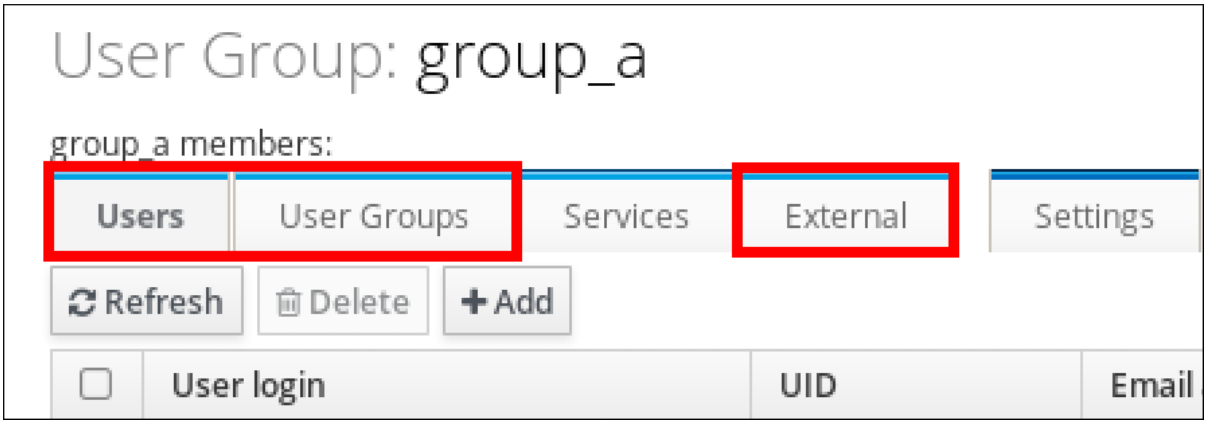
Vous pouvez ajouter des utilisateurs et des groupes d'utilisateurs en tant que membres d'un groupe d'utilisateurs. Pour plus d'informations, voir Les différents types de groupes dans IdM et Les membres directs et indirects d'un groupe.

Conditions préalables

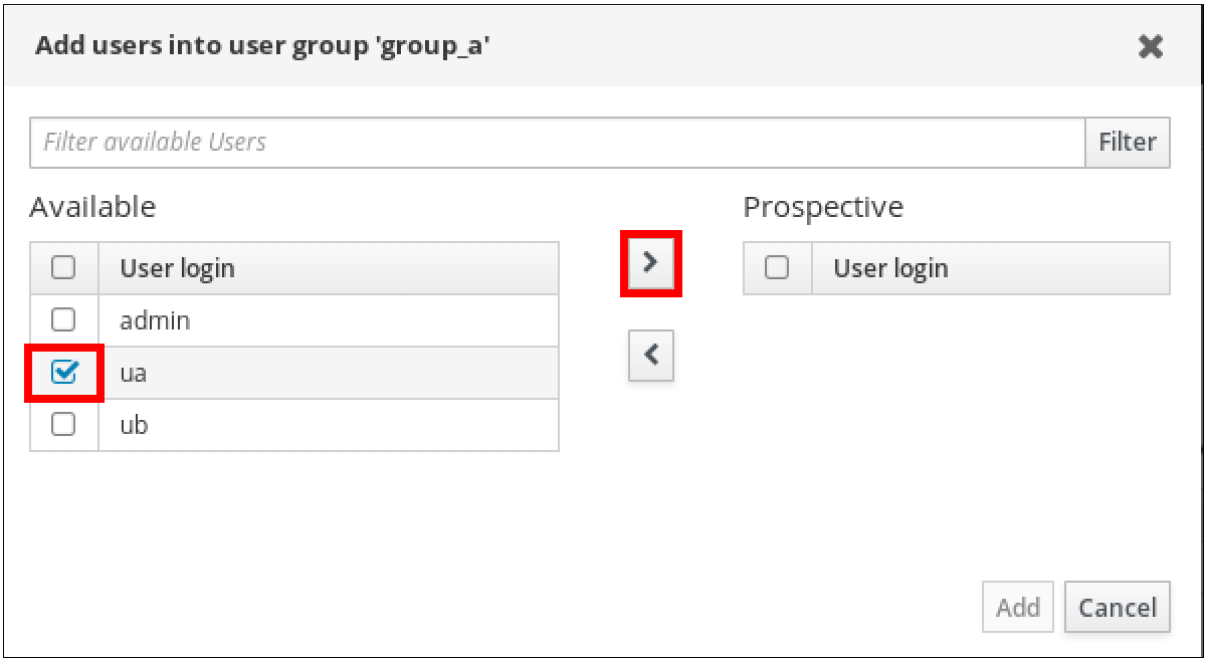
- Vous êtes connecté à l'interface Web IdM.

Procédure

1. Cliquez sur **Identity → Groups** et sélectionnez **User Groups** dans la barre latérale gauche.
2. Cliquez sur le nom du groupe.
3. Sélectionnez le type de membre du groupe que vous souhaitez ajouter : **Users, User Groups,** ou **External**.



4. Cliquez sur **Add**.
5. Cochez la case à côté d'un ou plusieurs membres que vous souhaitez ajouter.
6. Cliquez sur la flèche vers la droite pour déplacer les membres sélectionnés vers le groupe.



7. Cliquez sur **Add** pour confirmer.

20.6. AJOUT D'UTILISATEURS OU DE GROUPES EN TANT QUE GESTIONNAIRES MEMBRES D'UN GROUPE D'UTILISATEURS IDM À L'AIDE DE L'INTERFACE WEB

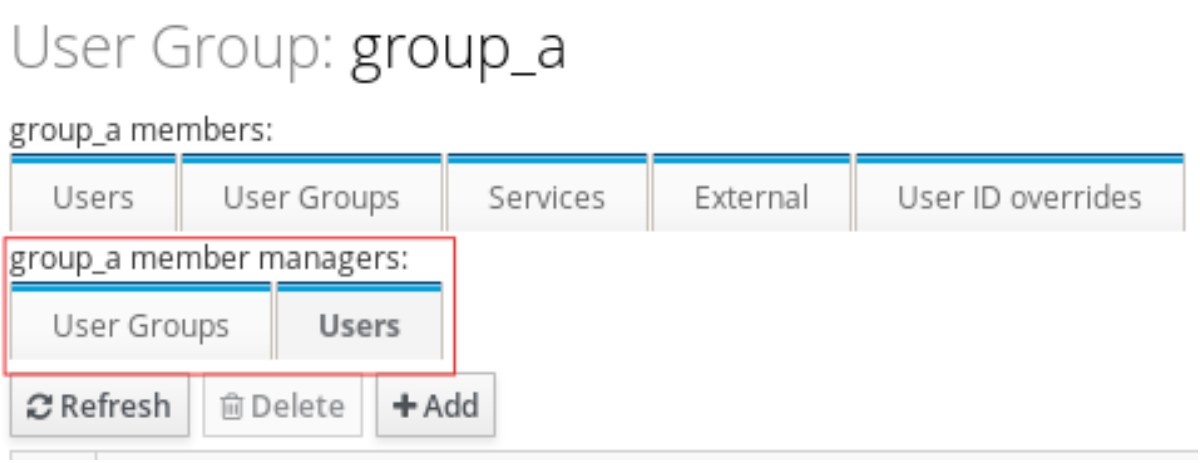
Cette section décrit comment ajouter des utilisateurs ou des groupes en tant que gestionnaires membres d'un groupe d'utilisateurs IdM à l'aide de l'interface Web. Les gestionnaires membres peuvent ajouter des utilisateurs ou des groupes aux groupes d'utilisateurs IdM, mais ne peuvent pas modifier les attributs d'un groupe.

Conditions préalables

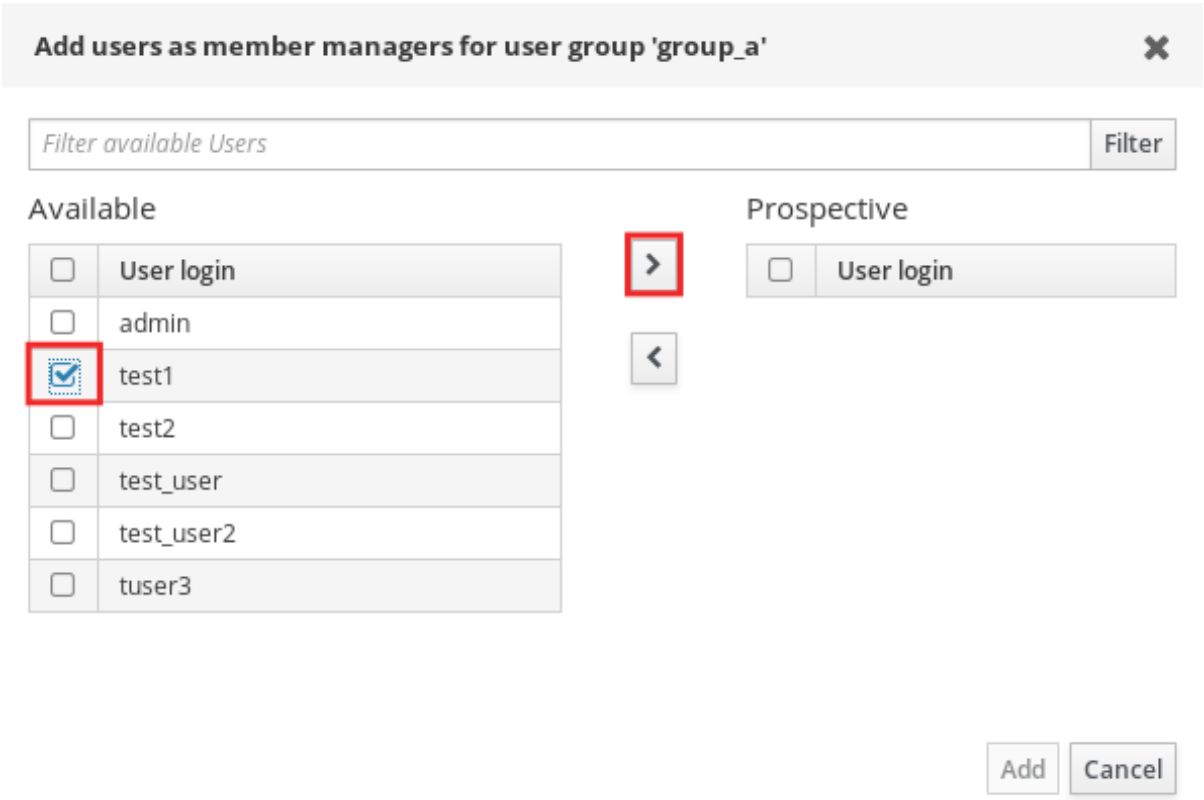
- Vous êtes connecté à l'interface Web IdM.
- Vous devez disposer du nom de l'utilisateur ou du groupe que vous ajoutez en tant que membres gestionnaires et du nom du groupe que vous souhaitez qu'ils gèrent.

Procédure

1. Cliquez sur **Identity → Groups** et sélectionnez **User Groups** dans la barre latérale gauche.
2. Cliquez sur le nom du groupe.
3. Sélectionnez le type de gestionnaire de groupe que vous souhaitez ajouter : **Users** ou **User Groups**.



4. Cliquez sur **Add**.
5. Cochez la case à côté d'un ou plusieurs membres que vous souhaitez ajouter.
6. Cliquez sur la flèche vers la droite pour déplacer les membres sélectionnés vers le groupe.



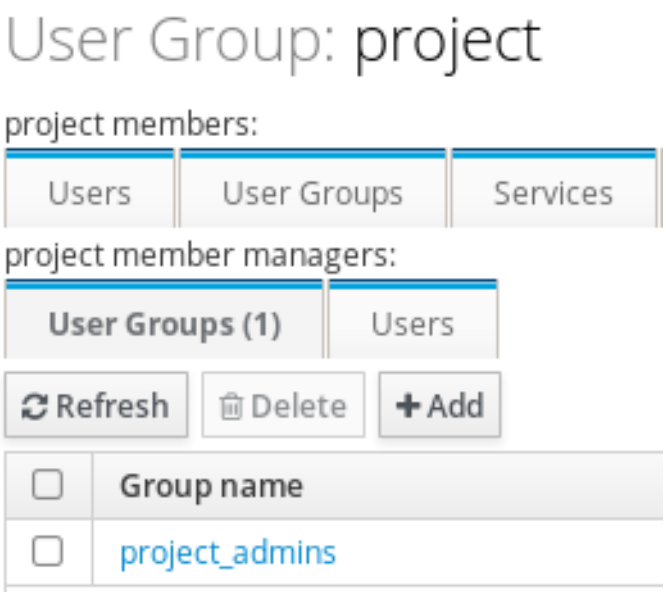
7. Cliquez sur **Add** pour confirmer.

NOTE

Après avoir ajouté un gestionnaire membre à un groupe d'utilisateurs, la mise à jour peut prendre un certain temps avant de se propager à tous les clients de votre environnement de gestion des identités.

Verification steps

- Vérifiez que l'utilisateur ou le groupe d'utilisateurs nouvellement ajouté a été ajouté à la liste des utilisateurs ou des groupes d'utilisateurs du gestionnaire des membres :



Ressources supplémentaires

- Voir **ipa group-add-member-manager --help** pour plus d'informations.

20.7. VISUALISATION DES MEMBRES D'UN GROUPE À L'AIDE DE L'INTERFACE WEB IDM

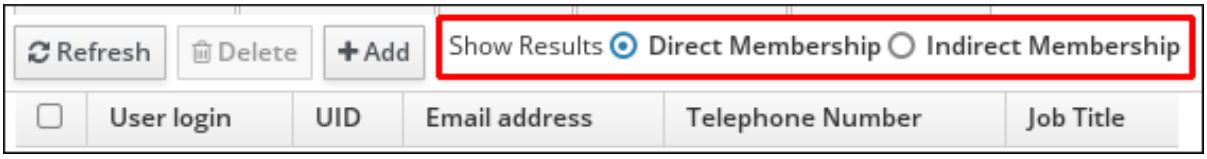
Cette section décrit comment visualiser les membres d'un groupe à l'aide de l'interface Web IdM. Vous pouvez afficher les membres directs et indirects d'un groupe. Pour plus d'informations, voir Membres directs et indirects d'un groupe.

Conditions préalables

- Vous êtes connecté à l'interface Web IdM.

Procédure

1. Sélectionnez **Identity → Groups**.
2. Sélectionnez **User Groups** dans la barre latérale gauche.
3. Cliquez sur le nom du groupe que vous souhaitez consulter.
4. Passer de **Direct Membership** à **Indirect Membership**.



20.8. SUPPRESSION D'UN MEMBRE D'UN GROUPE D'UTILISATEURS À L'AIDE DE L'INTERFACE WEB IDM

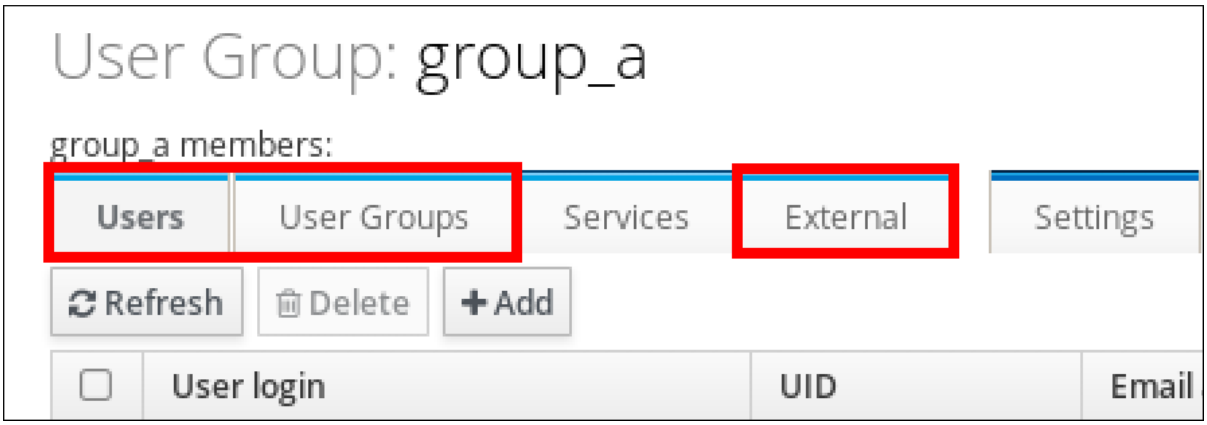
Cette section décrit comment supprimer un membre d'un groupe d'utilisateurs à l'aide de l'interface Web IdM.

Conditions préalables

- Vous êtes connecté à l'interface Web IdM.

Procédure

1. Cliquez sur **Identity → Groups** et sélectionnez **User Groups** dans la barre latérale gauche.
2. Cliquez sur le nom du groupe.
3. Sélectionnez le type de membre du groupe que vous souhaitez supprimer : **Users, User Groups**, ou **External**.



4. Cochez la case en regard du membre que vous souhaitez supprimer.
5. Cliquez sur **Delete**.
6. Cliquez sur **Delete** pour confirmer.

20.9. SUPPRESSION D'UTILISATEURS OU DE GROUPES EN TANT QUE GESTIONNAIRES MEMBRES D'UN GROUPE D'UTILISATEURS IDM À L'AIDE DE L'INTERFACE WEB

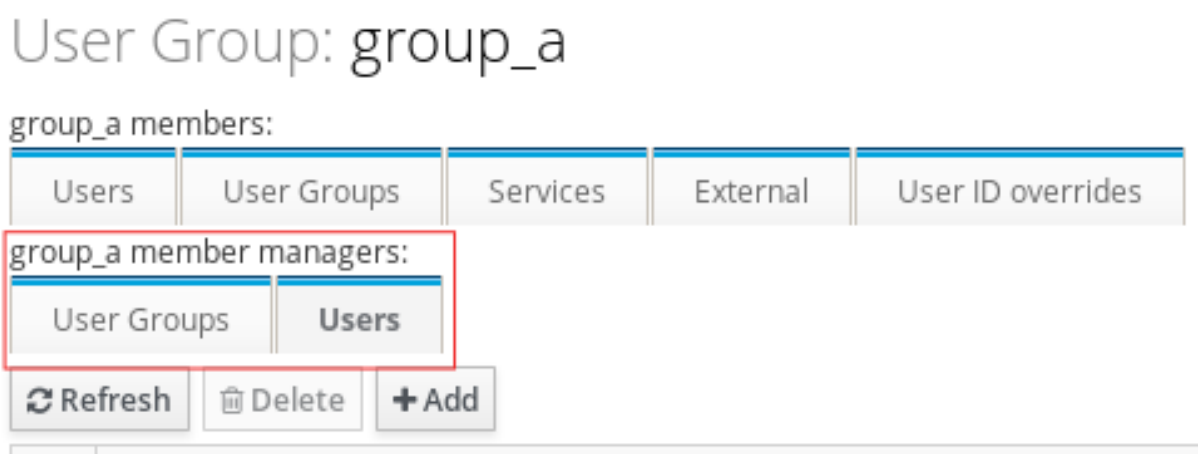
Cette section décrit comment supprimer des utilisateurs ou des groupes en tant que gestionnaires membres d'un groupe d'utilisateurs IdM à l'aide de l'interface Web. Les gestionnaires membres peuvent supprimer des utilisateurs ou des groupes de groupes d'utilisateurs IdM, mais ne peuvent pas modifier les attributs d'un groupe.

Conditions préalables

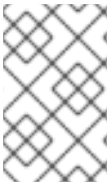
- Vous êtes connecté à l'interface Web IdM.
- Vous devez disposer du nom de l'utilisateur ou du groupe existant que vous supprimez et du nom du groupe qu'il gère.

Procédure

1. Cliquez sur **Identity → Groups** et sélectionnez **User Groups** dans la barre latérale gauche.
2. Cliquez sur le nom du groupe.
3. Sélectionnez le type de gestionnaire de membres que vous souhaitez supprimer : **Users** ou **User Groups**.



4. Cochez la case en regard du gestionnaire de membres que vous souhaitez supprimer.
5. Cliquez sur **Delete**.
6. Cliquez sur **Delete** pour confirmer.

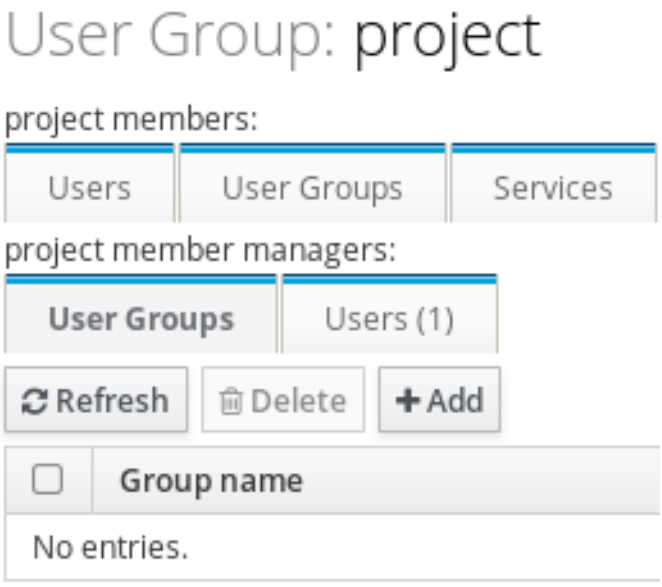


NOTE

Après avoir supprimé un gestionnaire membre d'un groupe d'utilisateurs, la mise à jour peut prendre un certain temps avant de se propager à tous les clients de votre environnement de gestion des identités.

Verification steps

- Vérifiez que l'utilisateur ou le groupe d'utilisateurs a été supprimé de la liste des utilisateurs ou des groupes d'utilisateurs du gestionnaire des membres :



Ressources supplémentaires

- Voir **ipa group-add-member-manager --help** pour plus de détails.

CHAPITRE 21. GÉRER LES GROUPES D'UTILISATEURS À L'AIDE DE PLAYBOOKS ANSIBLE

Cette section présente la gestion des groupes d'utilisateurs à l'aide des playbooks Ansible.

Un groupe d'utilisateurs est un ensemble d'utilisateurs ayant des privilèges, des politiques de mot de passe et d'autres caractéristiques communes.

Dans le cadre de la gestion des identités (IdM), un groupe d'utilisateurs peut inclure :

- Utilisateurs de l'IdM
- d'autres groupes d'utilisateurs de l'IdM
- les utilisateurs externes, c'est-à-dire les utilisateurs qui existent en dehors de l'IdM

La section comprend les sujets suivants :

- Les différents types de groupes dans l'IdM
- Membres directs et indirects du groupe
- Assurer la présence de groupes IdM et de membres de groupes à l'aide de playbooks Ansible
- Utiliser Ansible pour permettre aux utilisateurs AD d'administrer IdM
- Assurer la présence de gestionnaires de membres dans les groupes d'utilisateurs IDM à l'aide de playbooks Ansible
- Garantir l'absence de gestionnaires de membres dans les groupes d'utilisateurs IDM à l'aide de playbooks Ansible

21.1. LES DIFFÉRENTS TYPES DE GROUPES DANS L'IDM

IdM prend en charge les types de groupes suivants :

Groupes POSIX (par défaut)

Les groupes POSIX prennent en charge les attributs Linux POSIX pour leurs membres. Notez que les groupes qui interagissent avec Active Directory ne peuvent pas utiliser les attributs POSIX.
Les attributs POSIX identifient les utilisateurs en tant qu'entités distinctes. Parmi les exemples d'attributs POSIX concernant les utilisateurs, on peut citer **uidNumber**, un numéro d'utilisateur (UID), et **gidNumber**, un numéro de groupe (GID).

Groupes non-POSIX

Les groupes non-POSIX ne prennent pas en charge les attributs POSIX. Par exemple, ces groupes n'ont pas de GID défini.
Tous les membres de ce type de groupe doivent appartenir au domaine IdM.

Groupes externes

Utilisez les groupes externes pour ajouter des membres de groupes qui existent dans un magasin d'identité en dehors du domaine IdM, par exemple :

- Un système local

- Un domaine Active Directory
- Un service d'annuaire

Les groupes externes ne prennent pas en charge les attributs POSIX. Par exemple, ces groupes n'ont pas de GID défini.

Tableau 21.1. Groupes d'utilisateurs créés par défaut

Nom du groupe	Membres du groupe par défaut
ipausers	Tous les utilisateurs de l'IdM
admins	Utilisateurs disposant de privilèges administratifs, y compris l'utilisateur par défaut **admin**
editors	Il s'agit d'un groupe ancien qui ne bénéficie plus de privilèges particuliers
trust admins	Utilisateurs ayant des privilèges pour gérer les trusts Active Directory

Lorsque vous ajoutez un utilisateur à un groupe d'utilisateurs, celui-ci bénéficie des privilèges et des règles associés au groupe. Par exemple, pour accorder des privilèges administratifs à un utilisateur, ajoutez-le au groupe **admins**.

AVERTISSEMENT

Ne pas supprimer le groupe **admins**. Comme **admins** est un groupe prédéfini requis par IdM, cette opération pose des problèmes avec certaines commandes.

En outre, IdM crée *user private groups* par défaut chaque fois qu'un nouvel utilisateur est créé dans IdM. Pour plus d'informations sur les groupes privés, voir Ajouter des utilisateurs sans groupe privé.

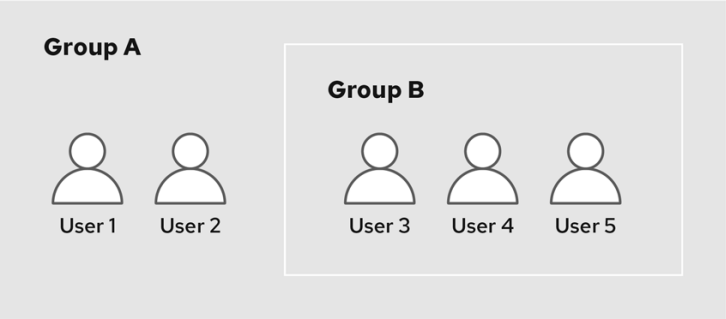
21.2. MEMBRES DIRECTS ET INDIRECTS DU GROUPE

Les attributs des groupes d'utilisateurs dans l'IdM s'appliquent à la fois aux membres directs et indirects : lorsque le groupe B est membre du groupe A, tous les utilisateurs du groupe B sont considérés comme des membres indirects du groupe A.

Par exemple, dans le diagramme suivant :

- L'utilisateur 1 et l'utilisateur 2 sont *direct members* du groupe A.
- L'utilisateur 3, l'utilisateur 4 et l'utilisateur 5 sont *indirect members* du groupe A.

Figure 21.1. Appartenance directe et indirecte à un groupe



Si vous définissez une politique de mot de passe pour le groupe d'utilisateurs A, cette politique s'applique également à tous les utilisateurs du groupe d'utilisateurs B.

21.3. ASSURER LA PRÉSENCE DE GROUPES IDM ET DE MEMBRES DE GROUPES À L'AIDE DE PLAYBOOKS ANSIBLE

La procédure suivante décrit comment assurer la présence de groupes IdM et de membres de groupes - à la fois des utilisateurs et des groupes d'utilisateurs - à l'aide d'un playbook Ansible.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.
- Les utilisateurs que vous souhaitez référencer dans votre manuel de jeu Ansible existent dans IdM. Pour plus de détails sur la façon de garantir la présence des utilisateurs à l'aide d'Ansible, voir Gérer les comptes d'utilisateurs à l'aide de carnets de commande Ansible .

Procédure

1. Créez un fichier d'inventaire, par exemple **inventory.file**, et définissez-y **ipaserver**:

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

2. Créez un fichier playbook Ansible avec les informations nécessaires sur l'utilisateur et le groupe :

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

```
vars_files:
- /home/user_name/MyPlaybooks/secret.yml
tasks:
- name: Create group ops with gid 1234
  ipagroup:
    ipaadmin_password: "{{ ipaadmin_password }}"
    name: ops
    gidnumber: 1234

- name: Create group sysops
  ipagroup:
    ipaadmin_password: "{{ ipaadmin_password }}"
    name: sysops
    user:
    - idm_user

- name: Create group appops
  ipagroup:
    ipaadmin_password: "{{ ipaadmin_password }}"
    name: appops

- name: Add group members sysops and appops to group ops
  ipagroup:
    ipaadmin_password: "{{ ipaadmin_password }}"
    name: ops
    group:
    - sysops
    - appops
```

3. Exécutez le manuel de jeu :

```
$ ansible-playbook --vault-password-file=password_file -v -i
path_to_inventory_directory/inventory.file path_to_playbooks_directory/add-group-
members.yml
```

Verification steps

Vous pouvez vérifier si le groupe **ops** contient **sysops** et **appops** en tant que membres directs et **idm_user** en tant que membre indirect en utilisant la commande **ipa group-show**:

1. Connectez-vous à **ipaserver** en tant qu'administrateur :

```
$ ssh admin@server.idm.example.com
Password:
[admin@server /]$
```

2. Afficher des informations sur *ops*:

```
ipaserver]$ ipa group-show ops
  Group name: ops
  GID: 1234
  Member groups: sysops, appops
  Indirect Member users: idm_user
```

Les groupes **appops** et **sysops** - ce dernier comprenant l'utilisateur **idm_user** - existent dans IdM.

Ressources supplémentaires

- Voir le fichier Markdown de **/usr/share/doc/ansible-freeipa/README-group.md**.

21.4. UTILISER ANSIBLE POUR PERMETTRE AUX UTILISATEURS AD D'ADMINISTRER IDM

Cette section décrit comment utiliser un playbook Ansible pour s'assurer qu'un remplacement d'ID d'utilisateur est présent dans un groupe de gestion des identités (IdM). Le remplacement de l'ID utilisateur est le remplacement d'un utilisateur Active Directory (AD) que vous avez créé dans la vue de confiance par défaut après avoir établi une confiance avec AD. Suite à l'exécution du playbook, un utilisateur AD, par exemple un administrateur AD, est en mesure d'administrer entièrement l'IdM sans avoir deux comptes et mots de passe différents.

Conditions préalables

- Vous connaissez le mot de passe de l'IdM **admin**.
- Vous avez installé un trust avec AD.
- Le remplacement de l'ID de l'utilisateur AD existe déjà dans l'IdM. Si ce n'est pas le cas, créez-le avec la commande **ipa idoverrideuser-add 'default trust view'** ***ad_user@ad.example.com*** commande.
- Le groupe auquel vous ajoutez le remplacement de l'ID utilisateur existe déjà dans IdM .
- Vous utilisez la version 4.8.7 d'IdM ou une version ultérieure. Pour connaître la version d'IdM installée sur votre serveur, entrez **ipa --version**.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

Procédure

1. Naviguez jusqu'à votre répertoire ***~/MyPlaybooks/*** répertoire :

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

2. Créez un playbook **add-useridoverride-to-group.yml** avec le contenu suivant :

```
---
- name: Playbook to ensure presence of users in a group
```

```
  hosts: ipaserver


  - name: Ensure the ad_user@ad.example.com user ID override is a member of the admins
group:
    ipagroup:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: admins
      idoverrideuser:
      - ad_user@ad.example.com
```

Dans l'exemple :

- Secret123 est le mot de passe de l'IdM **admin**.
- **admins** est le nom du groupe POSIX IdM auquel vous ajoutez l'annulation de l'ID **ad_user@ad.example.com**. Les membres de ce groupe disposent de tous les privilèges d'administrateur.
- **ad_user@ad.example.com** est le remplacement de l'ID utilisateur d'un administrateur AD. L'utilisateur est stocké dans le domaine AD avec lequel une confiance a été établie.

3. Enregistrer le fichier.
4. Exécutez le playbook Ansible. Spécifiez le fichier du livre de jeu, le fichier contenant le mot de passe protégeant le fichier **secret.yml** et le fichier d'inventaire :

```
$ ansible-playbook --vault-password-file=password_file -v -i inventory add-
useridoverride-to-group.yml
```

Ressources supplémentaires

- Remplacement des ID pour les utilisateurs AD
- /usr/share/doc/ansible-freeipa/README-group.md
- /usr/share/doc/ansible-freeipa/playbooks/user
- Utilisation des vues d'identification dans les environnements Active Directory
- Permettre aux utilisateurs AD d'administrer l'IdM

21.5. ASSURER LA PRÉSENCE DE GESTIONNAIRES MEMBRES DANS LES GROUPES D'UTILISATEURS IDM À L'AIDE DE PLAYBOOKS ANSIBLE

La procédure suivante décrit comment assurer la présence des gestionnaires membres de l'IdM - à la fois les utilisateurs et les groupes d'utilisateurs - à l'aide d'un playbook Ansible.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :

- Vous utilisez la version 2.8 ou ultérieure d'Ansible.
- Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
- L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
- L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

- Vous devez disposer du nom de l'utilisateur ou du groupe que vous ajoutez en tant que membres gestionnaires et du nom du groupe que vous souhaitez qu'ils gèrent.

Procédure

1. Créez un fichier d'inventaire, par exemple **inventory.file**, et définissez-y **ipaserver**:

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

2. Créer un fichier playbook Ansible avec les informations nécessaires à la gestion des utilisateurs et des membres du groupe :

```
---
- name: Playbook to handle membership management
  hosts: ipaserver

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Ensure user test is present for group_a
    ipagroup:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: group_a
      membermanager_user: test

  - name: Ensure group_admins is present for group_a
    ipagroup:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: group_a
      membermanager_group: group_admins
```

3. Exécutez le manuel de jeu :

```
$ ansible-playbook --vault-password-file=password_file -v -i
path_to_inventory_directory/inventory.file path_to_playbooks_directory/add-member-
managers-user-groups.yml
```

Verification steps

Vous pouvez vérifier si le groupe **group_a** contient **test** en tant que gestionnaire membre et si **group_admins** est un gestionnaire membre de **group_a** en utilisant la commande **ipa group-show**:

1. Connectez-vous à **ipaserver** en tant qu'administrateur :

```
$ ssh admin@server.idm.example.com
Password:
[admin@server /]$
```

2. Afficher des informations sur *managergroup1*:

```
ipaserver]$ ipa group-show group_a
  Group name: group_a
  GID: 1133400009
  Membership managed by groups: group_admins
  Membership managed by users: test
```

Ressources supplémentaires

- Voir **ipa host-add-member-manager --help**.
- Voir la page de manuel **ipa**.

21.6. ASSURER L'ABSENCE DE MEMBRES GESTIONNAIRES DANS LES GROUPES D'UTILISATEURS IDM À L'AIDE DE PLAYBOOKS ANSIBLE

La procédure suivante décrit comment garantir l'absence de gestionnaires membres de l'IdM - à la fois des utilisateurs et des groupes d'utilisateurs - à l'aide d'un playbook Ansible.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.
- Vous devez disposer du nom de l'utilisateur ou du groupe existant que vous supprimez et du nom du groupe qu'il gère.

Procédure

1. Créez un fichier d'inventaire, par exemple **inventory.file**, et définissez-y **ipaserver**:

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

2. Créer un fichier playbook Ansible avec les informations nécessaires à la gestion des utilisateurs et des membres du groupe :

```
---
- name: Playbook to handle membership management
  hosts: ipaserver

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Ensure member manager user and group members are absent for group_a
    ipagroup:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: group_a
      membermanager_user: test
      membermanager_group: group_admins
      action: member
      state: absent
```

3. Exécutez le manuel de jeu :

```
$ ansible-playbook --vault-password-file=password_file -v -i
path_to_inventory_directory/inventory.file path_to_playbooks_directory/ensure-
member-managers-are-absent.yml
```

Verification steps

Vous pouvez vérifier que le groupe **group_a** ne contient pas **test** en tant que gestionnaire membre et **group_admins** en tant que gestionnaire membre de **group_a** en utilisant la commande **ipa group-show**:

1. Connectez-vous à **ipaserver** en tant qu'administrateur :

```
$ ssh admin@server.idm.example.com
Password:
[admin@server /]$
```

2. Afficher des informations sur le groupe_a :

```
ipaserver]$ ipa group-show group_a
  Group name: group_a
  GID: 1133400009
```

Ressources supplémentaires

- Voir **ipa host-remove-member-manager --help**.
- Voir la page de manuel **ipa**.

CHAPITRE 22. AUTOMATISATION DE L'APPARTENANCE À UN GROUPE À L'AIDE DE LA CLI IDM

L'utilisation de l'appartenance automatique à un groupe vous permet d'affecter automatiquement des utilisateurs et des hôtes à des groupes en fonction de leurs attributs. Par exemple, vous pouvez

- Répartissez les entrées utilisateur des employés dans des groupes en fonction du responsable de l'employé, de son lieu de travail ou de tout autre attribut.
- Divisez les hôtes en fonction de leur classe, de leur lieu de résidence ou de tout autre attribut.
- Ajouter tous les utilisateurs ou tous les hôtes à un seul groupe global.

Ce chapitre couvre les sujets suivants :

22.1. AVANTAGES DE L'ADHÉSION AUTOMATIQUE À UN GROUPE

L'utilisation de l'adhésion automatique pour les utilisateurs vous permet de

- **Reduce the overhead of manually managing group memberships**
  Il n'est plus nécessaire d'affecter manuellement chaque utilisateur et chaque hôte à des groupes.
- **Improve consistency in user and host management**
  Les utilisateurs et les hôtes sont affectés à des groupes sur la base de critères strictement définis et évalués automatiquement.
- **Simplify the management of group-based settings**
  Divers paramètres sont définis pour les groupes, puis appliqués aux membres individuels du groupe, par exemple les règles **sudo**, l'automount ou le contrôle d'accès. L'ajout automatique d'utilisateurs et d'hôtes à des groupes facilite la gestion de ces paramètres.

22.2. RÈGLES DE L'AUTOMEMBER

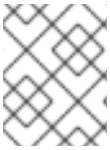
Lors de la configuration de l'appartenance automatique à un groupe, l'administrateur définit des règles d'appartenance automatique. Une règle automember s'applique à un groupe cible d'utilisateurs ou d'hôtes spécifique. Elle ne peut pas s'appliquer à plusieurs groupes à la fois.

Après avoir créé une règle, l'administrateur y ajoute des conditions. Celles-ci précisent quels utilisateurs ou hôtes sont inclus ou exclus du groupe cible :

- **Inclusive conditions**
  Lorsqu'une entrée utilisateur ou hôte remplit une condition d'inclusion, elle est incluse dans le groupe cible.

- **Exclusive conditions**
  Lorsqu'une entrée utilisateur ou hôte remplit une condition d'exclusivité, elle n'est pas incluse dans le groupe cible.

Les conditions sont spécifiées sous forme d'expressions régulières au format PCRE (Perl-compatible regular expressions). Pour plus d'informations sur PCRE, voir la page de manuel **pcresyntax(3)**.



NOTE

L'IdM évalue les conditions exclusives avant les conditions inclusives. En cas de conflit, les conditions exclusives l'emportent sur les conditions inclusives.

Une règle automember s'applique à toutes les entrées créées à l'avenir. Ces entrées seront automatiquement ajoutées au groupe cible spécifié. Si une entrée remplit les conditions spécifiées dans plusieurs règles automember, elle sera ajoutée à tous les groupes correspondants.

Les entrées existantes sont **not** affectées par la nouvelle règle. Si vous souhaitez modifier des entrées existantes, reportez-vous à la section Application des règles automember aux entrées existantes à l'aide de l'interface CLI de l'IdM.

22.3. AJOUT D'UNE RÈGLE D'APPARTENANCE AUTOMATIQUE À L'AIDE DE LA CLI D'IDM

Cette section décrit l'ajout d'une règle de membre automatique à l'aide de l'interface CLI de l'IdM. Pour plus d'informations sur les règles d'appartenance à un groupe, voir Règles d'appartenance à un groupe .

Après avoir ajouté une règle automember, vous pouvez y ajouter des conditions en suivant la procédure décrite dans la section Ajout d'une condition à une règle automember .

NOTE

Les entrées existantes sont **not** affectées par la nouvelle règle. Si vous souhaitez modifier des entrées existantes, reportez-vous à la section Application des règles automember aux entrées existantes à l'aide de l'interface CLI de l'IdM.

Conditions préalables

- Vous devez être connecté en tant qu'administrateur. Pour plus de détails, voir Utilisation de kinit pour se connecter manuellement à l'IdM.
- Le groupe cible de la nouvelle règle doit exister dans l'IdM.

Procédure

1. Entrez la commande **ipa automember-add** pour ajouter une règle automember.
2. Lorsque vous y êtes invité, précisez :
   - **Automember rule**. Il s'agit du nom du groupe cible.
   - **Grouping Type**. Ceci indique si la règle cible un groupe d'utilisateurs ou un groupe d'hôtes. Pour cibler un groupe d'utilisateurs, entrez **group**. Pour cibler un groupe d'hôtes, entrez **hostgroup**.

   Par exemple, pour ajouter une règle automember pour un groupe d'utilisateurs nommé **user_group**:

```
$ ipa automember-add
Automember Rule: user_group
Grouping Type: group
-------------------------------
Added automember rule "user_group"
-------------------------------
  Automember Rule: user_group
```

Verification steps

- Vous pouvez afficher les règles et les conditions existantes dans IdM en utilisant Visualisation des règles existantes pour les membres automatiques à l'aide de la CLI d'IdM.

22.4. AJOUT D'UNE CONDITION À UNE RÈGLE DE MEMBRE AUTOMATIQUE À L'AIDE DE LA CLI IDM

Cette section décrit comment ajouter une condition à une règle automember à l'aide de l'interface CLI de l'IdM. Pour plus d'informations sur les règles d'appartenance à un groupe, voir Règles d'appartenance à un groupe.

Conditions préalables

- Vous devez être connecté en tant qu'administrateur. Pour plus de détails, voir Utilisation de kinit pour se connecter manuellement à l'IdM.
- La règle cible doit exister dans IdM. Pour plus de détails, voir Ajout d'une règle automember à l'aide de IdM CLI.

Procédure

1. Définissez une ou plusieurs conditions inclusives ou exclusives à l'aide de la commande **ipa automember-add-condition**.
2. Lorsque vous y êtes invité, précisez :
   - **Automember rule**. Il s'agit du nom de la règle cible. Voir Règles Automember pour plus de détails.
   - **Attribute Key**. Ceci spécifie l'attribut d'entrée auquel le filtre s'appliquera. Par exemple, **uid** pour les utilisateurs.

- **Grouping Type**. Ceci indique si la règle cible un groupe d'utilisateurs ou un groupe d'hôtes. Pour cibler un groupe d'utilisateurs, entrez **group**. Pour cibler un groupe d'hôtes, entrez **hostgroup**.

- **Inclusive regex** et **Exclusive regex**. Elles spécifient une ou plusieurs conditions sous forme d'expressions régulières. Si vous ne souhaitez spécifier qu'une seule condition, appuyez sur **Enter** lorsque vous êtes invité à spécifier l'autre condition.

Par exemple, la condition suivante vise tous les utilisateurs ayant une valeur quelconque (.*) dans leur attribut de connexion (**uid**).

```
$ ipa automember-add-condition
Automember Rule: user_group
Attribute Key: uid
Grouping Type: group
[Inclusive Regex]: .*
[Exclusive Regex]:
---------------------------------
Added condition(s) to "user_group"
---------------------------------
  Automember Rule: user_group
  Inclusive Regex: uid=.*
----------------------------
Number of conditions added 1
----------------------------
```

Autre exemple, vous pouvez utiliser une règle d'appartenance automatique pour cibler tous les utilisateurs Windows synchronisés à partir d'Active Directory (AD). **objectClass** Pour ce faire, créez une condition qui cible tous les utilisateurs dont l'attribut **ntUser** est partagé par tous les utilisateurs AD :

```
$ ipa automember-add-condition
Automember Rule: ad_users
Attribute Key: objectclass
Grouping Type: group
[Inclusive Regex]: ntUser
[Exclusive Regex]:
-----------------------------------
Added condition(s) to "ad_users"
-----------------------------------
  Automember Rule: ad_users
  Inclusive Regex: objectclass=ntUser
----------------------------
Number of conditions added 1
----------------------------
```

Verification steps

- Vous pouvez afficher les règles et les conditions existantes dans IdM en utilisant Visualisation des règles existantes pour les membres automatiques à l'aide de la CLI d'IdM.

22.5. VISUALISATION DES RÈGLES EXISTANTES POUR LES MEMBRES AUTOMATIQUES À L'AIDE DE LA CLI IDM

Cette section décrit comment visualiser les règles automember existantes à l'aide de la CLI IdM.

Conditions préalables

- Vous devez être connecté en tant qu'administrateur. Pour plus de détails, voir Utilisation de kinit pour se connecter manuellement à l'IdM.

Procédure

1. Entrez la commande **ipa automember-find**.
2. Lorsque vous y êtes invité, indiquez l'adresse **Grouping type**:
   - Pour cibler un groupe d'utilisateurs, entrez **group**.
   - Pour cibler un groupe d'hôtes, entrez **hostgroup**.
     Par exemple :

```
$ ipa automember-find
Grouping Type: group
---------------
1 rules matched
---------------
  Automember Rule: user_group
  Inclusive Regex: uid=.*
----------------------------
Number of entries returned 1
----------------------------
```

22.6. SUPPRESSION D'UNE RÈGLE AUTOMEMBER À L'AIDE DE LA CLI IDM

Cette section décrit comment supprimer une règle automember à l'aide de l'interface CLI de l'IdM.

La suppression d'une règle membre automatique supprime également toutes les conditions associées à la règle. Pour supprimer uniquement des conditions spécifiques d'une règle, voir Suppression d'une condition d'une règle automember à l'aide de l'interface CLI d'IdM.

Conditions préalables

- Vous devez être connecté en tant qu'administrateur. Pour plus de détails, voir Utilisation de kinit pour se connecter manuellement à l'IdM.

Procédure

1. Entrez la commande **ipa automember-del**.
2. Lorsque vous y êtes invité, précisez :
   - **Automember rule**. Il s'agit de la règle que vous souhaitez supprimer.
   - **Grouping rule**. Indique si la règle à supprimer concerne un groupe d'utilisateurs ou un groupe d'hôtes. Saisissez **group** ou **hostgroup**.

22.7. SUPPRESSION D'UNE CONDITION D'UNE RÈGLE DE MEMBRE AUTOMATIQUE À L'AIDE DE L'INTERFACE CLI DE L'IDM

Cette section explique comment supprimer une condition spécifique d'une règle automember.

Conditions préalables

- Vous devez être connecté en tant qu'administrateur. Pour plus de détails, voir Utilisation de kinit pour se connecter manuellement à l'IdM.

Procédure

1. Entrez la commande **ipa automember-remove-condition**.
2. Lorsque vous y êtes invité, précisez :
   - **Automember rule**. Il s'agit du nom de la règle dont vous souhaitez supprimer une condition.
   - **Attribute Key**. Il s'agit de l'attribut de l'entrée cible. Par exemple, **uid** pour les utilisateurs.
   - **Grouping Type**. Indique si la condition à supprimer concerne un groupe d'utilisateurs ou un groupe d'hôtes. Saisissez **group** ou **hostgroup**.
   - **Inclusive regex** et **Exclusive regex**. Ils indiquent les conditions que vous souhaitez supprimer. Si vous ne souhaitez spécifier qu'une seule condition, appuyez sur **Enter** lorsque l'on vous demande l'autre condition.
     Par exemple :

```
$ ipa automember-remove-condition
Automember Rule: user_group
Attribute Key: uid
Grouping Type: group
[Inclusive Regex]: .*
[Exclusive Regex]:
-------------------------------------
Removed condition(s) from "user_group"
-------------------------------------
  Automember Rule: user_group
-----------------------------
Number of conditions removed 1
-----------------------------
```

22.8. APPLIQUER DES RÈGLES D'APPARTENANCE AUTOMATIQUE À DES ENTRÉES EXISTANTES À L'AIDE DE L'INTERFACE CLI DE L'IDM

Les règles Automember s'appliquent automatiquement aux entrées utilisateur et hôte créées après l'ajout des règles. Elles ne s'appliquent pas rétroactivement aux entrées qui existaient avant l'ajout des règles.

Pour appliquer les règles d'automember aux entrées ajoutées précédemment, vous devez reconstruire manuellement l'adhésion automatique. La reconstruction de l'adhésion automatique réévalue toutes les règles automember existantes et les applique soit à toutes les entrées d'utilisateurs ou d'hôtes, soit à des entrées spécifiques.

NOTE

Reconstruction de l'adhésion automatique **does not** supprimer les entrées utilisateur ou hôte des groupes, même si les entrées ne correspondent plus aux conditions d'inclusion du groupe. Pour les supprimer manuellement, voir Supprimer un membre d'un groupe d'utilisateurs à l'aide du CLI Id M ou Supprimer les membres d'un groupe d'hôtes IdM à l'aide du CLI.

Conditions préalables

- Vous devez être connecté en tant qu'administrateur. Pour plus de détails, voir le lien : Utiliser kinit pour se connecter manuellement à l'IdM.

Procédure

- Pour rétablir l'adhésion automatique, entrez la commande **ipa automember-rebuild**. Utilisez les options suivantes pour spécifier les entrées à cibler :
  - Pour rétablir l'adhésion automatique de tous les utilisateurs, utilisez l'option **--type=group**:

```
$ ipa automember-rebuild --type=group
---------------------------------------------------------
Automember rebuild task finished. Processed (9) entries.
---------------------------------------------------------
```

  - Pour rétablir l'adhésion automatique pour tous les hôtes, utilisez l'option **--type=hostgroup**.
  - Pour rétablir l'affiliation automatique pour un ou plusieurs utilisateurs spécifiés, utilisez l'option **--users=*target_user*** pour reconstruire l'adhésion automatique pour l'utilisateur ou les utilisateurs spécifiés :

```
$ ipa automember-rebuild --users=target_user1 --users=target_user2
---------------------------------------------------------
Automember rebuild task finished. Processed (2) entries.
---------------------------------------------------------
```

  - Pour reconstruire l'adhésion automatique pour un ou plusieurs hôtes spécifiés, utilisez l'option **--hosts=*client.idm.example.com*** pour reconstruire l'adhésion automatique d'un ou plusieurs hôtes.

22.9. CONFIGURATION D'UN GROUPE DE MEMBRES PAR DÉFAUT À L'AIDE DE LA CLI IDM

Lorsque vous configurez un groupe de membres automatiques par défaut, les nouvelles entrées d'utilisateurs ou d'hôtes qui ne correspondent à aucune règle de membres automatiques sont automatiquement ajoutées à ce groupe par défaut.

Conditions préalables

- Vous devez être connecté en tant qu'administrateur. Pour plus de détails, voir Utilisation de kinit pour se connecter manuellement à l'IdM.
- Le groupe cible que vous souhaitez définir par défaut existe dans IdM.

Procédure

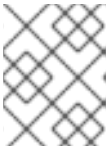
1. Entrez la commande **ipa automember-default-group-set** pour configurer un groupe de membres automatiques par défaut.

2. Lorsque vous y êtes invité, précisez :

   - **Default (fallback) Group**qui spécifie le nom du groupe cible.

   - **Grouping Type**qui précise si la cible est un groupe d'utilisateurs ou un groupe d'hôtes. Pour cibler un groupe d'utilisateurs, entrez **group**. Pour cibler un groupe d'hôtes, entrez **hostgroup**.
     Par exemple :

```
$ ipa automember-default-group-set
Default (fallback) Group: default_user_group
Grouping Type: group
-------------------------------------------------
Set default (fallback) group for automember "default_user_group"
-------------------------------------------------
  Default (fallback) Group:
cn=default_user_group,cn=groups,cn=accounts,dc=example,dc=com
```



NOTE

Pour supprimer le groupe automember par défaut actuel, entrez la commande **ipa automember-default-group-remove**.

Verification steps

- Pour vérifier que le groupe est correctement défini, entrez la commande **ipa automember-default-group-show**. La commande affiche le groupe automember par défaut actuel. Par exemple :

```
$ ipa automember-default-group-show
Grouping Type: group
  Default (fallback) Group:
cn=default_user_group,cn=groups,cn=accounts,dc=example,dc=com
```

CHAPITRE 23. AUTOMATISATION DE L'APPARTENANCE À UN GROUPE À L'AIDE DE L'INTERFACE WEB IDM

L'utilisation de l'appartenance automatique à un groupe vous permet d'affecter automatiquement des utilisateurs et des hôtes à des groupes en fonction de leurs attributs. Par exemple, vous pouvez

- Répartissez les entrées utilisateur des employés dans des groupes en fonction du responsable de l'employé, de son lieu de travail ou de tout autre attribut.
- Divisez les hôtes en fonction de leur classe, de leur lieu de résidence ou de tout autre attribut.
- Ajouter tous les utilisateurs ou tous les hôtes à un seul groupe global.

Ce chapitre couvre les sujets suivants :

23.1. AVANTAGES DE L'ADHÉSION AUTOMATIQUE À UN GROUPE

L'utilisation de l'adhésion automatique pour les utilisateurs vous permet de

- **Reduce the overhead of manually managing group memberships**
  Il n'est plus nécessaire d'affecter manuellement chaque utilisateur et chaque hôte à des groupes.
- **Improve consistency in user and host management**
  Les utilisateurs et les hôtes sont affectés à des groupes sur la base de critères strictement définis et évalués automatiquement.
- **Simplify the management of group-based settings**
  Divers paramètres sont définis pour les groupes, puis appliqués aux membres individuels du groupe, par exemple les règles **sudo**, l'automount ou le contrôle d'accès. L'ajout automatique d'utilisateurs et d'hôtes à des groupes facilite la gestion de ces paramètres.

23.2. RÈGLES DE L'AUTOMEMBER

Lors de la configuration de l'appartenance automatique à un groupe, l'administrateur définit des règles d'appartenance automatique. Une règle automember s'applique à un groupe cible d'utilisateurs ou d'hôtes spécifique. Elle ne peut pas s'appliquer à plusieurs groupes à la fois.

Après avoir créé une règle, l'administrateur y ajoute des conditions. Celles-ci précisent quels utilisateurs ou hôtes sont inclus ou exclus du groupe cible :

- **Inclusive conditions**
  Lorsqu'une entrée utilisateur ou hôte remplit une condition d'inclusion, elle est incluse dans le groupe cible.

- **Exclusive conditions**
  Lorsqu'une entrée utilisateur ou hôte remplit une condition d'exclusivité, elle n'est pas incluse dans le groupe cible.

Les conditions sont spécifiées sous forme d'expressions régulières au format PCRE (Perl-compatible regular expressions). Pour plus d'informations sur PCRE, voir la page de manuel **pcresyntax(3)**.

NOTE

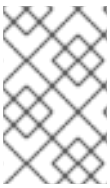
L'IdM évalue les conditions exclusives avant les conditions inclusives. En cas de conflit, les conditions exclusives l'emportent sur les conditions inclusives.

Une règle automember s'applique à toutes les entrées créées à l'avenir. Ces entrées seront automatiquement ajoutées au groupe cible spécifié. Si une entrée remplit les conditions spécifiées dans plusieurs règles automember, elle sera ajoutée à tous les groupes correspondants.

Les entrées existantes sont **not** affectées par la nouvelle règle. Si vous souhaitez modifier des entrées existantes, reportez-vous à la section Application des règles automember aux entrées existantes à l'aide de l'interface Web IdM.

23.3. AJOUT D'UNE RÈGLE DE MEMBRE AUTOMATIQUE À L'AIDE DE L'INTERFACE WEB IDM

Cette section décrit l'ajout d'une règle de membre automatique à l'aide de l'interface Web IdM. Pour plus d'informations sur les règles de membre automatique, voir Règles de membre automatique .



NOTE

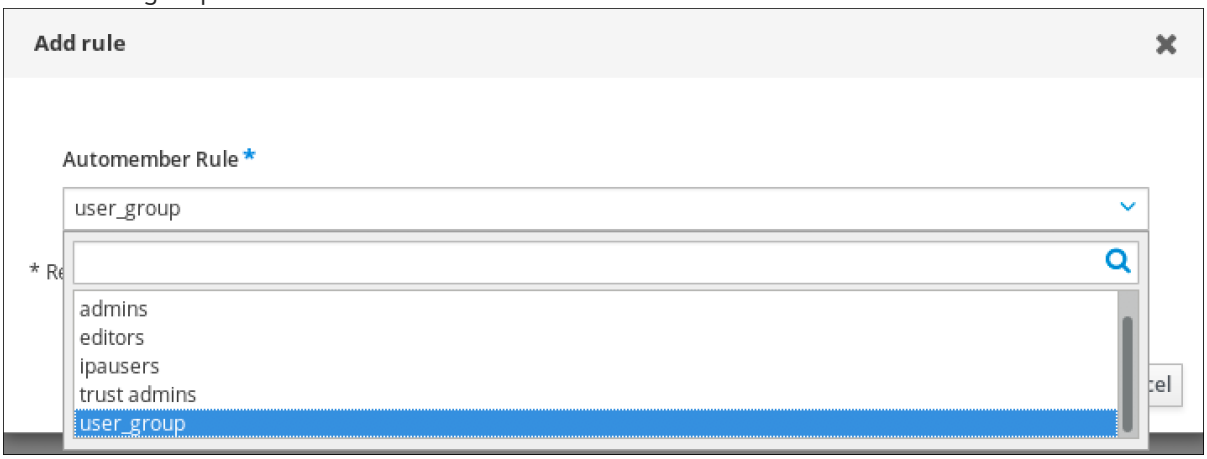
Les entrées existantes sont **not** affectées par la nouvelle règle. Si vous souhaitez modifier des entrées existantes, reportez-vous à la section Application des règles automember aux entrées existantes à l'aide de l'interface Web IdM.

Conditions préalables

- Vous êtes connecté à l'interface Web IdM.
- Vous devez être membre du groupe **admins**.
- Le groupe cible de la nouvelle règle existe dans IdM.

Procédure

1. Cliquez sur **Identity → Automember**, puis sélectionnez **User group rules** ou **Host group rules**.
2. Cliquez sur **Add**.
3. Dans le champ **Automember rule**, sélectionnez le groupe auquel la règle s'appliquera. Il s'agit du nom du groupe cible.



4. Cliquez sur **Add** pour confirmer.
5. Facultatif : vous pouvez ajouter des conditions à la nouvelle règle en suivant la procédure décrite dans la section Ajout d'une condition à une règle automember à l'aide de l'interface Web IdM.

23.4. AJOUT D'UNE CONDITION À UNE RÈGLE DE MEMBRE AUTOMATIQUE À L'AIDE DE L'INTERFACE WEB IDM

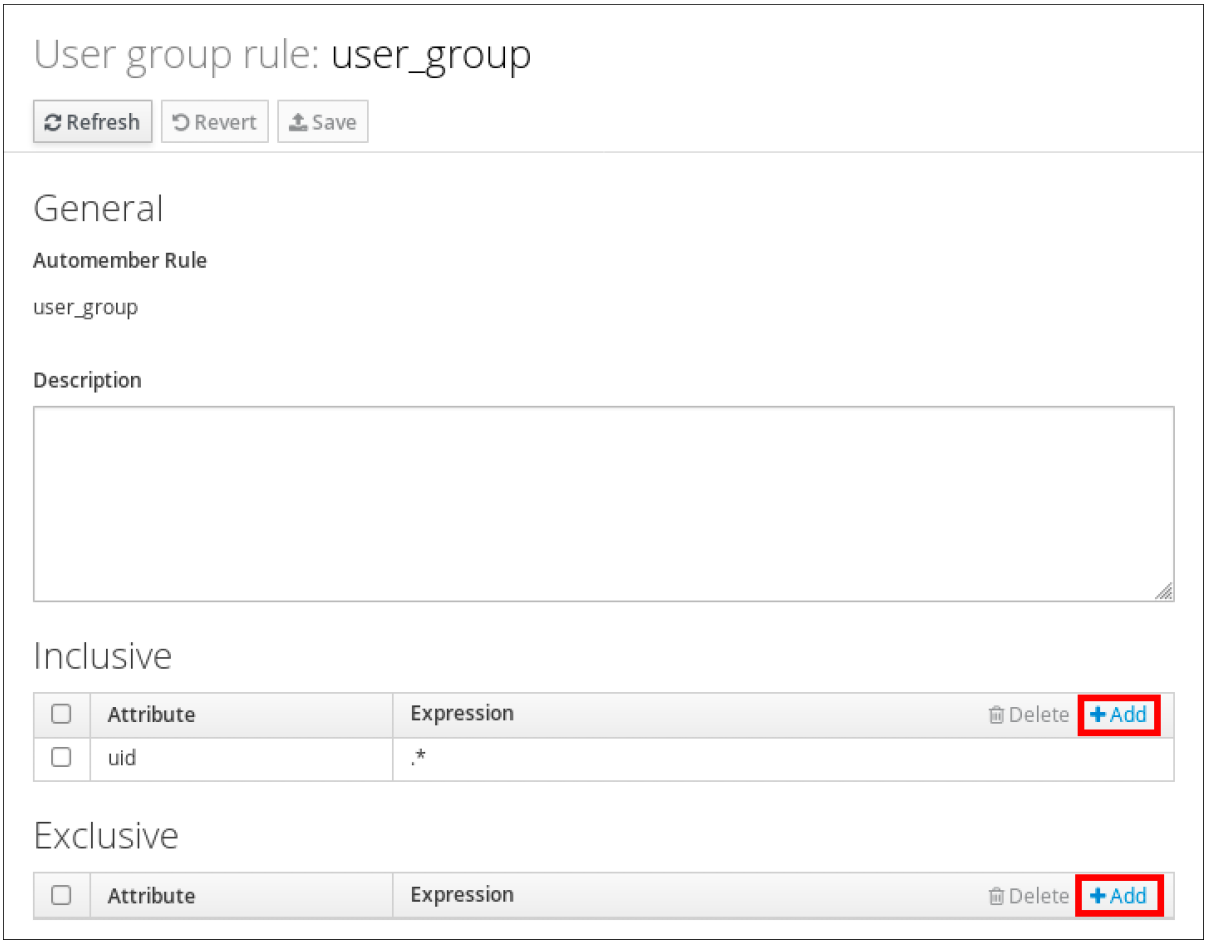
Cette section explique comment ajouter une condition à une règle d'appartenance à un groupe à l'aide de l'interface Web de l'IdM. Pour plus d'informations sur les règles de membre automatique, voir Règles de membre automatique.

Conditions préalables

- Vous êtes connecté à l'interface Web IdM.
- Vous devez être membre du groupe **admins**.
- La règle cible existe dans l'IdM.

Procédure

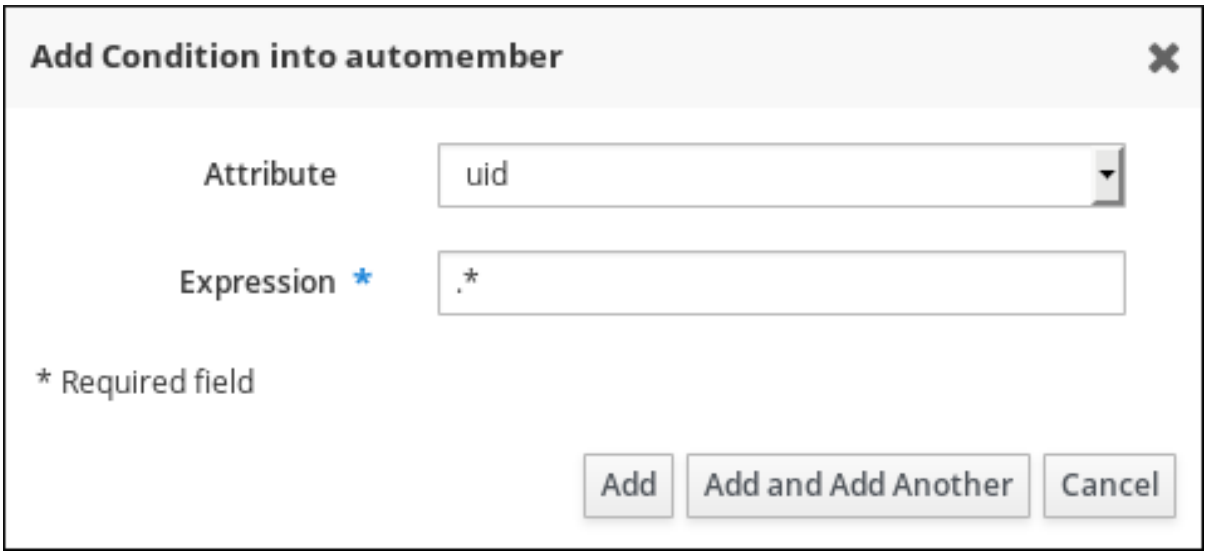
1. Cliquez sur **Identity → Automember**, puis sélectionnez **User group rules** ou **Host group rules**.
2. Cliquez sur la règle à laquelle vous souhaitez ajouter une condition.
3. Dans les sections **Inclusive** ou **Exclusive**, cliquez sur Ajouter.



4. Dans le champ **Attribute**, sélectionnez l'attribut requis, par exemple *uid*.

5. Dans le champ **Expression**, définissez une expression régulière.

6. Cliquez sur **Add**.
   Par exemple, la condition suivante vise tous les utilisateurs dont l'attribut d'identification de l'utilisateur (uid) contient une valeur quelconque (.*).



23.5. VISUALISATION DES RÈGLES ET CONDITIONS EXISTANTES POUR LES MEMBRES AUTOMATIQUES À L'AIDE DE L'INTERFACE WEB IDM

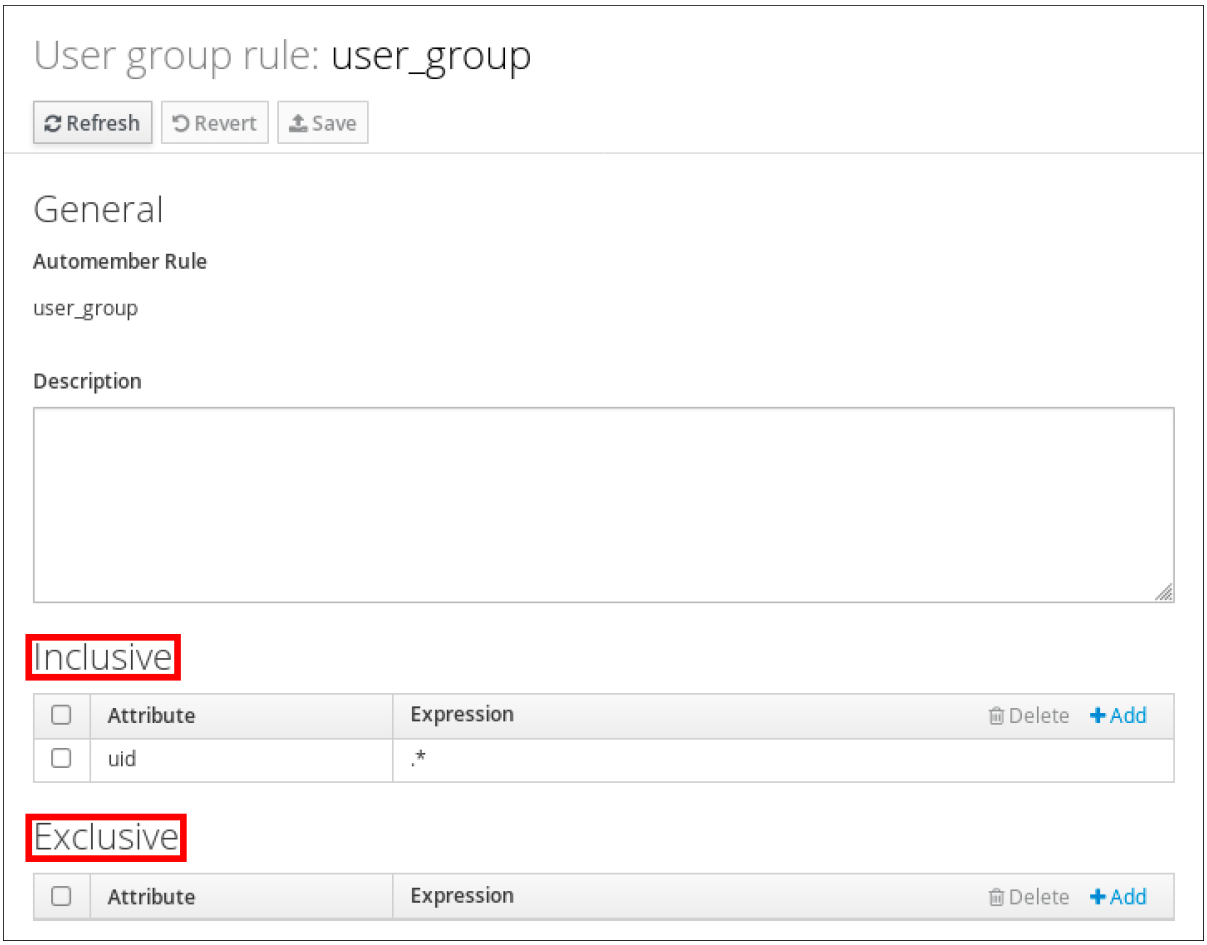
Cette section décrit comment visualiser les règles et conditions existantes pour les membres de l'association à l'aide de l'interface Web IdM.

Conditions préalables

- Vous êtes connecté à l'interface Web IdM.
- Vous devez être membre du groupe **admins**.

Procédure

1. Cliquez sur **Identity → Automember**, et sélectionnez **User group rules** ou **Host group rules** pour afficher les règles respectives des membres automatiques.
2. Facultatif : Cliquez sur une règle pour afficher les conditions de cette règle dans les sections **Inclusive** ou **Exclusive**.



23.6. SUPPRESSION D'UNE RÈGLE DE MEMBRE AUTOMATIQUE À L'AIDE DE L'INTERFACE WEB IDM

Cette section décrit comment supprimer une règle automember à l'aide de l'interface Web IdM.

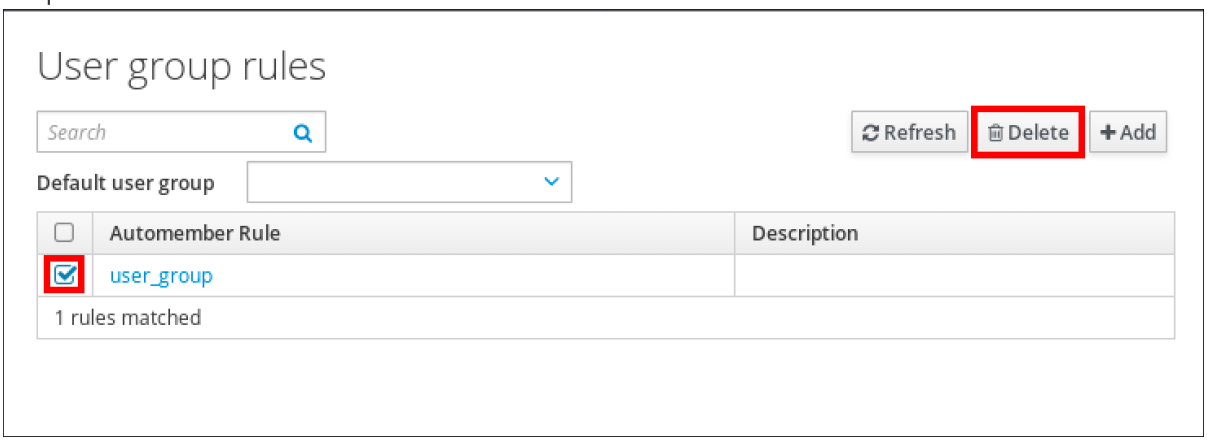
La suppression d'une règle automember supprime également toutes les conditions associées à la règle. Pour supprimer uniquement des conditions spécifiques d'une règle, voir Suppression d'une condition d'une règle automember à l'aide de l'interface Web IdM.

Conditions préalables

- Vous êtes connecté à l'interface Web IdM.
- Vous devez être membre du groupe **admins**.

Procédure

1. Cliquez sur **Identity → Automember**, et sélectionnez **User group rules** ou **Host group rules** pour afficher les règles respectives des membres automatiques.
2. Cochez la case en regard de la règle que vous souhaitez supprimer.
3. Cliquez sur **Delete**.



4. Cliquez sur **Delete** pour confirmer.

23.7. SUPPRESSION D'UNE CONDITION D'UNE RÈGLE DE MEMBRE AUTOMATIQUE À L'AIDE DE L'INTERFACE WEB IDM

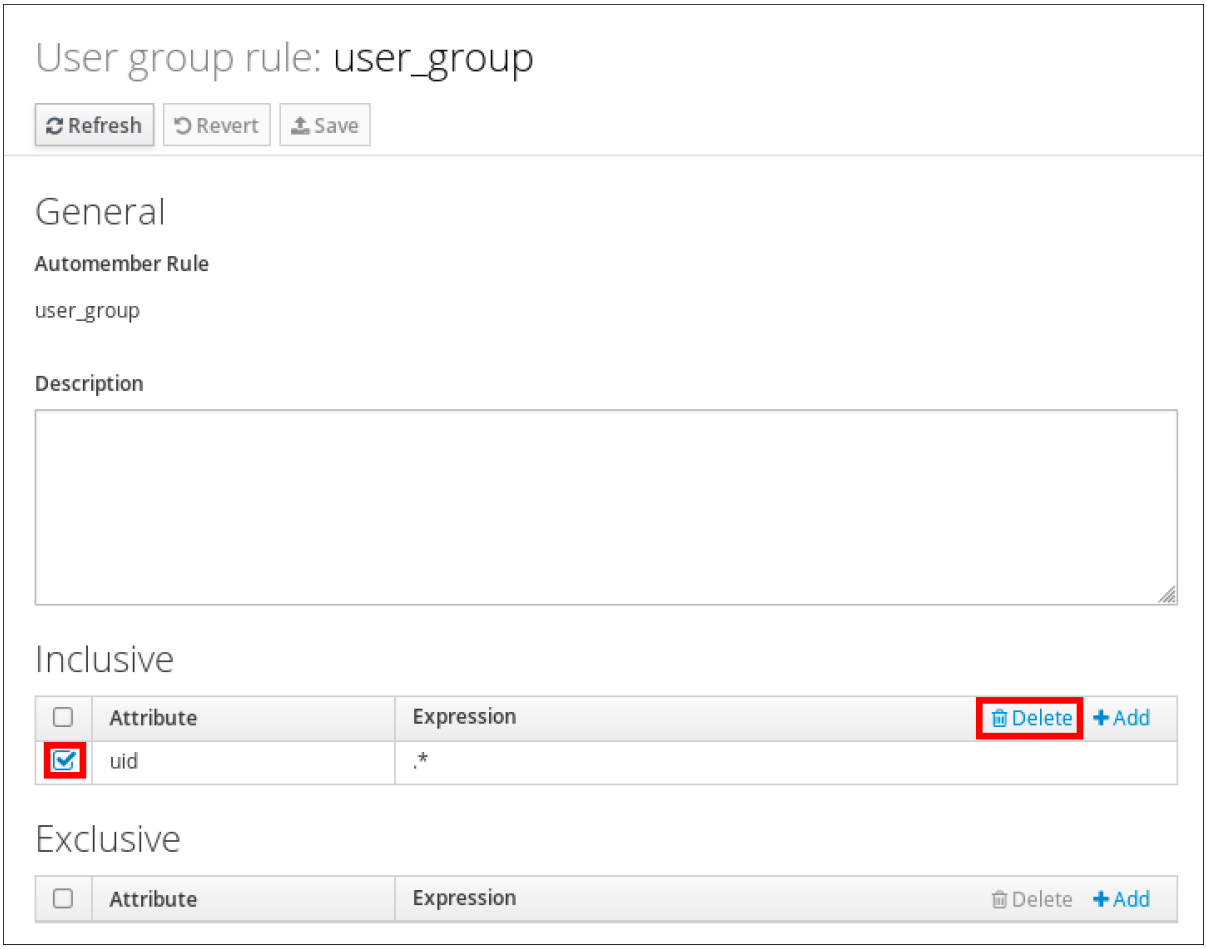
Cette section décrit comment supprimer une condition spécifique d'une règle automember à l'aide de l'interface Web IdM.

Conditions préalables

- Vous êtes connecté à l'interface Web IdM.
- Vous devez être membre du groupe **admins**.

Procédure

1. Cliquez sur **Identity → Automember**, et sélectionnez **User group rules** ou **Host group rules** pour afficher les règles respectives des membres automatiques.
2. Cliquez sur une règle pour voir les conditions de cette règle dans les sections **Inclusive** ou **Exclusive**.
3. Cochez la case en regard des conditions que vous souhaitez supprimer.
4. Cliquez sur **Delete**.

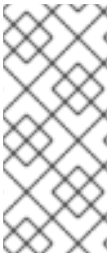


5. Cliquez sur **Delete** pour confirmer.

23.8. APPLICATION DE RÈGLES D'APPARTENANCE À UN MEMBRE AUTOMATIQUE À DES ENTRÉES EXISTANTES À L'AIDE DE L'INTERFACE WEB DE L'IDM

Les règles Automember s'appliquent automatiquement aux entrées utilisateur et hôte créées après l'ajout des règles. Elles ne s'appliquent pas rétroactivement aux entrées qui existaient avant l'ajout des règles.

Pour appliquer les règles d'automember aux entrées ajoutées précédemment, vous devez reconstruire manuellement l'adhésion automatique. La reconstruction de l'adhésion automatique réévalue toutes les règles automember existantes et les applique soit à toutes les entrées d'utilisateurs ou d'hôtes, soit à des entrées spécifiques.



NOTE

Reconstruction de l'adhésion automatique **does not** supprimer les entrées d'utilisateurs ou d'hôtes des groupes, même si les entrées ne correspondent plus aux conditions d'inclusion du groupe. Pour les supprimer manuellement, voir Supprimer un membre d'un groupe d'utilisateurs à l'aide de l'interface Web d'IdM ou Supprimer les membres d'un groupe d'hôtes dans l'interface Web d'IdM.

23.8.1. Reconstruction de l'adhésion automatique pour tous les utilisateurs ou hôtes

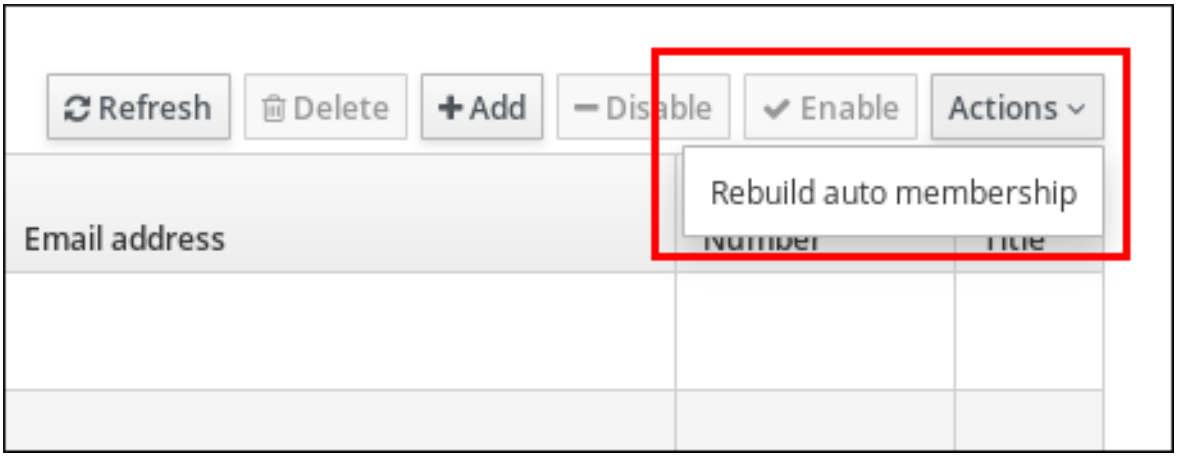
Cette section décrit comment rétablir l'affiliation automatique pour toutes les entrées d'utilisateurs ou d'hôtes.

Conditions préalables

- Vous êtes connecté à l'interface Web IdM.
- Vous devez être membre du groupe **admins**.

Procédure

1. Sélectionnez **Identity → Users** ou **Hosts**.
2. Cliquez sur **Actions → Rebuild auto membership**.



23.8.2. Reconstruction de l'adhésion automatique pour un seul utilisateur ou hôte

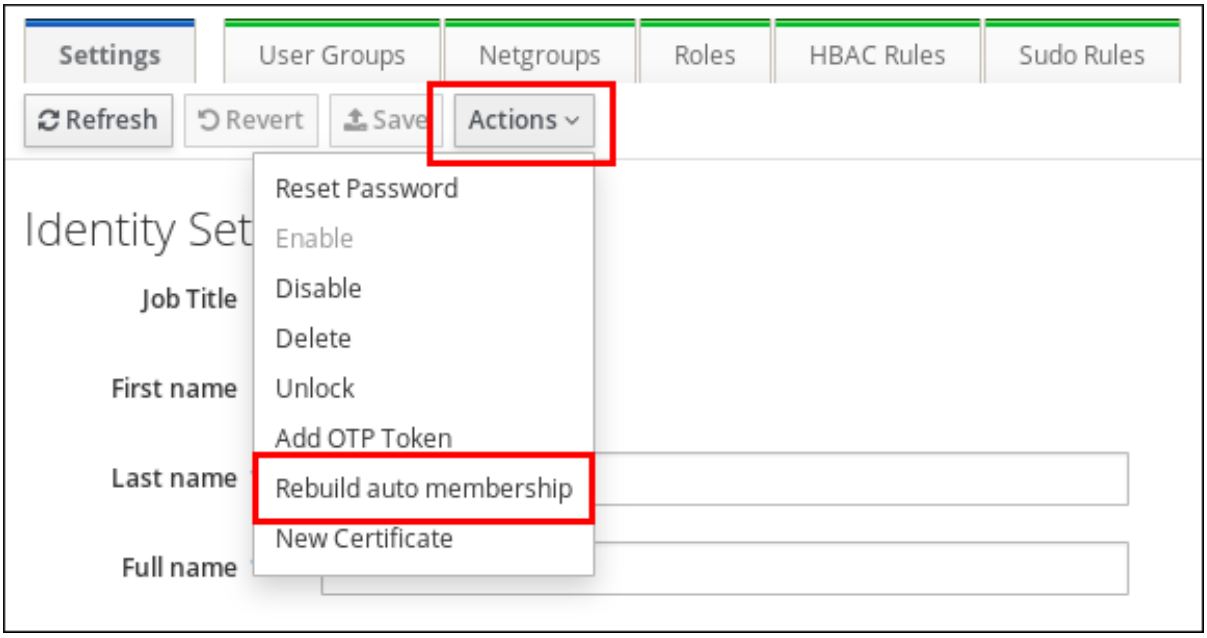
Cette section décrit comment reconstruire l'adhésion automatique pour une entrée utilisateur ou hôte spécifique.

Conditions préalables

- Vous êtes connecté à l'interface Web IdM.
- Vous devez être membre du groupe **admins**.

Procédure

1. Sélectionnez **Identity → Users** ou **Hosts**.
2. Cliquez sur le nom d'utilisateur ou d'hôte requis.
3. Cliquez sur **Actions → Rebuild auto membership**.



23.9. CONFIGURATION D'UN GROUPE D'UTILISATEURS PAR DÉFAUT À L'AIDE DE L'INTERFACE WEB IDM

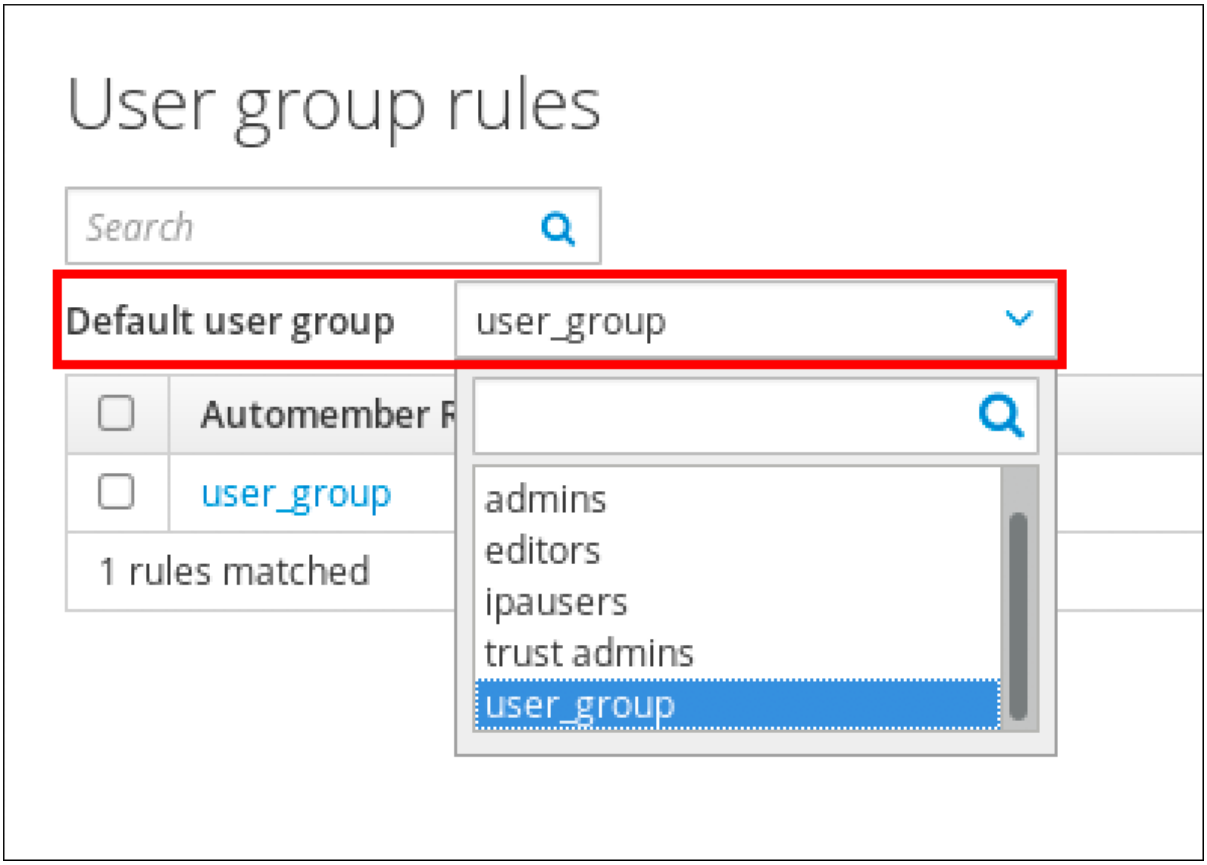
Lorsque vous configurez un groupe d'utilisateurs par défaut, les nouvelles entrées d'utilisateurs qui ne correspondent à aucune règle automember sont automatiquement ajoutées à ce groupe par défaut.

Conditions préalables

- Vous êtes connecté à l'interface Web IdM.
- Vous devez être membre du groupe **admins**.
- Le groupe d'utilisateurs cible que vous souhaitez définir par défaut existe dans l'IdM.

Procédure

1. Cliquez sur **Identity → Automember**, puis sélectionnez **User group rules**.
2. Dans le champ **Default user group**, sélectionnez le groupe que vous souhaitez définir comme groupe d'utilisateurs par défaut.



23.10. CONFIGURATION D'UN GROUPE D'HÔTES PAR DÉFAUT À L'AIDE DE L'INTERFACE WEB IDM

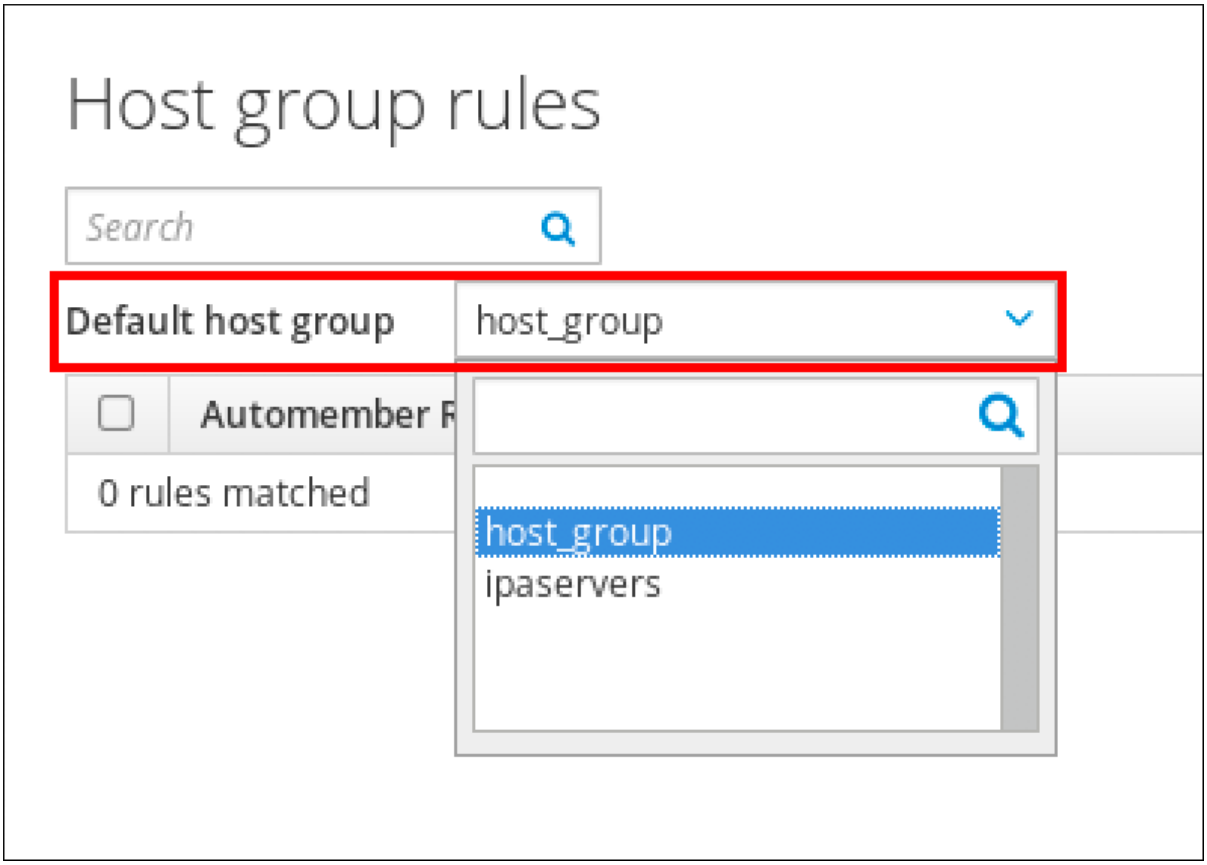
Lorsque vous configurez un groupe d'hôtes par défaut, les nouvelles entrées d'hôtes qui ne correspondent à aucune règle automember sont automatiquement ajoutées à ce groupe par défaut.

Conditions préalables

- Vous êtes connecté à l'interface Web IdM.
- Vous devez être membre du groupe **admins**.
- Le groupe d'hôtes cible que vous souhaitez définir par défaut existe dans IdM.

Procédure

1. Cliquez sur **Identity → Automember**, puis sélectionnez **Host group rules**.
2. Dans le champ **Default host group**, sélectionnez le groupe que vous souhaitez définir comme groupe d'hôtes par défaut.


Host group rules
Search
Default host group
host_group
Automember R
0 rules matched
host_group
ipaservers

CHAPITRE 24. UTILISER ANSIBLE POUR AUTOMATISER L'APPARTENANCE À UN GROUPE DANS IDM

L'appartenance automatique à un groupe vous permet d'affecter aux utilisateurs et aux hôtes des groupes d'utilisateurs et des groupes d'hôtes automatiquement, en fonction de leurs attributs. Par exemple, vous pouvez

- Répartissez les entrées utilisateur des employés dans des groupes en fonction du responsable, de la localisation, du poste ou de tout autre attribut de l'employé. Vous pouvez dresser la liste de tous les attributs en saisissant **ipa user-add --help** sur la ligne de commande.
- Divisez les hôtes en groupes en fonction de leur classe, de leur emplacement ou de tout autre attribut. Vous pouvez dresser la liste de tous les attributs en entrant **ipa host-add --help** dans la ligne de commande.
- Ajouter tous les utilisateurs ou tous les hôtes à un seul groupe global.

Vous pouvez utiliser Red Hat Ansible Engine pour automatiser la gestion de l'appartenance automatique à un groupe dans Identity Management (IdM).

Cette section couvre les sujets suivants :

- Préparation du nœud de contrôle Ansible pour la gestion de l'IdM
- Utiliser Ansible pour s'assurer qu'une règle automember pour un groupe d'utilisateurs IdM est présente
- Utiliser Ansible pour s'assurer qu'une condition est présente dans une règle de membre automatique d'un groupe d'utilisateurs IdM
- Utiliser Ansible pour s'assurer qu'une condition est absente dans une règle de membre automatique d'un groupe d'utilisateurs IdM
- Utiliser Ansible pour s'assurer qu'une règle automember pour un groupe IdM est absente
- Utiliser Ansible pour s'assurer qu'une condition est présente dans une règle de membre automatique d'un groupe d'hôtes IdM

24.1. PRÉPARATION DU NŒUD DE CONTRÔLE ANSIBLE POUR LA GESTION DE L'IDM

En tant qu'administrateur système gérant la gestion des identités (IdM), lorsque vous travaillez avec Red Hat Ansible Engine, il est recommandé de procéder comme suit :

- Créez un sous-répertoire dédié aux playbooks Ansible dans votre répertoire personnel, par exemple **~/MyPlaybooks**.
- Copiez et adaptez les exemples de playbooks Ansible des répertoires et sous-répertoires **/usr/share/doc/ansible-freeipa/*** et **/usr/share/doc/rhel-system-roles/*** dans votre répertoire **~/MyPlaybooks**.
- Incluez votre fichier d'inventaire dans votre répertoire **~/MyPlaybooks**.

En suivant cette pratique, vous pouvez trouver tous vos playbooks en un seul endroit et vous pouvez exécuter vos playbooks sans invoquer les privilèges root.

NOTE

Vous n'avez besoin que des privilèges **root** sur les nœuds gérés pour exécuter les rôles **ipaserver**, **ipareplica**, **ipaclient**, **ipabackup**, **ipasmartcard_server** et **ipasmartcard_client ansible-freeipa** . Ces rôles nécessitent un accès privilégié aux répertoires et au gestionnaire de paquets logiciels **dnf**.

Cette section décrit comment créer le répertoire **~/MyPlaybooks** et le configurer de manière à ce que vous puissiez l'utiliser pour stocker et exécuter des playbooks Ansible.

Conditions préalables

- Vous avez installé un serveur IdM sur vos nœuds gérés, ***server.idm.example.com*** et ***replica.idm.example.com***.
- Vous avez configuré le DNS et le réseau pour pouvoir vous connecter aux nœuds gérés, ***server.idm.example.com*** et ***replica.idm.example.com***directement à partir du nœud de contrôle.
- Vous connaissez le mot de passe de l'IdM **admin**.

Procédure

1. Créez un répertoire pour votre configuration Ansible et vos playbooks dans votre répertoire personnel :

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

2. Allez dans le répertoire **~/MyPlaybooks/**:

```
$ cd ~/MyPlaybooks
```

3. Créez le fichier **~/MyPlaybooks/ansible.cfg** avec le contenu suivant :

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

[privilege_escalation]
become=True
```

4. Créez le fichier **~/MyPlaybooks/inventory** avec le contenu suivant :

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

[ipareplicas]
replica1.idm.example.com
replica2.idm.example.com

[ipacluster:children]
ipaserver
ipareplicas

[ipacluster:vars]
ipaadmin_password=SomeADMINpassword
```

```
[ipaclients]
ipaclient1.example.com
ipaclient2.example.com

[ipaclients:vars]
ipaadmin_password=SomeADMINpassword
```

Cette configuration définit deux groupes d'hôtes, **eu** et **us**, pour les hôtes de ces sites. En outre, cette configuration définit le groupe d'hôtes **ipaserver**, qui contient tous les hôtes des groupes **eu** et **us**.

5. [Facultatif] Créez une clé publique et une clé privée SSH. Pour simplifier l'accès dans votre environnement de test, ne définissez pas de mot de passe pour la clé privée :

```
$ ssh-keygen
```

6. Copiez la clé publique SSH dans le compte IdM **admin** sur chaque nœud géré :

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

Vous devez saisir le mot de passe IdM **admin** lorsque vous entrez dans ces commandes.

Ressources supplémentaires

- Installation d'un serveur de gestion des identités à l'aide d'un playbook Ansible .
- Comment constituer votre inventaire .

24.2. UTILISER ANSIBLE POUR S'ASSURER QU'UNE RÈGLE AUTOMEMBER POUR UN GROUPE D'UTILISATEURS IDM EST PRÉSENTE

La procédure suivante décrit comment utiliser un playbook Ansible pour s'assurer de l'existence d'une règle **automember** pour un groupe de gestion des identités (IdM). Dans l'exemple, la présence d'une règle **automember** est assurée pour le groupe d'utilisateurs **testing_group**.

Conditions préalables

- Vous connaissez le mot de passe de l'IdM **admin**.
- Le groupe d'utilisateurs **testing_group** existe dans IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.

- L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

Procédure

1. Naviguez jusqu'à votre répertoire ***~/MyPlaybooks/*** répertoire :

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

2. Copiez le fichier **automember-group-present.yml** Ansible playbook situé dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/automember/**:

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/automember/automember-group-present.yml automember-group-present-copy.yml
```

3. Ouvrez le fichier **automember-group-present-copy.yml** pour le modifier.
4. Adaptez le fichier en définissant les variables suivantes dans la section **ipaautomember** task :
   - Fixer la variable **ipaadmin_password** au mot de passe de l'IdM **admin**.
   - Fixer la variable **name** à **testing_group**.
   - Fixer la variable **automember_type** à **group**.
   - Assurez-vous que la variable **state** est définie sur **present**.

   Il s'agit du fichier playbook Ansible modifié pour l'exemple actuel :

```
---
- name: Automember group present example
  hosts: ipaserver
  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Ensure group automember rule admins is present
    ipaautomember:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: testing_group
      automember_type: group
      state: present
```

5. Enregistrer le fichier.
6. Exécutez le playbook Ansible. Spécifiez le fichier du livre de jeu, le fichier contenant le mot de passe protégeant le fichier **secret.yml** et le fichier d'inventaire :

```
$ ansible-playbook --vault-password-file=password_file -v -i inventory automember-group-present-copy.yml
```

Ressources supplémentaires

- Voir Avantages de l'adhésion automatique à un groupe et Règles Automember.

- Voir Utiliser Ansible pour s'assurer qu'une condition est présente dans une règle de membre automatique d'un groupe d'utilisateurs IdM.
- Voir le fichier **README-automember.md** dans le répertoire **/usr/share/doc/ansible-freeipa/**.
- Voir le répertoire **/usr/share/doc/ansible-freeipa/playbooks/automember**.

24.3. UTILISER ANSIBLE POUR S'ASSURER QU'UNE CONDITION SPÉCIFIÉE EST PRÉSENTE DANS UNE RÈGLE DE MEMBRE AUTOMATIQUE D'UN GROUPE D'UTILISATEURS IDM

La procédure suivante décrit comment utiliser un playbook Ansible pour s'assurer qu'une condition spécifiée existe dans une règle **automember** pour un groupe de gestion des identités (IdM). Dans l'exemple, la présence d'une condition liée à l'UID dans la règle **automember** est assurée pour le groupe **testing_group**. En spécifiant la condition **.***, vous vous assurez que tous les futurs utilisateurs IdM deviennent automatiquement membres du groupe **testing_group**.

Conditions préalables

- Vous connaissez le mot de passe de l'IdM **admin**.
- Le groupe d'utilisateurs **testing_group** et la règle du groupe d'utilisateurs automember existent dans IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

Procédure

1. Naviguez jusqu'à votre répertoire ***~/MyPlaybooks/*** répertoire :

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

2. Copiez le fichier **automember-hostgroup-rule-present.yml** Ansible playbook situé dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/automember/** et nommez-le, par exemple, **automember-usergroup-rule-present.yml**:

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/automember/automember-hostgroup-rule-present.yml automember-usergroup-rule-present.yml
```

3. Ouvrez le fichier **automember-usergroup-rule-present.yml** pour le modifier.
4. Adapter le fichier en modifiant les paramètres suivants :

- Renommez le playbook pour qu'il corresponde à votre cas d'utilisation, par exemple : **Automember user group rule member present**
- Renommez la tâche pour qu'elle corresponde à votre cas d'utilisation, par exemple : **Ensure an automember condition for a user group is present**.
- Définissez les variables suivantes dans la section **ipaautomember** task :
  - Fixer la variable **ipaadmin_password** au mot de passe de l'IdM **admin**.
  - Fixer la variable **name** à **testing_group**.
  - Fixer la variable **automember_type** à **group**.
  - Assurez-vous que la variable **state** est définie sur **present**.
  - Assurez-vous que la variable **action** est définie sur **member**.
  - Fixez la variable **inclusive key** à **UID**.
  - Fixer la variable **inclusive expression** à **.***

Il s'agit du fichier playbook Ansible modifié pour l'exemple actuel :

```
---
- name: Automember user group rule member present
  hosts: ipaserver
  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Ensure an automember condition for a user group is present
    ipaautomember:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: testing_group
      automember_type: group
      state: present
      action: member
      inclusive:
        - key: UID
          expression: .*
```

5. Enregistrer le fichier.
6. Exécutez le playbook Ansible. Spécifiez le fichier du livre de jeu, le fichier contenant le mot de passe protégeant le fichier **secret.yml** et le fichier d'inventaire :

```
$ ansible-playbook --vault-password-file=password_file -v -i inventory automember-usergroup-rule-present.yml
```

Verification steps

1. Se connecter en tant qu'administrateur IdM.

```
$ kinit admin
```

2. Ajouter un utilisateur, par exemple :

```
$ ipa user-add user101 --first user --last 101
----------------------
Added user "user101"
----------------------
  User login: user101
  First name: user
  Last name: 101
  ...
  Member of groups: ipausers, testing_group
  ...
```

Ressources supplémentaires

- Voir Application des règles automember aux entrées existantes à l'aide de l'interface CLI de l'IdM.
- Voir Avantages de l'adhésion automatique à un groupe et Règles Automember.
- Voir le fichier **README-automember.md** dans le répertoire **/usr/share/doc/ansible-freeipa/**.
- Voir le répertoire **/usr/share/doc/ansible-freeipa/playbooks/automember**.

24.4. UTILISER ANSIBLE POUR S'ASSURER QU'UNE CONDITION EST ABSENTE D'UNE RÈGLE DE MEMBRE AUTOMATIQUE D'UN GROUPE D'UTILISATEURS IDM

La procédure suivante décrit comment utiliser un playbook Ansible pour s'assurer qu'une condition est absente d'une règle **automember** pour un groupe de gestion des identités (IdM). Dans l'exemple, l'absence d'une condition dans la règle **automember** est garantie et spécifie que les utilisateurs dont **initials** est **dp** doivent être inclus. La règle automember est appliquée au groupe **testing_group**. En appliquant la condition, vous vous assurez qu'aucun futur utilisateur IdM dont les initiales sont **dp** ne devienne membre du groupe **testing_group**.

Conditions préalables

- Vous connaissez le mot de passe de l'IdM **admin**.
- Le groupe d'utilisateurs **testing_group** et la règle du groupe d'utilisateurs automember existent dans IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

Procédure

1. Naviguez jusqu'à votre répertoire ***~/MyPlaybooks/*** répertoire :

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

2. Copiez le fichier **automember-hostgroup-rule-absent.yml** Ansible playbook situé dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/automember/** et nommez-le, par exemple, **automember-usergroup-rule-absent.yml**:

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/automember/automember-hostgroup-rule-absent.yml automember-usergroup-rule-absent.yml
```

3. Ouvrez le fichier **automember-usergroup-rule-absent.yml** pour le modifier.

4. Adapter le fichier en modifiant les paramètres suivants :

   - Renommez le playbook pour qu'il corresponde à votre cas d'utilisation, par exemple : **Automember user group rule member absent**
   - Renommez la tâche pour qu'elle corresponde à votre cas d'utilisation, par exemple : **Ensure an automember condition for a user group is absent**.
   - Définissez les variables suivantes dans la section **ipaautomember** task :
     - Fixer la variable **ipaadmin_password** au mot de passe de l'IdM **admin**.
     - Fixer la variable **name** à **testing_group**.
     - Fixer la variable **automember_type** à **group**.
     - Assurez-vous que la variable **state** est définie sur **absent**.
     - Assurez-vous que la variable **action** est définie sur **member**.
     - Fixez la variable **inclusive key** à **initials**.
     - Fixez la variable **inclusive expression** à **dp**.

Il s'agit du fichier playbook Ansible modifié pour l'exemple actuel :

```
---
- name: Automember user group rule member absent
  hosts: ipaserver
  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Ensure an automember condition for a user group is absent
    ipaautomember:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: testing_group
      automember_type: group
      state: absent
      action: member
      inclusive:
        - key: initials
          expression: dp
```

5. Enregistrer le fichier.

6. Exécutez le playbook Ansible. Spécifiez le fichier du livre de jeu, le fichier contenant le mot de passe protégeant le fichier **secret.yml** et le fichier d'inventaire :

```
$ ansible-playbook --vault-password-file=password_file -v -i inventory automember-usergroup-rule-absent.yml
```

Verification steps

1. Se connecter en tant qu'administrateur IdM.

```
$ kinit admin
```

2. Affichez le groupe automember :

```
$ ipa automember-show --type=group testing_group
 Automember Rule: testing_group
```

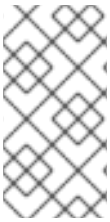
L'absence d'une entrée **Inclusive Regex: initials=dp** dans la sortie confirme que la règle de membre automatique **testing_group** ne contient pas la condition spécifiée.

Ressources supplémentaires

- Voir Application des règles automember aux entrées existantes à l'aide de l'interface CLI de l'IdM.
- Voir Avantages de l'adhésion automatique à un groupe et Règles Automember.
- Voir le fichier **README-automember.md** dans le répertoire **/usr/share/doc/ansible-freeipa/**.
- Voir le répertoire **/usr/share/doc/ansible-freeipa/playbooks/automember**.

24.5. UTILISER ANSIBLE POUR S'ASSURER QU'UNE RÈGLE AUTOMEMBER POUR UN GROUPE D'UTILISATEURS IDM EST ABSENTE

La procédure suivante décrit comment utiliser un playbook Ansible pour s'assurer qu'une règle **automember** est absente pour un groupe de gestion d'identité (IdM). Dans l'exemple, l'absence d'une règle **automember** est garantie pour le groupe **testing_group**.



NOTE

La suppression d'une règle de membre automatique supprime également toutes les conditions associées à la règle. Pour ne supprimer que des conditions spécifiques d'une règle, voir Utiliser Ansible pour s'assurer qu'une condition est absente d'une règle de membre automatique d'un groupe d'utilisateurs IdM.

Conditions préalables

- Vous connaissez le mot de passe de l'IdM **admin**.

- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

Procédure

1. Naviguez jusqu'à votre répertoire ***~/MyPlaybooks/*** répertoire :

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

2. Copiez le fichier **automember-group-absent.yml** Ansible playbook situé dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/automember/**:

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/automember/automember-group-absent.yml automember-group-absent-copy.yml
```

3. Ouvrez le fichier **automember-group-absent-copy.yml** pour le modifier.
4. Adaptez le fichier en définissant les variables suivantes dans la section **ipaautomember** task :
   - Fixer la variable **ipaadmin_password** au mot de passe de l'IdM **admin**.
   - Fixer la variable **name** à **testing_group**.
   - Fixer la variable **automember_type** à **group**.
   - Assurez-vous que la variable **state** est définie sur **absent**.

   Il s'agit du fichier playbook Ansible modifié pour l'exemple actuel :

```
---
- name: Automember group absent example
  hosts: ipaserver
  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Ensure group automember rule admins is absent
    ipaautomember:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: testing_group
      automember_type: group
      state: absent
```

5. Enregistrer le fichier.
6. Exécutez le playbook Ansible. Spécifiez le fichier du livre de jeu, le fichier contenant le mot de passe protégeant le fichier **secret.yml** et le fichier d'inventaire :

```
$ ansible-playbook --vault-password-file=password_file -v -i inventory automember-group-absent.yml
```

Ressources supplémentaires

- Voir Avantages de l'adhésion automatique à un groupe et Règles Automember.
- Voir le fichier **README-automember.md** dans le répertoire **/usr/share/doc/ansible-freeipa/**.
- Voir le répertoire **/usr/share/doc/ansible-freeipa/playbooks/automember**.

24.6. UTILISER ANSIBLE POUR S'ASSURER QU'UNE CONDITION EST PRÉSENTE DANS UNE RÈGLE DE MEMBRE AUTOMATIQUE D'UN GROUPE D'HÔTES IDM

Cette section décrit comment utiliser Ansible pour s'assurer qu'une condition est présente dans une règle de membre automatique de groupe d'hôtes IdM. L'exemple décrit comment s'assurer que les hôtes dont le **FQDN** est **.*.idm.example.com** sont membres du groupe d'hôtes **primary_dns_domain_hosts** et que les hôtes dont le **FQDN** est **.*.example.org** ne sont pas membres du groupe d'hôtes **primary_dns_domain_hosts**.

Conditions préalables

- Vous connaissez le mot de passe de l'IdM **admin**.
- Le groupe d'hôtes **primary_dns_domain_hosts** et la règle du groupe d'hôtes automember existent dans IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

Procédure

1. Naviguez jusqu'à votre répertoire ***~/MyPlaybooks/*** répertoire :

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

2. Copiez le fichier **automember-hostgroup-rule-present.yml** Ansible playbook situé dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/automember/**:

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/automember/automember-hostgroup-rule-present.yml automember-hostgroup-rule-present-copy.yml
```

3. Ouvrez le fichier **automember-hostgroup-rule-present-copy.yml** pour le modifier.
4. Adaptez le fichier en définissant les variables suivantes dans la section **ipaautomember** task :
   - Fixer la variable **ipaadmin_password** au mot de passe de l'IdM **admin**.
   - Fixer la variable **name** à **primary_dns_domain_hosts**.
   - Fixer la variable **automember_type** à **hostgroup**.
   - Assurez-vous que la variable **state** est définie sur **present**.
   - Assurez-vous que la variable **action** est définie sur **member**.
   - Assurez-vous que la variable **inclusive key** est fixée à **fqdn**.
   - Définissez la variable **inclusive expression** correspondante à **.*.idm.example.com**.
   - Fixez la variable **exclusive key** à **fqdn**.
   - Définissez la variable **exclusive expression** correspondante à **.*.example.org**.

   Il s'agit du fichier playbook Ansible modifié pour l'exemple actuel :

```
---
- name: Automember user group rule member present
  hosts: ipaserver
  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Ensure an automember condition for a user group is present
    ipaautomember:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: primary_dns_domain_hosts
      automember_type: hostgroup
      state: present
      action: member
      inclusive:
        - key: fqdn
          expression: .*.idm.example.com
      exclusive:
        - key: fqdn
          expression: .*.example.org
```

5. Enregistrer le fichier.
6. Exécutez le playbook Ansible. Spécifiez le fichier du livre de jeu, le fichier contenant le mot de passe protégeant le fichier **secret.yml** et le fichier d'inventaire :

```
$ ansible-playbook --vault-password-file=password_file -v -i inventory automember-hostgroup-rule-present-copy.yml
```

Ressources supplémentaires

- Voir Application des règles automember aux entrées existantes à l'aide de l'interface CLI de l'IdM.

- Voir Avantages de l'adhésion automatique à un groupe et Règles Automember.
- Voir le fichier **README-automember.md** dans le répertoire **/usr/share/doc/ansible-freeipa/**.
- Voir le répertoire **/usr/share/doc/ansible-freeipa/playbooks/automember**.

24.7. RESSOURCES SUPPLÉMENTAIRES

- Gérer les comptes d'utilisateurs à l'aide de playbooks Ansible
- Gérer les hôtes à l'aide des playbooks Ansible
- Gérer les groupes d'utilisateurs à l'aide de playbooks Ansible
- Gestion des groupes d'hôtes à l'aide de la CLI IdM

CHAPITRE 25. DÉLÉGATION DE PERMISSIONS À DES GROUPES D'UTILISATEURS POUR GÉRER LES UTILISATEURS À L'AIDE DE LA CLI IDM

La délégation est l'une des méthodes de contrôle d'accès dans IdM, avec les règles en libre-service et le contrôle d'accès basé sur les rôles (RBAC). Vous pouvez utiliser la délégation pour attribuer des autorisations à un groupe d'utilisateurs afin de gérer des entrées pour un autre groupe d'utilisateurs.

Cette section couvre les sujets suivants :

- Règles de délégation
- Création d'une règle de délégation à l'aide de l'interface CLI de l'IdM
- Visualisation des règles de délégation existantes à l'aide de la CLI IdM
- Modification d'une règle de délégation à l'aide de la CLI IdM
- Suppression d'une règle de délégation à l'aide de la CLI IdM

25.1. RÈGLES DE DÉLÉGATION

Vous pouvez déléguer des autorisations à des groupes d'utilisateurs pour gérer les utilisateurs en créant **delegation rules**.

Les règles de délégation permettent à un groupe d'utilisateurs spécifique d'effectuer des opérations d'écriture (modification) sur des attributs spécifiques pour les utilisateurs d'un autre groupe d'utilisateurs. Cette forme de règle de contrôle d'accès se limite à la modification des valeurs d'un sous-ensemble d'attributs que vous spécifiez dans une règle de délégation ; elle ne permet pas d'ajouter ou de supprimer des entrées entières ni de contrôler des attributs non spécifiés.

Les règles de délégation accordent des autorisations aux groupes d'utilisateurs existants dans IdM. Vous pouvez utiliser la délégation pour, par exemple, permettre au groupe d'utilisateurs **managers** de gérer certains attributs des utilisateurs du groupe d'utilisateurs **employees**.

25.2. CRÉATION D'UNE RÈGLE DE DÉLÉGATION À L'AIDE DE L'INTERFACE CLI DE L'IDM

Cette section décrit comment créer une règle de délégation à l'aide de la CLI IdM.

Conditions préalables

- Vous êtes connecté en tant que membre du groupe **admins**.

Procédure

- Entrez la commande **ipa delegation-add**. Spécifiez les options suivantes :
  - **--group**: le groupe qui *is being granted permissions* aux entrées des utilisateurs dans le groupe d'utilisateurs.
  - **--membergroup**: le groupe *whose entries can be edited* par les membres du groupe de délégation.

- **--permissions**: si les utilisateurs auront le droit de voir les attributs donnés (*read*) et d'ajouter ou de modifier les attributs donnés (*write*). Si vous ne spécifiez pas de permissions, seule la permission *write* sera ajoutée.
- **--attrs**: les attributs que les utilisateurs du groupe de membres sont autorisés à voir ou à modifier.

Par exemple :

```
$ ipa delegation-add "basic manager attributes" --permissions=read --permissions=write --attrs=businesscategory --attrs=departmentnumber --attrs=employeetype --attrs=employeenumber --group=managers --membergroup=employees
----------------------------------------
Added delegation "basic manager attributes"
----------------------------------------
  Delegation name: basic manager attributes
  Permissions: read, write
  Attributes: businesscategory, departmentnumber, employeetype, employeenumber
  Member user group: employees
  User group: managers
```

25.3. VISUALISATION DES RÈGLES DE DÉLÉGATION EXISTANTES À L'AIDE DE LA CLI IDM

Cette section décrit comment visualiser les règles de délégation existantes à l'aide de la CLI IdM.

Conditions préalables

- Vous êtes connecté en tant que membre du groupe **admins**.

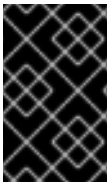
Procédure

- Entrez la commande **ipa delegation-find**:

```
$ ipa delegation-find
--------------------
1 delegation matched
--------------------
  Delegation name: basic manager attributes
  Permissions: read, write
  Attributes: businesscategory, departmentnumber, employeenumber, employeetype
  Member user group: employees
  User group: managers
----------------------------
Number of entries returned 1
----------------------------
```

25.4. MODIFICATION D'UNE RÈGLE DE DÉLÉGATION À L'AIDE DE LA CLI IDM

Cette section décrit comment modifier une règle de délégation existante à l'aide de l'interface CLI de l'IdM.



IMPORTANT

L'option **--attrs** écrase la liste précédente des attributs pris en charge. Il faut donc toujours inclure la liste complète des attributs ainsi que les nouveaux attributs. Ceci s'applique également à l'option **--permissions**.

Conditions préalables

- Vous êtes connecté en tant que membre du groupe **admins**.

Procédure

- Entrez la commande **ipa delegation-mod** avec les modifications souhaitées. Par exemple, pour ajouter l'attribut **displayname** à la règle d'exemple **basic manager attributes**:

```
$ ipa delegation-mod "basic manager attributes" --attrs=businesscategory --attrs=departmentnumber --attrs=employeetype --attrs=employeenumber --attrs=displayname
---------------------------------------------
Modified delegation "basic manager attributes"
---------------------------------------------
  Delegation name: basic manager attributes
  Permissions: read, write
  Attributes: businesscategory, departmentnumber, employeetype, employeenumber, displayname
  Member user group: employees
  User group: managers
```

25.5. SUPPRESSION D'UNE RÈGLE DE DÉLÉGATION À L'AIDE DE LA CLI IDM

Cette section décrit comment supprimer une règle de délégation existante à l'aide de l'interface CLI de l'IdM.

Conditions préalables

- Vous êtes connecté en tant que membre du groupe **admins**.

Procédure

- Entrez la commande **ipa delegation-del**.
- Lorsque vous y êtes invité, saisissez le nom de la règle de délégation que vous souhaitez supprimer :

```
$ ipa delegation-del
Delegation name: basic manager attributes
---------------------------------------------
Deleted delegation "basic manager attributes"
---------------------------------------------
```

CHAPITRE 26. DÉLÉGATION DE PERMISSIONS À DES GROUPES D'UTILISATEURS POUR GÉRER LES UTILISATEURS À L'AIDE DE L'INTERFACE WEB IDM

La délégation est l'une des méthodes de contrôle d'accès dans IdM, avec les règles en libre-service et le contrôle d'accès basé sur les rôles (RBAC). Vous pouvez utiliser la délégation pour attribuer des autorisations à un groupe d'utilisateurs afin de gérer des entrées pour un autre groupe d'utilisateurs.

Cette section couvre les sujets suivants :

- Règles de délégation
- Création d'une règle de délégation à l'aide de l'interface Web IdM
- Visualisation des règles de délégation existantes à l'aide de l'interface Web IdM
- Modifier une règle de délégation à l'aide de l'interface Web IdM
- Suppression d'une règle de délégation à l'aide de l'interface Web IdM

26.1. RÈGLES DE DÉLÉGATION

Vous pouvez déléguer des autorisations à des groupes d'utilisateurs pour gérer les utilisateurs en créant **delegation rules**.

Les règles de délégation permettent à un groupe d'utilisateurs spécifique d'effectuer des opérations d'écriture (modification) sur des attributs spécifiques pour les utilisateurs d'un autre groupe d'utilisateurs. Cette forme de règle de contrôle d'accès se limite à la modification des valeurs d'un sous-ensemble d'attributs que vous spécifiez dans une règle de délégation ; elle ne permet pas d'ajouter ou de supprimer des entrées entières ni de contrôler des attributs non spécifiés.

Les règles de délégation accordent des autorisations aux groupes d'utilisateurs existants dans IdM. Vous pouvez utiliser la délégation pour, par exemple, permettre au groupe d'utilisateurs **managers** de gérer certains attributs des utilisateurs du groupe d'utilisateurs **employees**.

26.2. CRÉATION D'UNE RÈGLE DE DÉLÉGATION À L'AIDE DE L'INTERFACE WEB IDM

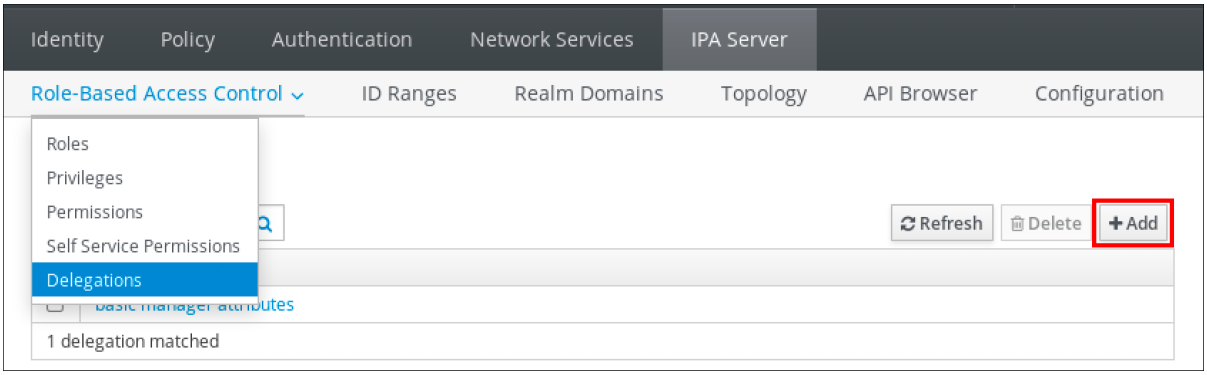
Cette section décrit comment créer une règle de délégation à l'aide de l'interface Web IdM.

Conditions préalables

- Vous êtes connecté à l'interface Web IdM en tant que membre du groupe **admins**.

Procédure

1. Dans le menu **IPA Server**, cliquez sur **Role-Based Access Control → Delegations**.
2. Cliquez sur **Add**.



3. Dans la fenêtre **Add delegation**, procédez comme suit :

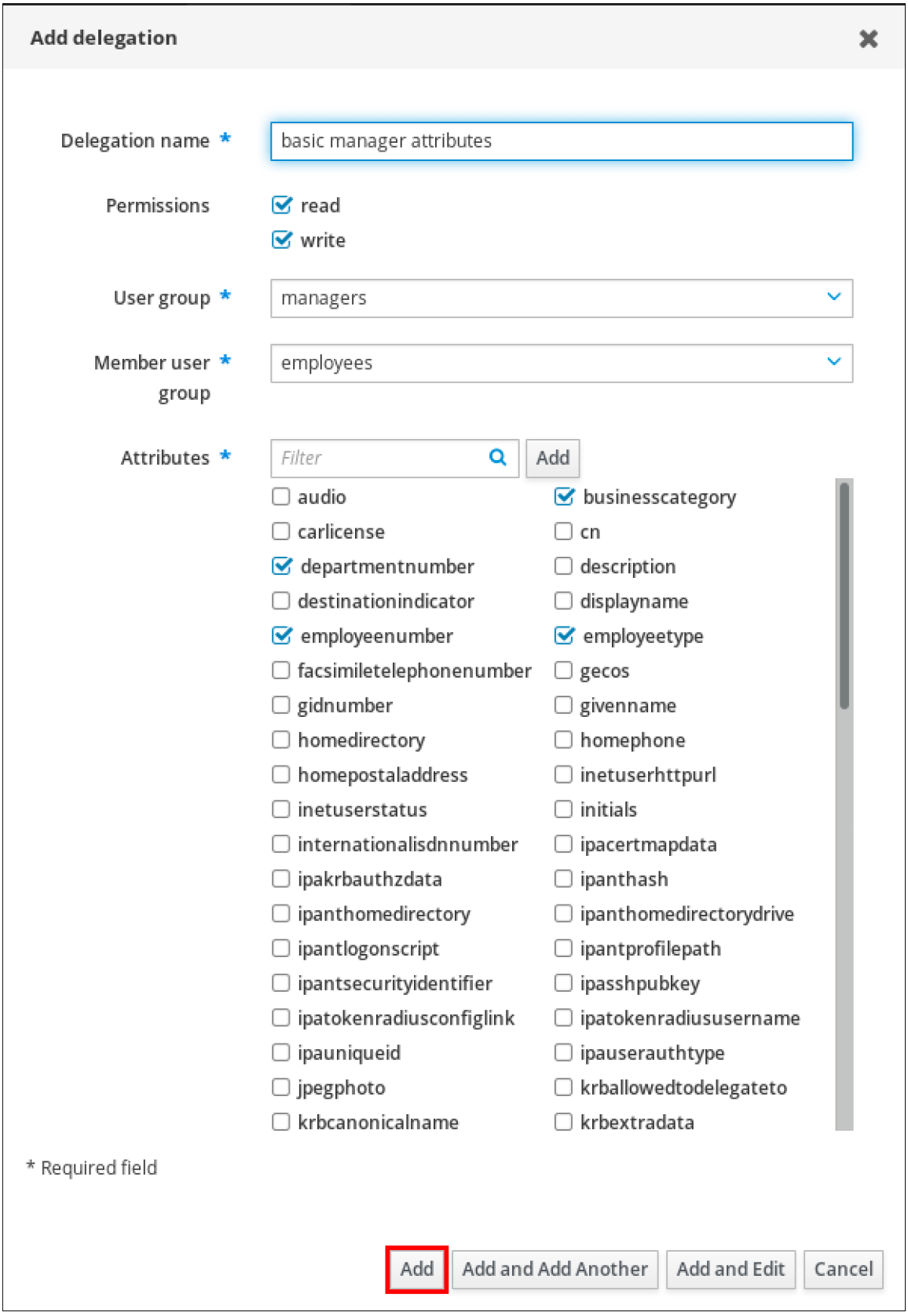
   a. Nommez la nouvelle règle de délégation.

   b. Définissez les autorisations en cochant les cases qui indiquent si les utilisateurs auront le droit de voir les attributs donnés (*read*) et d'ajouter ou de modifier les attributs donnés (*write*).

   c. Dans le menu déroulant Groupe d'utilisateurs, sélectionnez le groupe *who is being granted permissions* pour afficher ou modifier les entrées des utilisateurs du groupe membre.

   d. Dans le menu déroulant **Member user group**, sélectionnez le groupe *whose entries can be edited* par les membres du groupe de délégation.

   e. Dans la case attributs, cochez les cases des attributs auxquels vous souhaitez accorder des autorisations.



f. Cliquez sur le bouton **Add** pour enregistrer la nouvelle règle de délégation.

26.3. VISUALISATION DES RÈGLES DE DÉLÉGATION EXISTANTES À L'AIDE DE L'INTERFACE WEB IDM

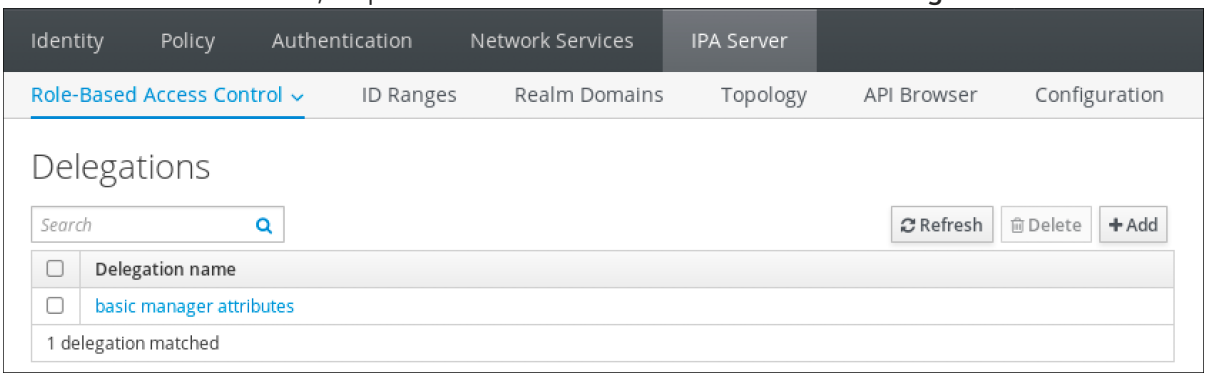
Cette section décrit comment visualiser les règles de délégation existantes à l'aide de l'interface Web IdM.

Conditions préalables

- Vous êtes connecté à l'interface Web IdM en tant que membre du groupe **admins**.

Procédure

- Dans le menu **IPA Server**, cliquez sur **Role-Based Access Control → Delegations**.



26.4. MODIFIER UNE RÈGLE DE DÉLÉGATION À L'AIDE DE L'INTERFACE WEB IDM

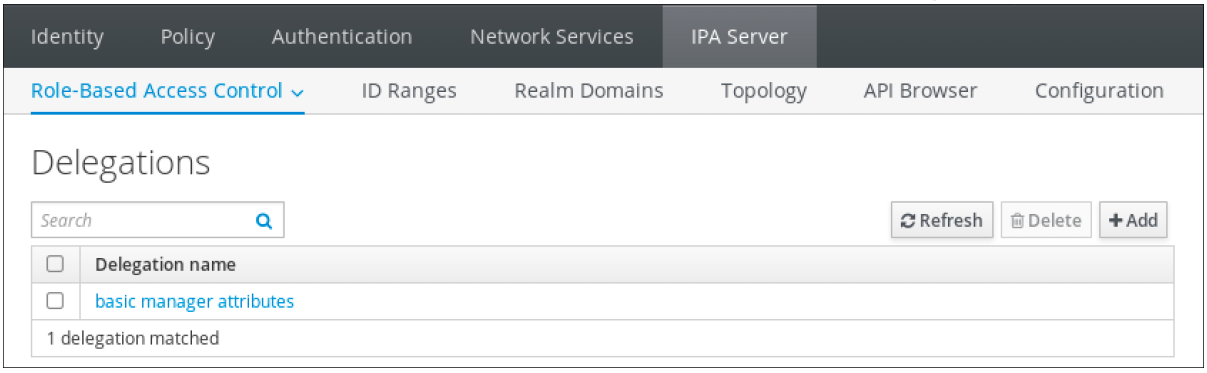
Cette section décrit comment modifier une règle de délégation existante à l'aide de l'interface Web IdM.

Conditions préalables

- Vous êtes connecté à l'interface Web IdM en tant que membre du groupe **admins**.

Procédure

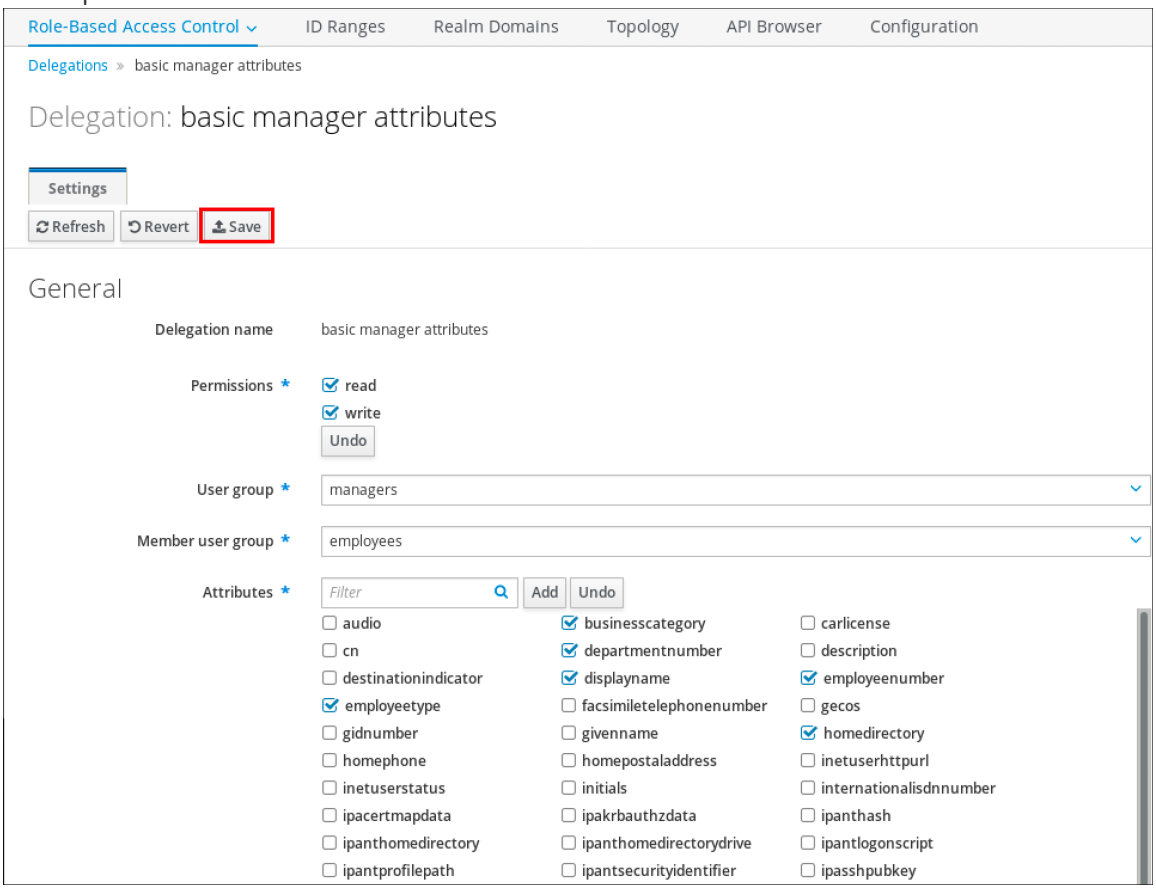
1. Dans le menu **IPA Server**, cliquez sur **Role-Based Access Control → Delegations**.



2. Cliquez sur la règle que vous souhaitez modifier.
3. Effectuez les modifications souhaitées :
   - Modifier le nom de la règle.
   - Modifiez les autorisations accordées en cochant les cases qui indiquent si les utilisateurs auront le droit de voir les attributs donnés (*read*) et d'ajouter ou de modifier les attributs donnés (*write*).
   - Dans le menu déroulant Groupe d'utilisateurs, sélectionnez le groupe *who is being granted permissions* pour afficher ou modifier les entrées des utilisateurs du groupe membre.

- Dans le menu déroulant **Member user group**, sélectionnez le groupe *whose entries can be edited* par les membres du groupe de délégation.

- Dans la case attributs, cochez les cases des attributs auxquels vous souhaitez accorder des autorisations. Pour supprimer les autorisations d'un attribut, décochez la case correspondante.



- Cliquez sur le bouton **Save** pour enregistrer les modifications.

26.5. SUPPRESSION D'UNE RÈGLE DE DÉLÉGATION À L'AIDE DE L'INTERFACE WEB IDM

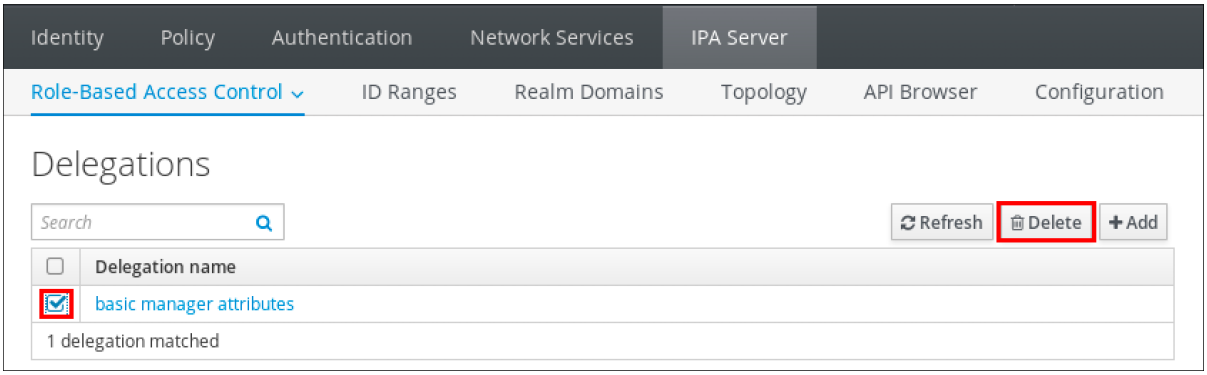
Cette section décrit comment supprimer une règle de délégation existante à l'aide de l'interface Web IdM.

Conditions préalables

- Vous êtes connecté à l'interface Web IdM en tant que membre du groupe **admins**.

Procédure

1. Dans le menu **IPA Server**, cliquez sur **Role-Based Access Control → Delegations**.
2. Cochez la case en regard de la règle que vous souhaitez supprimer.
3. Cliquez sur **Delete**.



4. Cliquez sur **Delete** pour confirmer.

CHAPITRE 27. DÉLÉGUER DES PERMISSIONS À DES GROUPES D'UTILISATEURS POUR GÉRER LES UTILISATEURS À L'AIDE DE PLAYBOOKS ANSIBLE

La délégation est l'une des méthodes de contrôle d'accès dans IdM, avec les règles en libre-service et le contrôle d'accès basé sur les rôles (RBAC). Vous pouvez utiliser la délégation pour attribuer des autorisations à un groupe d'utilisateurs afin de gérer des entrées pour un autre groupe d'utilisateurs.

Cette section couvre les sujets suivants :

- Règles de délégation
- Création du fichier d'inventaire Ansible pour IdM
- Utiliser Ansible pour s'assurer qu'une règle de délégation est présente
- Utiliser Ansible pour s'assurer qu'une règle de délégation est absente
- Utiliser Ansible pour s'assurer qu'une règle de délégation possède des attributs spécifiques
- Utiliser Ansible pour s'assurer qu'une règle de délégation n'a pas d'attributs spécifiques

27.1. RÈGLES DE DÉLÉGATION

Vous pouvez déléguer des autorisations à des groupes d'utilisateurs pour gérer les utilisateurs en créant **delegation rules**.

Les règles de délégation permettent à un groupe d'utilisateurs spécifique d'effectuer des opérations d'écriture (modification) sur des attributs spécifiques pour les utilisateurs d'un autre groupe d'utilisateurs. Cette forme de règle de contrôle d'accès se limite à la modification des valeurs d'un sous-ensemble d'attributs que vous spécifiez dans une règle de délégation ; elle ne permet pas d'ajouter ou de supprimer des entrées entières ni de contrôler des attributs non spécifiés.

Les règles de délégation accordent des autorisations aux groupes d'utilisateurs existants dans IdM. Vous pouvez utiliser la délégation pour, par exemple, permettre au groupe d'utilisateurs **managers** de gérer certains attributs des utilisateurs du groupe d'utilisateurs **employees**.

27.2. CRÉATION D'UN FICHIER D'INVENTAIRE ANSIBLE POUR IDM

Lorsque vous travaillez avec Ansible, il est bon de créer, dans votre répertoire personnel, un sous-répertoire dédié aux playbooks Ansible que vous copiez et adaptez à partir des sous-répertoires **/usr/share/doc/ansible-freeipa/*** et **/usr/share/doc/rhel-system-roles/***. Cette pratique présente les avantages suivants :

- Vous pouvez retrouver tous vos playbooks en un seul endroit.
- Vous pouvez exécuter vos playbooks sans invoquer les privilèges de **root**.

Procédure

1. Créez un répertoire pour votre configuration Ansible et vos playbooks dans votre répertoire personnel :

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

2. Allez dans le répertoire **~/MyPlaybooks/**:

```
$ cd ~/MyPlaybooks
```

3. Créez le fichier **~/MyPlaybooks/ansible.cfg** avec le contenu suivant :

```
[defaults]
inventory = /home/<username>/MyPlaybooks/inventory

[privilege_escalation]
become=True
```

4. Créez le fichier **~/MyPlaybooks/inventory** avec le contenu suivant :

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

[us]
replica.idm.example.com

[ipaserver:children]
eu
us
```

Cette configuration définit deux groupes d'hôtes, **eu** et **us**, pour les hôtes de ces sites. En outre, cette configuration définit le groupe d'hôtes **ipaserver**, qui contient tous les hôtes des groupes **eu** et **us**.

27.3. UTILISER ANSIBLE POUR S'ASSURER QU'UNE RÈGLE DE DÉLÉGATION EST PRÉSENTE

La procédure suivante décrit comment utiliser un playbook Ansible pour définir les privilèges d'une nouvelle règle de délégation IdM et assurer sa présence. Dans l'exemple, la nouvelle règle de délégation **basic manager attributes** accorde au groupe **managers** la possibilité de lire et d'écrire les attributs suivants pour les membres du groupe **employees**:

- **businesscategory**
- **departmentnumber**
- **employeenumber**
- **employeetype**

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.

- L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
- L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

Procédure

1. Naviguez jusqu'au répertoire ***~/MyPlaybooks/*** répertoire :

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

2. Faites une copie du fichier **delegation-present.yml** situé dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/delegation/**:

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/delegation/delegation-present.yml
delegation-present-copy.yml
```

3. Ouvrez le fichier **delegation-present-copy.yml** Ansible playbook pour l'éditer.
4. Adaptez le fichier en définissant les variables suivantes dans la section **ipadelegation** task :
   - Définissez la variable **ipaadmin_password** avec le mot de passe de l'administrateur IdM.
   - Attribuez à la variable **name** le nom de la nouvelle règle de délégation.
   - Attribuez à la variable **permission** une liste de permissions à accorder, séparées par des virgules : **read** et **write**.
   - Définissez la variable **attribute** avec une liste d'attributs que le groupe d'utilisateurs délégué peut gérer : **businesscategory**, **departmentnumber**, **employeenumber**, et **employeetype**.
   - Définissez la variable **group** avec le nom du groupe auquel on donne accès à la visualisation ou à la modification des attributs.
   - Définissez la variable **membergroup** avec le nom du groupe dont les attributs peuvent être visualisés ou modifiés.

   Il s'agit du fichier playbook Ansible modifié pour l'exemple actuel :

```
---
- name: Playbook to manage a delegation rule
  hosts: ipaserver

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Ensure delegation "basic manager attributes" is present
    ipadelegation:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: "basic manager attributes"
      permission: read, write
      attribute:
      - businesscategory
      - departmentnumber
      - employeenumber
```

```
    - employeetype
    group: managers
    membergroup: employees
```

5. Enregistrer le fichier.

6. Exécutez le playbook Ansible. Spécifiez le fichier du livre de jeu, le fichier contenant le mot de passe protégeant le fichier **secret.yml** et le fichier d'inventaire :

```
$ ansible-playbook --vault-password-file=password_file -v -i ~/MyPlaybooks/inventory
delegation-present-copy.yml
```

Ressources supplémentaires

- Voir règles de délégation.
- Voir le fichier **README-delegation.md** dans le répertoire **/usr/share/doc/ansible-freeipa/**.
- Voir les exemples de playbooks dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/ipadelegation**.

27.4. UTILISER ANSIBLE POUR S'ASSURER QU'UNE RÈGLE DE DÉLÉGATION EST ABSENTE

La procédure suivante décrit comment utiliser un playbook Ansible pour s'assurer qu'une règle de délégation spécifiée est absente de votre configuration IdM. L'exemple ci-dessous décrit comment s'assurer que la règle de délégation personnalisée **basic manager attributes** n'existe pas dans IdM.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire **~/MyPlaybooks/** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

Procédure

1. Naviguez jusqu'au répertoire **~/MyPlaybooks/** répertoire :

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

2. Faites une copie du fichier **delegation-absent.yml** situé dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/delegation/**:

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/delegation/delegation-present.yml
delegation-absent-copy.yml
```

3. Ouvrez le fichier **delegation-absent-copy.yml** Ansible playbook pour l'éditer.
4. Adaptez le fichier en définissant les variables suivantes dans la section **ipadelegation** task :
   - Définissez la variable **ipaadmin_password** avec le mot de passe de l'administrateur IdM.
   - Attribuez à la variable **name** le nom de la règle de délégation.
   - Fixer la variable **state** à **absent**.

   Il s'agit du fichier playbook Ansible modifié pour l'exemple actuel :

```
---
- name: Delegation absent
  hosts: ipaserver

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Ensure delegation "basic manager attributes" is absent
    ipadelegation:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: "basic manager attributes"
      state: absent
```

5. Enregistrer le fichier.
6. Exécutez le playbook Ansible. Spécifiez le fichier du livre de jeu, le fichier contenant le mot de passe protégeant le fichier **secret.yml** et le fichier d'inventaire :

```
$ ansible-playbook --vault-password-file=password_file -v -i ~/MyPlaybooks/inventory
delegation-absent-copy.yml
```

Ressources supplémentaires

- Voir règles de délégation.
- Voir le fichier **README-delegation.md** dans le répertoire **/usr/share/doc/ansible-freeipa/**.
- Voir les exemples de playbooks dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/ipadelegation**.

27.5. UTILISER ANSIBLE POUR S'ASSURER QU'UNE RÈGLE DE DÉLÉGATION POSSÈDE DES ATTRIBUTS SPÉCIFIQUES

La procédure suivante décrit comment utiliser une séquence Ansible pour s'assurer qu'une règle de délégation dispose de paramètres spécifiques. Vous pouvez utiliser ce livre de jeu pour modifier un rôle de délégation que vous avez précédemment créé. Dans l'exemple, vous vous assurez que la règle de délégation **basic manager attributes** possède uniquement l'attribut membre **departmentnumber**.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.
- La règle de délégation **basic manager attributes** existe dans IdM.

Procédure

1. Naviguez jusqu'au répertoire ***~/MyPlaybooks/*** répertoire :

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

2. Faites une copie du fichier **delegation-member-present.yml** situé dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/delegation/**:

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/delegation/delegation-member-present.yml delegation-member-present-copy.yml
```

3. Ouvrez le fichier **delegation-member-present-copy.yml** Ansible playbook pour l'éditer.
4. Adaptez le fichier en définissant les variables suivantes dans la section **ipadelegation** task :
   - Définissez la variable **ipaadmin_password** avec le mot de passe de l'administrateur IdM.
   - Attribuez à la variable **name** le nom de la règle de délégation à modifier.
   - Fixer la variable **attribute** à **departmentnumber**.
   - Fixer la variable **action** à **member**.

   Il s'agit du fichier playbook Ansible modifié pour l'exemple actuel :

```
---
- name: Delegation member present
  hosts: ipaserver

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Ensure delegation "basic manager attributes" member attribute departmentnumber
is present
    ipadelegation:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: "basic manager attributes"
```

```
    attribute:
    - departmentnumber
    action: member
```

5. Enregistrer le fichier.

6. Exécutez le playbook Ansible. Spécifiez le fichier du livre de jeu, le fichier contenant le mot de passe protégeant le fichier **secret.yml** et le fichier d'inventaire :

```
$ ansible-playbook --vault-password-file=password_file -v -i ~/MyPlaybooks/inventory
delegation-member-present-copy.yml
```

Ressources supplémentaires

- Voir règles de délégation.
- Voir le fichier **README-delegation.md** dans le répertoire **/usr/share/doc/ansible-freeipa/**.
- Voir les exemples de playbooks dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/ipadelegation**.

27.6. UTILISER ANSIBLE POUR S'ASSURER QU'UNE RÈGLE DE DÉLÉGATION N'A PAS D'ATTRIBUTS SPÉCIFIQUES

La procédure suivante décrit comment utiliser une séquence Ansible pour s'assurer qu'une règle de délégation n'a pas de paramètres spécifiques. Vous pouvez utiliser ce livre de lecture pour vous assurer qu'un rôle de délégation n'accorde pas d'accès indésirable. Dans l'exemple, vous vous assurez que la règle de délégation **basic manager attributes** n'a pas les attributs de membre **employeenumber** et **employeetype**.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.
- La règle de délégation **basic manager attributes** existe dans IdM.

Procédure

1. Naviguez jusqu'au répertoire ***~/MyPlaybooks/*** répertoire :

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

2. Faites une copie du fichier **delegation-member-absent.yml** situé dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/delegation/**:

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/delegation/delegation-member-absent.yml delegation-member-absent-copy.yml
```

3. Ouvrez le fichier **delegation-member-absent-copy.yml** Ansible playbook pour l'éditer.

4. Adaptez le fichier en définissant les variables suivantes dans la section **ipadelegation** task :

   - Définissez la variable **ipaadmin_password** avec le mot de passe de l'administrateur IdM.
   - Attribuez à la variable **name** le nom de la règle de délégation à modifier.
   - Fixez la variable **attribute** à **employeenumber** et **employeetype**.
   - Fixer la variable **action** à **member**.
   - Fixer la variable **state** à **absent**.

   Il s'agit du fichier playbook Ansible modifié pour l'exemple actuel :

```
---
- name: Delegation member absent
  hosts: ipaserver

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Ensure delegation "basic manager attributes" member attributes employeenumber and employeetype are absent
    ipadelegation:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: "basic manager attributes"
      attribute:
      - employeenumber
      - employeetype
      action: member
      state: absent
```

5. Enregistrer le fichier.

6. Exécutez le playbook Ansible. Spécifiez le fichier du livre de jeu, le fichier contenant le mot de passe protégeant le fichier **secret.yml** et le fichier d'inventaire :

```
$ ansible-playbook --vault-password-file=password_file -v -i ~/MyPlaybooks/inventory delegation-member-absent-copy.yml
```

Ressources supplémentaires

- Voir règles de délégation.
- Voir le fichier **README-delegation.md** dans le répertoire **/usr/share/doc/ansible-freeipa/**.

- Voir les exemples de playbooks dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/ipadelegation**.

CHAPITRE 28. GESTION DES CONTRÔLES D'ACCÈS BASÉS SUR LES RÔLES DANS L'IDM À L'AIDE DE LA CLI

Ce chapitre présente le contrôle d'accès basé sur les rôles dans la gestion des identités (IdM) et décrit les opérations suivantes dans l'interface de ligne de commande (CLI) :

- Gestion des autorisations
- Gestion des privilèges
- Gestion des rôles

28.1. CONTRÔLE D'ACCÈS BASÉ SUR LES RÔLES DANS L'IDM

Le contrôle d'accès basé sur les rôles (RBAC) dans l'IdM accorde un type d'autorité très différent aux utilisateurs par rapport aux contrôles d'accès en libre-service et par délégation.

Le contrôle d'accès basé sur les rôles se compose de trois parties :

- **Permissions** accorder le droit d'effectuer une tâche spécifique telle que l'ajout ou la suppression d'utilisateurs, la modification d'un groupe, l'activation de l'accès en lecture, etc.
- **Privileges** combiner les autorisations, par exemple toutes les autorisations nécessaires pour ajouter un nouvel utilisateur.
- **Roles** accorder un ensemble de privilèges à des utilisateurs, des groupes d'utilisateurs, des hôtes ou des groupes d'hôtes.

28.1.1. Permissions dans l'IdM

Les autorisations sont l'unité de niveau le plus bas du contrôle d'accès basé sur les rôles. Elles définissent les opérations ainsi que les entrées LDAP auxquelles ces opérations s'appliquent. Comparables à des blocs de construction, les autorisations peuvent être attribuées à autant de privilèges que nécessaire.
Une ou plusieurs adresses **rights** définissent les opérations autorisées :

- **write**
- **read**
- **search**
- **compare**
- **add**
- **delete**
- **all**

Ces opérations s'appliquent à trois sites de base **targets**:

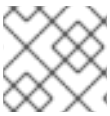
- **subtree**: un nom de domaine (DN) ; la sous-arborescence sous ce DN
- **target filter**un filtre LDAP

- **target**: DN avec des caractères génériques possibles pour spécifier les entrées

En outre, les options de commodité suivantes définissent le(s) attribut(s) correspondant(s) :

- **type**: un type d'objet (utilisateur, groupe, etc.) ; définit **subtree** et **target filter**
- **memberof**: membres d'un groupe ; fixe un **target filter**
- **targetgroup**: accorde l'accès à la modification d'un groupe spécifique (par exemple en accordant les droits de gérer l'appartenance à un groupe) ; définit une valeur de **target**

Grâce aux autorisations IdM, vous pouvez contrôler quels utilisateurs ont accès à quels objets et même à quels attributs de ces objets. L'IdM vous permet d'autoriser ou de bloquer des attributs individuels ou de modifier la visibilité totale d'une fonction IdM spécifique, telle que les utilisateurs, les groupes ou sudo, pour tous les utilisateurs anonymes, tous les utilisateurs authentifiés ou seulement un certain groupe d'utilisateurs privilégiés.
Par exemple, la flexibilité de cette approche des autorisations est utile pour un administrateur qui souhaite limiter l'accès des utilisateurs ou des groupes uniquement aux sections spécifiques auxquelles ces utilisateurs ou groupes ont besoin d'accéder et rendre les autres sections complètement cachées pour eux.



NOTE

Une autorisation ne peut pas contenir d'autres autorisations.

28.1.2. Permissions gérées par défaut

Les autorisations gérées sont des autorisations fournies par défaut avec l'IdM. Elles se comportent comme les autres autorisations créées par l'utilisateur, avec les différences suivantes :

- Vous ne pouvez pas les supprimer ni modifier leur nom, leur emplacement et leurs attributs cibles.
- Ils ont trois séries d'attributs :
  - **Default** l'utilisateur ne peut pas les modifier, car ils sont gérés par IdM
  - **Included** les attributs, qui sont des attributs supplémentaires ajoutés par l'utilisateur
  - **Excluded** les attributs, qui sont des attributs supprimés par l'utilisateur

Une autorisation gérée s'applique à tous les attributs qui figurent dans les ensembles d'attributs par défaut et inclus, mais pas dans l'ensemble exclu.

NOTE

Bien que vous ne puissiez pas supprimer une autorisation gérée, le fait de définir son type de liaison sur autorisation et de supprimer l'autorisation gérée de tous les privilèges la désactive effectivement.

Les noms de toutes les autorisations gérées commencent par **System:**, par exemple **System: Add Sudo rule** ou **System: Modify Services**. Les versions antérieures de l'IdM utilisaient un schéma différent pour les autorisations par défaut. Par exemple, l'utilisateur ne pouvait pas les supprimer et ne pouvait les attribuer qu'à des privilèges. La plupart de ces autorisations par défaut ont été transformées en autorisations gérées, mais les autorisations suivantes utilisent toujours le schéma précédent :

- Ajouter une tâche de reconstruction de l'adhésion automatique
- Ajouter des sous-enregistrements de configuration
- Ajouter des accords de réplication
- Certificat Supprimer la retenue
- Obtenir l'état des certificats de l'autorité de certification
- Plage de lecture de l'ADN
- Modifier la portée de l'ADN
- Lire la configuration des gestionnaires PassSync
- Modifier la configuration des gestionnaires PassSync
- Lire les accords de réplication
- Modifier les accords de réplication
- Supprimer les accords de réplication
- Lire la configuration de la base de données LDBM
- Demande de certificat
- Demande de certificat ignorant les listes de contrôle de l'autorité de certification
- Demander des certificats à un autre hôte
- Récupérer des certificats auprès de l'autorité de certification
- Révoquer le certificat
- Écriture de la configuration de l'IPA

NOTE

Si vous tentez de modifier une autorisation gérée à partir de la ligne de commande, le système ne vous permet pas de changer les attributs que vous ne pouvez pas modifier, la commande échoue. Si vous tentez de modifier une autorisation gérée à partir de l'interface Web, les attributs que vous ne pouvez pas modifier sont désactivés.

28.1.3. Privilèges dans l'IdM

Un privilège est un groupe de permissions applicables à un rôle.
Alors qu'une autorisation donne le droit d'effectuer une seule opération, certaines tâches de l'IdM nécessitent plusieurs autorisations pour être menées à bien. Par conséquent, un privilège combine les différentes autorisations requises pour effectuer une tâche spécifique.
Par exemple, la création d'un compte pour un nouvel utilisateur IdM nécessite les autorisations suivantes :

- Création d'une nouvelle entrée utilisateur
- Réinitialisation du mot de passe d'un utilisateur

- Ajout du nouvel utilisateur au groupe d'utilisateurs IPA par défaut

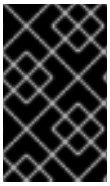
La combinaison de ces trois tâches de bas niveau en une tâche de plus haut niveau sous la forme d'un privilège personnalisé nommé, par exemple, **Add User** facilite la gestion des rôles par l'administrateur du système. L'IdM contient déjà plusieurs privilèges par défaut. Outre les utilisateurs et les groupes d'utilisateurs, des privilèges sont également attribués aux hôtes et aux groupes d'hôtes, ainsi qu'aux services de réseau. Cette pratique permet un contrôle fin des opérations effectuées par un ensemble d'utilisateurs sur un ensemble d'hôtes utilisant des services de réseau spécifiques.

NOTE

Un privilège ne peut pas contenir d'autres privilèges.

28.1.4. Rôles dans l'IdM

Un rôle est une liste de privilèges que les utilisateurs spécifiés pour ce rôle possèdent.
En effet, les autorisations permettent d'effectuer certaines tâches de bas niveau (créer une entrée utilisateur, ajouter une entrée à un groupe, etc.), tandis que les privilèges combinent une ou plusieurs de ces autorisations nécessaires à une tâche de plus haut niveau (comme la création d'un nouvel utilisateur dans un groupe donné). Les rôles regroupent les privilèges selon les besoins : par exemple, un administrateur d'utilisateurs peut ajouter, modifier et supprimer des utilisateurs.



IMPORTANT

Les rôles sont utilisés pour classer les actions autorisées. Ils ne sont pas utilisés comme outil pour mettre en œuvre la séparation des privilèges ou pour se protéger contre l'escalade des privilèges.

NOTE

Les rôles ne peuvent pas contenir d'autres rôles.

28.1.5. Rôles prédéfinis dans la gestion de l'identité

Red Hat Identity Management fournit la gamme suivante de rôles prédéfinis :

Tableau 28.1. Rôles prédéfinis dans la gestion des identités

Rôle	Privilège	Description
Administrateur des inscriptions	Inscription au programme d'accueil	Responsable de l'inscription du client ou de l'hôte
helpdesk	Modifier les utilisateurs et réinitialiser les mots de passe, Modifier l'appartenance à un groupe	Effectuer des tâches simples d'administration des utilisateurs
Spécialiste de la sécurité informatique	Administrateurs Netgroups, Administrateur HBAC, Administrateur Sudo	Responsable de la gestion de la politique de sécurité telle que les contrôles d'accès basés sur l'hôte, les règles sudo

Rôle	Privilège	Description
Spécialiste des technologies de l'information	Administrateurs d'hôtes, administrateurs de groupes d'hôtes, administrateurs de services, administrateurs de montages automatiques	Responsable de la gestion des hôtes
Architecte de sécurité	Administrateur de la délégation, Administrateurs de la réplication, Configuration de l'IPA en écriture, Administrateur de la politique des mots de passe	Responsable de la gestion de l'environnement de gestion des identités, de la création de trusts et d'accords de réplication
Administrateur des utilisateurs	Administrateurs d'utilisateurs, administrateurs de groupes, administrateurs d'utilisateurs de scène	Responsable de la création des utilisateurs et des groupes

28.2. GESTION DES AUTORISATIONS IDM DANS L'INTERFACE DE PROGRAMMATION

Cette section explique comment gérer les autorisations de gestion des identités (IdM) à l'aide de l'interface de ligne de commande (CLI).

Conditions préalables

- Privilèges d'administrateur pour la gestion de l'IdM ou le rôle **User Administrator**.
- Un ticket Kerberos actif. Pour plus de détails, voir Utilisation de kinit pour se connecter manuellement à IdM.

Procédure

1. Créez de nouvelles entrées d'autorisation à l'aide de la commande **ipa permission-add**. Par exemple, pour ajouter une autorisation nommée *dns admin*:

```
$ ipa permission-add "dns admin"
```

2. Spécifiez les propriétés de l'autorisation à l'aide des options suivantes :
   - **--bindtype** spécifie le type de règle de liaison. Cette option accepte les arguments **all**, **anonymous**, et **permission**. Le type de règle de liaison **permission** signifie que seuls les utilisateurs auxquels cette autorisation a été accordée par l'intermédiaire d'un rôle peuvent l'exercer.
     Par exemple :

```
$ ipa permission-add "dns admin" --bindtype=all
```

     Si vous n'indiquez pas **--bindtype**, la valeur par défaut est **permission**.

NOTE

Il n'est pas possible d'ajouter aux privilèges des autorisations dont le type de règle de liaison n'est pas par défaut. Il n'est pas non plus possible de définir une autorisation déjà présente dans un privilège avec un type de règle de liaison autre que celui par défaut.

- **--right** énumère les droits accordés par la permission, elle remplace l'option **--permissions** qui est obsolète. Les valeurs disponibles sont **add**, **delete**, **read**, **search**, **compare**, **write**, **all**. Vous pouvez définir plusieurs attributs en utilisant plusieurs options **--right** ou une liste séparée par des virgules à l'intérieur d'accolades. Par exemple :

```
$ ipa permission-add "dns admin" --right=read --right=write
```

```
$ ipa permission-add "dns admin" --right={read,write}
```

NOTE

add et **delete** sont des opérations d'entrée de gamme (par exemple, supprimer un utilisateur, ajouter un groupe, etc.) tandis que **read**, **search**, **compare** et **write** sont davantage des opérations de niveau attributaire : vous pouvez écrire sur **userCertificate** mais pas lire **userPassword**.

- **--attrs** donne la liste des attributs sur lesquels l'autorisation est accordée.
  Vous pouvez définir plusieurs attributs en utilisant plusieurs options **--attrs** ou en énumérant les options dans une liste séparée par des virgules à l'intérieur d'accolades. Par exemple :

```
$ ipa permission-add "dns admin" --attrs=description --attrs=automountKey
```

```
$ ipa permission-add "dns admin" --attrs={description,automountKey}
```

  Les attributs fournis avec **--attrs** doivent exister et être des attributs autorisés pour le type d'objet donné, sinon la commande échoue avec des erreurs de syntaxe de schéma.

- **--type** définit le type d'objet d'entrée auquel l'autorisation s'applique, tel que l'utilisateur, l'hôte ou le service. Chaque type possède son propre ensemble d'attributs autorisés.
  Par exemple :

```
$ ipa permission-add "manage service" --right=all --type=service --attrs=krbprincipalkey --attrs=krbprincipalname --attrs=managedby
```

- **--subtree** donne une entrée de sous-arbre ; le filtre cible alors toutes les entrées situées sous cette entrée de sous-arbre. Fournir une entrée de sous-arbre existante ; **--subtree** n'accepte pas les caractères génériques ou les noms de domaine (DN) inexistants. Inclure un DN dans le répertoire.
  Comme IdM utilise une arborescence simplifiée et plate, **--subtree** peut être utilisé pour cibler certains types d'entrées, comme les emplacements de montage automatique, qui sont des conteneurs ou des entrées parentes pour d'autres configurations. Par exemple :

```
$ ipa permission-add "manage automount locations" --subtree="ldap://ldap.example.com:389/cn=automount,dc=example,dc=com" --right=write --attrs=automountmapname --attrs=automountkey --attrs=automountInformation
```

NOTE

Les options **--type** et **--subtree** s'excluent mutuellement : vous pouvez considérer l'inclusion de filtres pour **--type** comme une simplification de **--subtree**, destinée à faciliter la vie de l'administrateur.

- **--filter** utilise un filtre LDAP pour identifier les entrées auxquelles l'autorisation s'applique. L'IdM vérifie automatiquement la validité du filtre donné. Le filtre peut être n'importe quel filtre LDAP valide, par exemple :

```
$ ipa permission-add \N "Gérer les groupes Windows" --filtre="( !
(objectclass=posixgroup))\N" -droit=écriture --attrs=description --right=write --
attrs=description
```

- **--memberof** définit le filtre cible pour les membres du groupe donné après avoir vérifié que le groupe existe. Par exemple, pour permettre aux utilisateurs disposant de cette autorisation de modifier le shell de connexion des membres du groupe des ingénieurs :

```
$ ipa permission-add ManageShell --right="write" --type=user --attr=loginshell --
memberof=engineers
```

- **--targetgroup** attribue la cible au groupe d'utilisateurs spécifié après avoir vérifié que le groupe existe. Par exemple, pour permettre à ceux qui ont l'autorisation d'écrire l'attribut membre dans le groupe d'ingénieurs (afin qu'ils puissent ajouter ou supprimer des membres) :

```
$ ipa permission-add ManageMembers --right="write" --
subtree=cn=groups,cn=accounts,dc=example,dc=test --attr=member --
targetgroup=engineers
```

- En option, vous pouvez spécifier un nom de domaine cible (DN) :
  - **--target** spécifie le DN auquel appliquer l'autorisation. Les caractères génériques sont acceptés.
  - **--targetto** spécifie le sous-arbre DN dans lequel une entrée peut être déplacée.
  - **--targetfrom** spécifie le sous-arbre DN à partir duquel une entrée peut être déplacée.

28.3. OPTIONS DE COMMANDE POUR LES AUTORISATIONS EXISTANTES

Les variantes suivantes permettent de modifier les autorisations existantes en fonction des besoins :

- Pour modifier les autorisations existantes, utilisez la commande **ipa permission-mod**. Vous pouvez utiliser les mêmes options de commande que pour l'ajout de permissions.
- Pour trouver les autorisations existantes, utilisez la commande **ipa permission-find**. Vous pouvez utiliser les mêmes options de commande que pour l'ajout de permissions.
- Pour visualiser une autorisation spécifique, utilisez la commande **ipa permission-show**. L'argument **--raw** indique l'ACI 389-ds brut qui est généré. Par exemple :

```
$ ipa permission-show <permission> --raw
```

- La commande **ipa permission-del** supprime complètement une autorisation.

Ressources supplémentaires

- Voir la page de manuel **ipa**.
- Voir la commande **ipa help**.

28.4. GESTION DES PRIVILÈGES IDM DANS L'INTERFACE DE PROGRAMMATION

Cette section explique comment gérer les privilèges de la gestion des identités (IdM) à l'aide de l'interface de ligne de commande (CLI).

Conditions préalables

- Privilèges d'administrateur pour la gestion de l'IdM ou le rôle **User Administrator**.
- Un ticket Kerberos actif. Pour plus de détails, voir le lien : Utiliser kinit pour se connecter manuellement à IdM.
- Les autorisations existantes. Pour plus d'informations sur les autorisations, voir Gestion des autorisations IdM dans l'interface de programmation.

Procédure

1. Ajoutez des entrées de privilèges à l'aide de la commande **ipa privilege-add**
   Par exemple, pour ajouter un privilège nommé *managing filesystems* avec une description :

```
$ ipa privilege-add "gestion des systèmes de fichiers" --desc="pour les systèmes de fichiers"
$ ipa privilege-add "gestion des systèmes de fichiers" --desc="pour les systèmes de fichiers"
```

2. Attribuez les autorisations requises au groupe de privilèges à l'aide de la commande **privilege-add-permission**
   Par exemple, pour ajouter les autorisations nommées *managing automount* et *managing ftp services* au privilège *managing filesystems*:

```
$ ipa privilege-add-permission "gérer les systèmes de fichiers" --permissions="gérer les
montages automatiques" --permissions="gérer les services ftp"
```

28.5. OPTIONS DE COMMANDE POUR LES PRIVILÈGES EXISTANTS

Les variantes suivantes permettent de modifier les privilèges existants en fonction des besoins :

- Pour modifier les privilèges existants, utilisez la commande **ipa privilege-mod**.
- Pour trouver les privilèges existants, utilisez la commande **ipa privilege-find**.
- Pour afficher un privilège spécifique, utilisez la commande **ipa privilege-show**.

- La commande **ipa privilege-remove-permission** supprime une ou plusieurs autorisations d'un privilège.
- La commande **ipa privilege-del** supprime complètement un privilège.

Ressources supplémentaires

- Voir la page de manuel **ipa**.
- Voir la commande **ipa help**.

28.6. GESTION DES RÔLES IDM DANS L'INTERFACE DE LIGNE DE COMMANDE

Cette section décrit comment gérer les rôles de gestion des identités (IdM) à l'aide de l'interface de ligne de commande (CLI).

Conditions préalables

- Privilèges d'administrateur pour la gestion de l'IdM ou le rôle **User Administrator**.
- Un ticket Kerberos actif. Pour plus de détails, voir Utilisation de kinit pour se connecter manuellement à IdM.
- Privilèges existants. Pour plus d'informations sur les privilèges, voir Gestion des privilèges IdM dans l'interface de programmation.

Procédure

1. Ajoutez de nouvelles entrées de rôle à l'aide de la commande **ipa role-add**:

```
$ ipa role-add --desc="User Administrator" useradmin
-----------------------
Added role "useradmin"
-----------------------
Role name: useradmin
Description: User Administrator
```

2. Ajoutez les privilèges requis au rôle à l'aide de la commande **ipa role-add-privilege**:

```
$ ipa role-add-privilege --privileges="user administrators" useradmin
Role name: useradmin
Description: User Administrator
Privileges: user administrators
----------------------------
Number of privileges added 1
----------------------------
```

3. Ajoutez les membres requis au rôle à l'aide de la commande **ipa role-add-member**. Les types de membres autorisés sont les suivants : utilisateurs, groupes, hôtes et groupes d'hôtes.
Par exemple, pour ajouter le groupe nommé *useradmins* au rôle *useradmin* précédemment créé :

```
$ ipa role-add-member --groups=useradmins useradmin
Role name: useradmin
```

```
  Description: User Administrator
  Member groups: useradmins
  Privileges: user administrators
  -------------------------
  Number of members added 1
  -------------------------
```

28.7. OPTIONS DE COMMANDE POUR LES RÔLES EXISTANTS

Les variantes suivantes permettent de modifier les rôles existants en fonction des besoins :

- Pour modifier les rôles existants, utilisez la commande **ipa role-mod**.
- Pour trouver les rôles existants, utilisez la commande **ipa role-find**.
- Pour afficher un rôle spécifique, utilisez la commande **ipa role-show**.
- Pour retirer un membre du rôle, utilisez la commande **ipa role-remove-member**.
- La commande **ipa role-remove-privilege** supprime un ou plusieurs privilèges d'un rôle.
- La commande **ipa role-del** supprime complètement un rôle.

Ressources supplémentaires

- Voir la page de manuel **ipa**
- Voir la commande **ipa help**.

CHAPITRE 29. GESTION DES CONTRÔLES D'ACCÈS BASÉS SUR LES RÔLES À L'AIDE DE L'INTERFACE WEB IDM

Ce chapitre présente le contrôle d'accès basé sur les rôles dans la gestion des identités (IdM) et décrit les opérations suivantes dans l'interface web (Web UI) :

- Gestion des autorisations
- Gestion des privilèges
- Gestion des rôles

29.1. CONTRÔLE D'ACCÈS BASÉ SUR LES RÔLES DANS L'IDM

Le contrôle d'accès basé sur les rôles (RBAC) dans l'IdM accorde un type d'autorité très différent aux utilisateurs par rapport aux contrôles d'accès en libre-service et par délégation.

Le contrôle d'accès basé sur les rôles se compose de trois parties :

- **Permissions** accorder le droit d'effectuer une tâche spécifique telle que l'ajout ou la suppression d'utilisateurs, la modification d'un groupe, l'activation de l'accès en lecture, etc.
- **Privileges** combiner les autorisations, par exemple toutes les autorisations nécessaires pour ajouter un nouvel utilisateur.
- **Roles** accorder un ensemble de privilèges à des utilisateurs, des groupes d'utilisateurs, des hôtes ou des groupes d'hôtes.

29.1.1. Permissions dans l'IdM

Les autorisations sont l'unité de niveau le plus bas du contrôle d'accès basé sur les rôles. Elles définissent les opérations ainsi que les entrées LDAP auxquelles ces opérations s'appliquent. Comparables à des blocs de construction, les autorisations peuvent être attribuées à autant de privilèges que nécessaire.
Une ou plusieurs adresses **rights** définissent les opérations autorisées :

- **write**
- **read**
- **search**
- **compare**
- **add**
- **delete**
- **all**

Ces opérations s'appliquent à trois sites de base **targets**:

- **subtree**: un nom de domaine (DN) ; la sous-arborescence sous ce DN
- **target filter**un filtre LDAP

- **target**: DN avec des caractères génériques possibles pour spécifier les entrées

En outre, les options de commodité suivantes définissent le(s) attribut(s) correspondant(s) :

- **type**: un type d'objet (utilisateur, groupe, etc.) ; définit **subtree** et **target filter**
- **memberof**: membres d'un groupe ; fixe un **target filter**
- **targetgroup**: accorde l'accès à la modification d'un groupe spécifique (par exemple en accordant les droits de gérer l'appartenance à un groupe) ; définit une valeur de **target**

Grâce aux autorisations IdM, vous pouvez contrôler quels utilisateurs ont accès à quels objets et même à quels attributs de ces objets. L'IdM vous permet d'autoriser ou de bloquer des attributs individuels ou de modifier la visibilité totale d'une fonction IdM spécifique, telle que les utilisateurs, les groupes ou sudo, pour tous les utilisateurs anonymes, tous les utilisateurs authentifiés ou seulement un certain groupe d'utilisateurs privilégiés.
Par exemple, la flexibilité de cette approche des autorisations est utile pour un administrateur qui souhaite limiter l'accès des utilisateurs ou des groupes uniquement aux sections spécifiques auxquelles ces utilisateurs ou groupes ont besoin d'accéder et rendre les autres sections complètement cachées pour eux.

NOTE

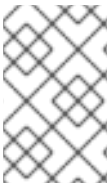
Une autorisation ne peut pas contenir d'autres autorisations.

29.1.2. Permissions gérées par défaut

Les autorisations gérées sont des autorisations fournies par défaut avec l'IdM. Elles se comportent comme les autres autorisations créées par l'utilisateur, avec les différences suivantes :

- Vous ne pouvez pas les supprimer ni modifier leur nom, leur emplacement et leurs attributs cibles.
- Ils ont trois séries d'attributs :
  - **Default** l'utilisateur ne peut pas les modifier, car ils sont gérés par IdM
  - **Included** les attributs, qui sont des attributs supplémentaires ajoutés par l'utilisateur
  - **Excluded** les attributs, qui sont des attributs supprimés par l'utilisateur

Une autorisation gérée s'applique à tous les attributs qui figurent dans les ensembles d'attributs par défaut et inclus, mais pas dans l'ensemble exclu.



NOTE

Bien que vous ne puissiez pas supprimer une autorisation gérée, le fait de définir son type de liaison sur autorisation et de supprimer l'autorisation gérée de tous les privilèges la désactive effectivement.

Les noms de toutes les autorisations gérées commencent par **System:**, par exemple **System: Add Sudo rule** ou **System: Modify Services**. Les versions antérieures de l'IdM utilisaient un schéma différent pour les autorisations par défaut. Par exemple, l'utilisateur ne pouvait pas les supprimer et ne pouvait les attribuer qu'à des privilèges. La plupart de ces autorisations par défaut ont été transformées en autorisations gérées, mais les autorisations suivantes utilisent toujours le schéma précédent :

- Ajouter une tâche de reconstruction de l'adhésion automatique
- Ajouter des sous-enregistrements de configuration
- Ajouter des accords de réplication
- Certificat Supprimer la retenue
- Obtenir l'état des certificats de l'autorité de certification
- Plage de lecture de l'ADN
- Modifier la portée de l'ADN
- Lire la configuration des gestionnaires PassSync
- Modifier la configuration des gestionnaires PassSync
- Lire les accords de réplication
- Modifier les accords de réplication
- Supprimer les accords de réplication
- Lire la configuration de la base de données LDBM
- Demande de certificat
- Demande de certificat ignorant les listes de contrôle de l'autorité de certification
- Demander des certificats à un autre hôte
- Récupérer des certificats auprès de l'autorité de certification
- Révoquer le certificat
- Écriture de la configuration de l'IPA

NOTE

Si vous tentez de modifier une autorisation gérée à partir de la ligne de commande, le système ne vous permet pas de changer les attributs que vous ne pouvez pas modifier, la commande échoue. Si vous tentez de modifier une autorisation gérée à partir de l'interface Web, les attributs que vous ne pouvez pas modifier sont désactivés.

29.1.3. Privilèges dans l'IdM

Un privilège est un groupe de permissions applicables à un rôle.
Alors qu'une autorisation donne le droit d'effectuer une seule opération, certaines tâches de l'IdM nécessitent plusieurs autorisations pour être menées à bien. Par conséquent, un privilège combine les différentes autorisations requises pour effectuer une tâche spécifique.
Par exemple, la création d'un compte pour un nouvel utilisateur IdM nécessite les autorisations suivantes :

- Création d'une nouvelle entrée utilisateur
- Réinitialisation du mot de passe d'un utilisateur

- Ajout du nouvel utilisateur au groupe d'utilisateurs IPA par défaut

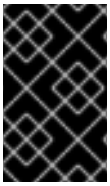
La combinaison de ces trois tâches de bas niveau en une tâche de plus haut niveau sous la forme d'un privilège personnalisé nommé, par exemple, **Add User** facilite la gestion des rôles par l'administrateur du système. L'IdM contient déjà plusieurs privilèges par défaut. Outre les utilisateurs et les groupes d'utilisateurs, des privilèges sont également attribués aux hôtes et aux groupes d'hôtes, ainsi qu'aux services de réseau. Cette pratique permet un contrôle fin des opérations effectuées par un ensemble d'utilisateurs sur un ensemble d'hôtes utilisant des services de réseau spécifiques.

NOTE

Un privilège ne peut pas contenir d'autres privilèges.

29.1.4. Rôles dans l'IdM

Un rôle est une liste de privilèges que les utilisateurs spécifiés pour ce rôle possèdent.
En effet, les autorisations permettent d'effectuer certaines tâches de bas niveau (créer une entrée utilisateur, ajouter une entrée à un groupe, etc.), tandis que les privilèges combinent une ou plusieurs de ces autorisations nécessaires à une tâche de plus haut niveau (comme la création d'un nouvel utilisateur dans un groupe donné). Les rôles regroupent les privilèges selon les besoins : par exemple, un administrateur d'utilisateurs peut ajouter, modifier et supprimer des utilisateurs.



IMPORTANT

Les rôles sont utilisés pour classer les actions autorisées. Ils ne sont pas utilisés comme outil pour mettre en œuvre la séparation des privilèges ou pour se protéger contre l'escalade des privilèges.

NOTE

Les rôles ne peuvent pas contenir d'autres rôles.

29.1.5. Rôles prédéfinis dans la gestion de l'identité

Red Hat Identity Management fournit la gamme suivante de rôles prédéfinis :

Tableau 29.1. Rôles prédéfinis dans la gestion des identités

Rôle	Privilège	Description
Administrateur des inscriptions	Inscription au programme d'accueil	Responsable de l'inscription du client ou de l'hôte
helpdesk	Modifier les utilisateurs et réinitialiser les mots de passe, Modifier l'appartenance à un groupe	Effectuer des tâches simples d'administration des utilisateurs
Spécialiste de la sécurité informatique	Administrateurs Netgroups, Administrateur HBAC, Administrateur Sudo	Responsable de la gestion de la politique de sécurité telle que les contrôles d'accès basés sur l'hôte, les règles sudo

Rôle	Privilège	Description
Spécialiste des technologies de l'information	Administrateurs d'hôtes, administrateurs de groupes d'hôtes, administrateurs de services, administrateurs de montages automatiques	Responsable de la gestion des hôtes
Architecte de sécurité	Administrateur de la délégation, Administrateurs de la réplication, Configuration de l'IPA en écriture, Administrateur de la politique des mots de passe	Responsable de la gestion de l'environnement de gestion des identités, de la création de trusts et d'accords de réplication
Administrateur des utilisateurs	Administrateurs d'utilisateurs, administrateurs de groupes, administrateurs d'utilisateurs de scène	Responsable de la création des utilisateurs et des groupes

29.2. GESTION DES AUTORISATIONS DANS L'INTERFACE WEB IDM

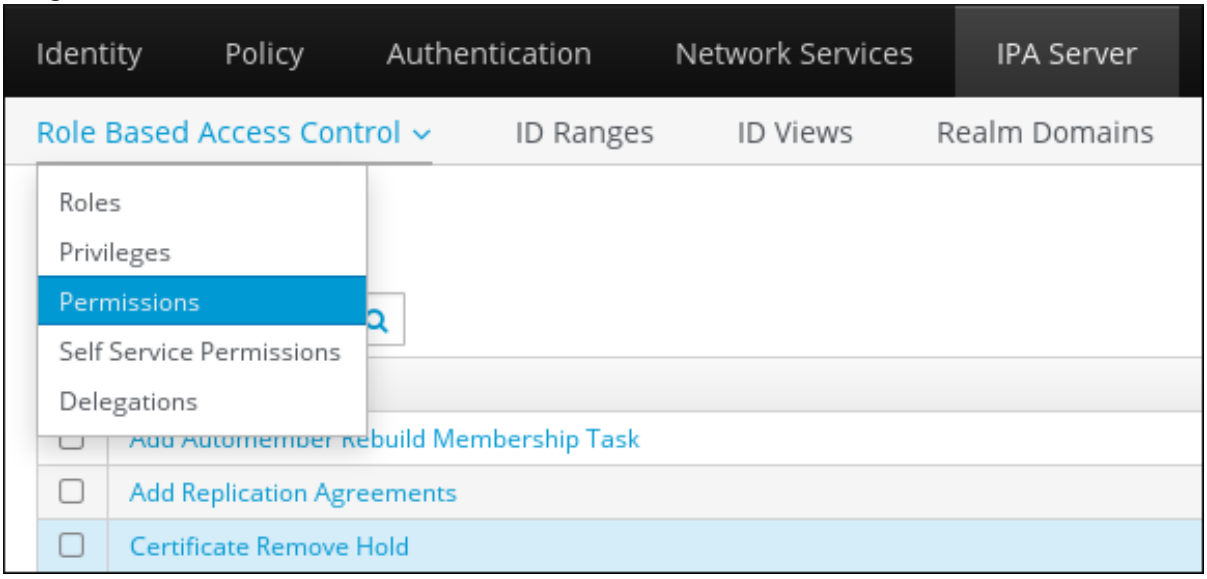
Cette section décrit comment gérer les autorisations dans la gestion des identités (IdM) à l'aide de l'interface web (IdM Web UI).

Conditions préalables

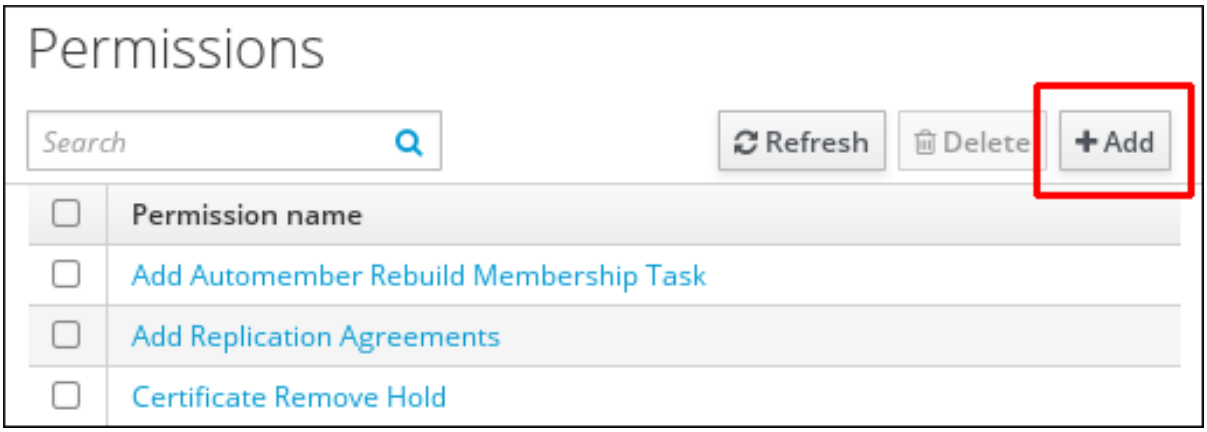
- Privilèges d'administrateur pour la gestion de l'IdM ou le rôle **User Administrator**.
- Vous êtes connecté à l'interface Web IdM. Pour plus de détails, voir Accès à l'IdM Web UI dans un navigateur web.

Procédure

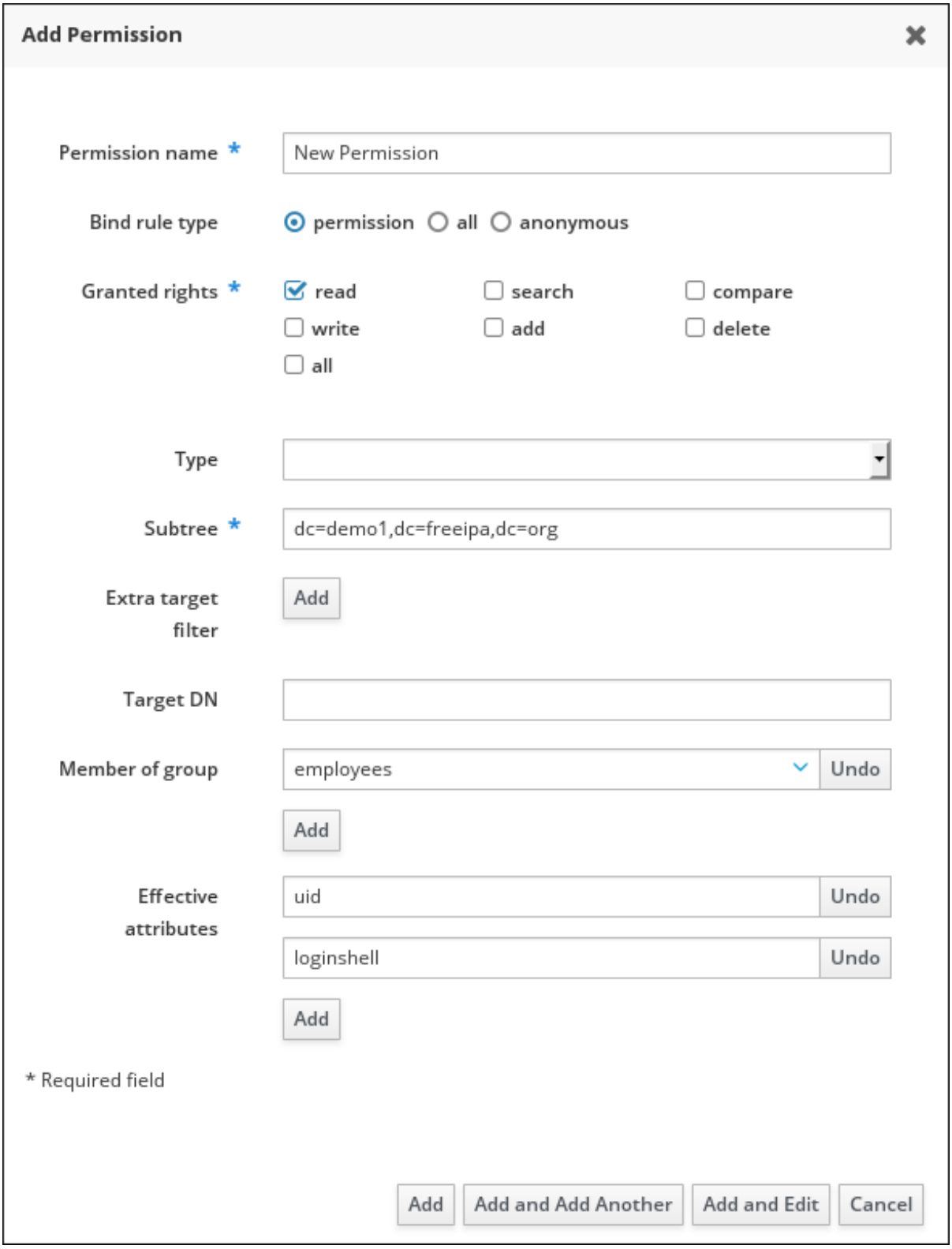
1. Pour ajouter une nouvelle autorisation, ouvrez le sous-menu **Role-Based Access Control** dans l'onglet **IPA Server** et sélectionnez **Permissions**:



2. La liste des autorisations s'ouvre : Cliquez sur le bouton **Add** en haut de la liste des autorisations :



3. Le formulaire **Add Permission** s'ouvre. Spécifiez le nom de la nouvelle autorisation et définissez ses propriétés en conséquence :



4. Sélectionnez le type de règle de liaison approprié :

   - **permission** est le type d'autorisation par défaut, accordant l'accès par le biais de privilèges et de rôles
   - **all** spécifie que l'autorisation s'applique à tous les utilisateurs authentifiés
   - **anonymous** spécifie que l'autorisation s'applique à tous les utilisateurs, y compris les utilisateurs non authentifiés

NOTE

Il n'est pas possible d'ajouter aux privilèges des autorisations dont le type de règle de liaison n'est pas par défaut. Il n'est pas non plus possible de définir une autorisation déjà présente dans un privilège avec un type de règle de liaison autre que celui par défaut.

5. Choisissez les droits à accorder avec cette permission dans **Granted rights**.

6. Définir la méthode d'identification des entrées cibles pour l'autorisation :

   - **Type** spécifie un type d'entrée, tel que l'utilisateur, l'hôte ou le service. Si vous choisissez une valeur pour le paramètre **Type**, une liste de tous les attributs possibles qui seront accessibles via cet ACI pour ce type d'entrée apparaît sous **Effective Attributes**. La définition de **Type** attribue à **Subtree** et **Target DN** l'une des valeurs prédéfinies.

   - **Subtree** (obligatoire) spécifie une entrée de sous-arbre ; chaque entrée située sous cette entrée de sous-arbre est alors ciblée. Fournissez une entrée de sous-arbre existante, car **Subtree** n'accepte pas les caractères génériques ou les noms de domaine (DN) inexistants. Par exemple : **cn=automount,dc=example,dc=com**

   - **Extra target filter** utilise un filtre LDAP pour identifier les entrées auxquelles l'autorisation s'applique. Le filtre peut être n'importe quel filtre LDAP valide, par exemple : **(!(objectclass=posixgroup))**
     L'IdM vérifie automatiquement la validité du filtre donné. Si vous saisissez un filtre non valide, l'IdM vous en avertit lorsque vous tentez d'enregistrer l'autorisation.

   - **Target DN** spécifie le nom de domaine (DN) et accepte les caractères génériques. Par exemple : **uid=*,cn=users,cn=accounts,dc=com**

   - **Member of group** définit le filtre cible pour les membres du groupe donné. Après avoir spécifié les paramètres du filtre et cliqué sur **Add**, l'IdM valide le filtre. Si tous les paramètres de permission sont corrects, l'IdM effectuera la recherche. Si certains des paramètres de permission sont incorrects, l'IdM affichera un message vous informant des paramètres incorrects.

7. Ajouter des attributs à l'autorisation :

   - Si vous avez défini **Type**, choisissez **Effective attributes** dans la liste des attributs ACI disponibles.

   - Si vous n'avez pas utilisé **Type**, ajoutez les attributs manuellement en les écrivant dans le champ **Effective attributes**. Ajoutez un seul attribut à la fois ; pour ajouter plusieurs attributs, cliquez sur **Add** pour ajouter un autre champ de saisie.

IMPORTANT

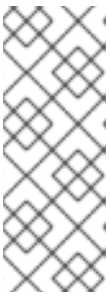
Si vous ne définissez aucun attribut pour la permission, celle-ci inclut tous les attributs par défaut.

8. Terminez l'ajout des autorisations à l'aide des boutons **Add** au bas du formulaire :

   - Cliquez sur le bouton **Add** pour enregistrer l'autorisation et revenir à la liste des autorisations.

- Vous pouvez également enregistrer l'autorisation et continuer à ajouter des autorisations supplémentaires dans le même formulaire en cliquant sur le bouton **Add and Add another**
- Le bouton **Add and Edit** vous permet d'enregistrer et de continuer à modifier l'autorisation nouvellement créée.

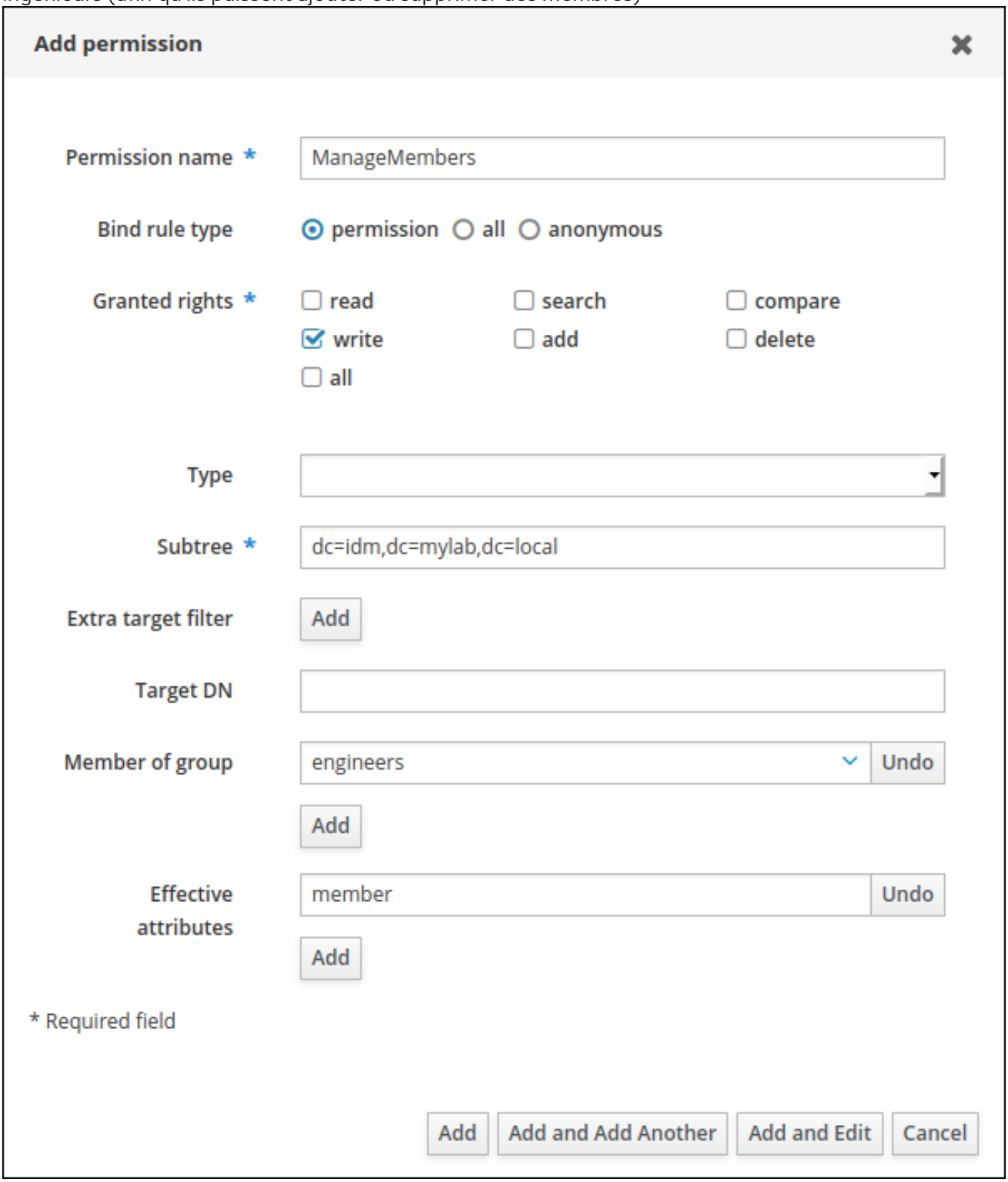
9. *Optional.* Vous pouvez également modifier les propriétés d'une autorisation existante en cliquant sur son nom dans la liste des autorisations pour afficher la page **Permission settings**.
10. *Optional.* Si vous devez supprimer une autorisation existante, cliquez sur le bouton **Delete** après avoir coché la case située à côté de son nom dans la liste, pour afficher la boîte de dialogue **Remove permissions**.



NOTE

Les opérations sur les autorisations gérées par défaut sont limitées : les attributs que vous ne pouvez pas modifier sont désactivés dans l'interface Web IdM et vous ne pouvez pas supprimer complètement les autorisations gérées. Toutefois, vous pouvez désactiver efficacement une autorisation gérée dont le type de liaison est défini comme une autorisation, en supprimant l'autorisation gérée de tous les privilèges.

Par exemple, pour permettre à ceux qui ont la permission d'écrire l'attribut membre dans le groupe des ingénieurs (afin qu'ils puissent ajouter ou supprimer des membres)



29.3. GESTION DES PRIVILÈGES DANS L'INTERFACE WEB DE L'IDM

Cette section décrit comment gérer les privilèges dans IdM à l'aide de l'interface web (IdM Web UI).

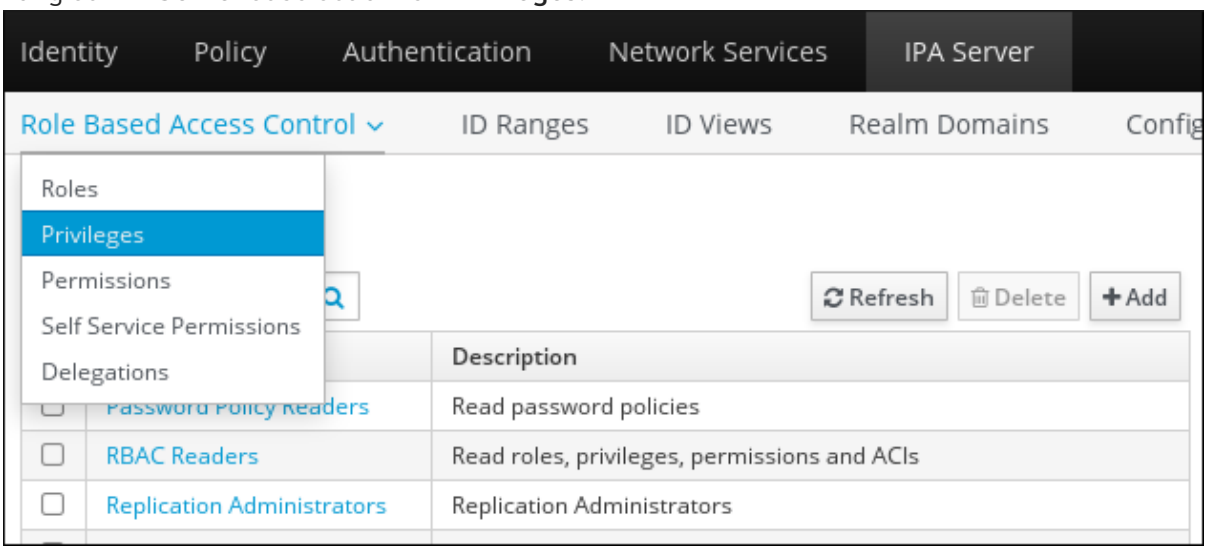
Conditions préalables

- Privilèges d'administrateur pour la gestion de l'IdM ou le rôle **User Administrator**.
- Vous êtes connecté à l'interface Web IdM. Pour plus de détails, voir Accès à l'IdM Web UI dans un navigateur web.

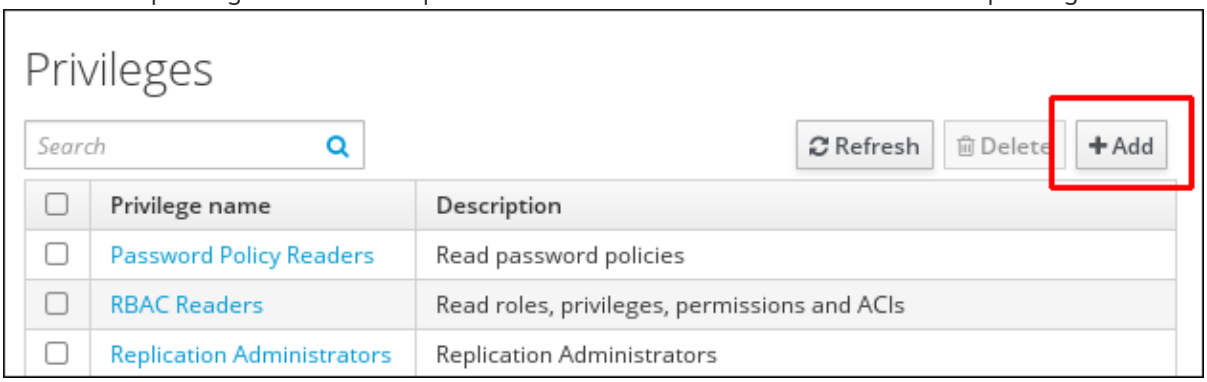
- Les autorisations existantes. Pour plus de détails sur les autorisations, voir Gérer les autorisations dans l'interface Web IdM.

Procédure

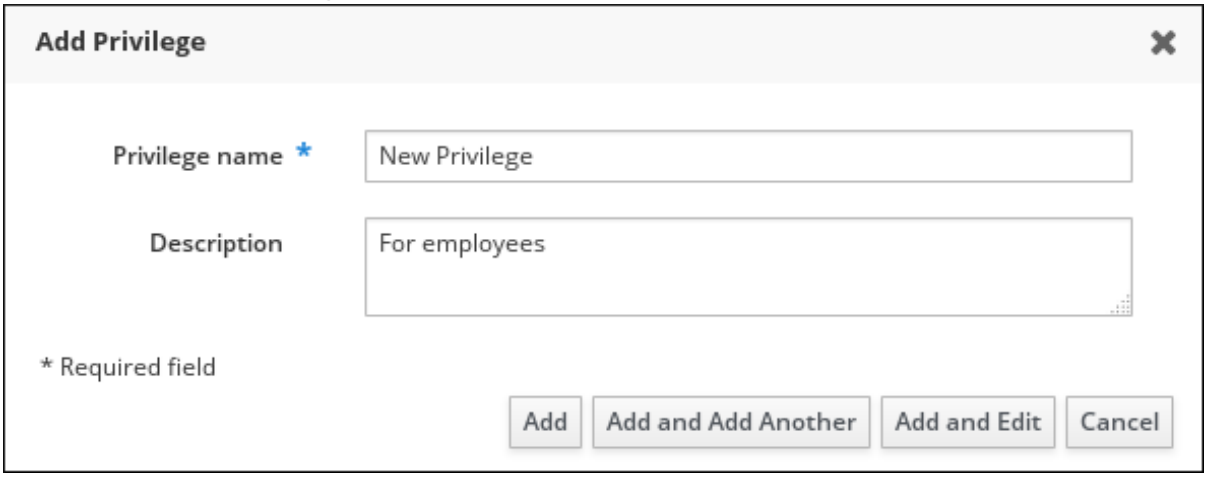
1. Pour ajouter un nouveau privilège, ouvrez le sous-menu **Role-Based Access Control** dans l'onglet **IPA Server** et sélectionnez **Privileges**:



2. La liste des privilèges s'ouvre. Cliquez sur le bouton **Add** en haut de la liste des privilèges :



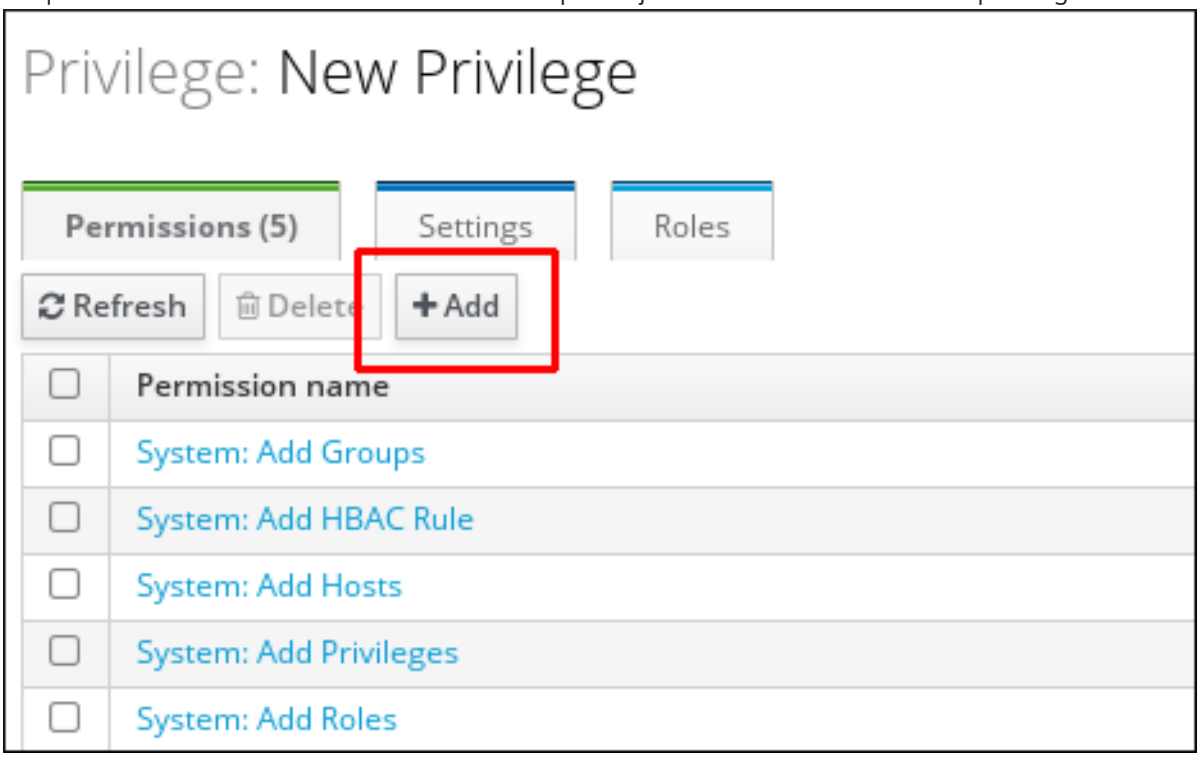
3. Le formulaire **Add Privilege** s'ouvre. Saisissez le nom et une description du privilège :



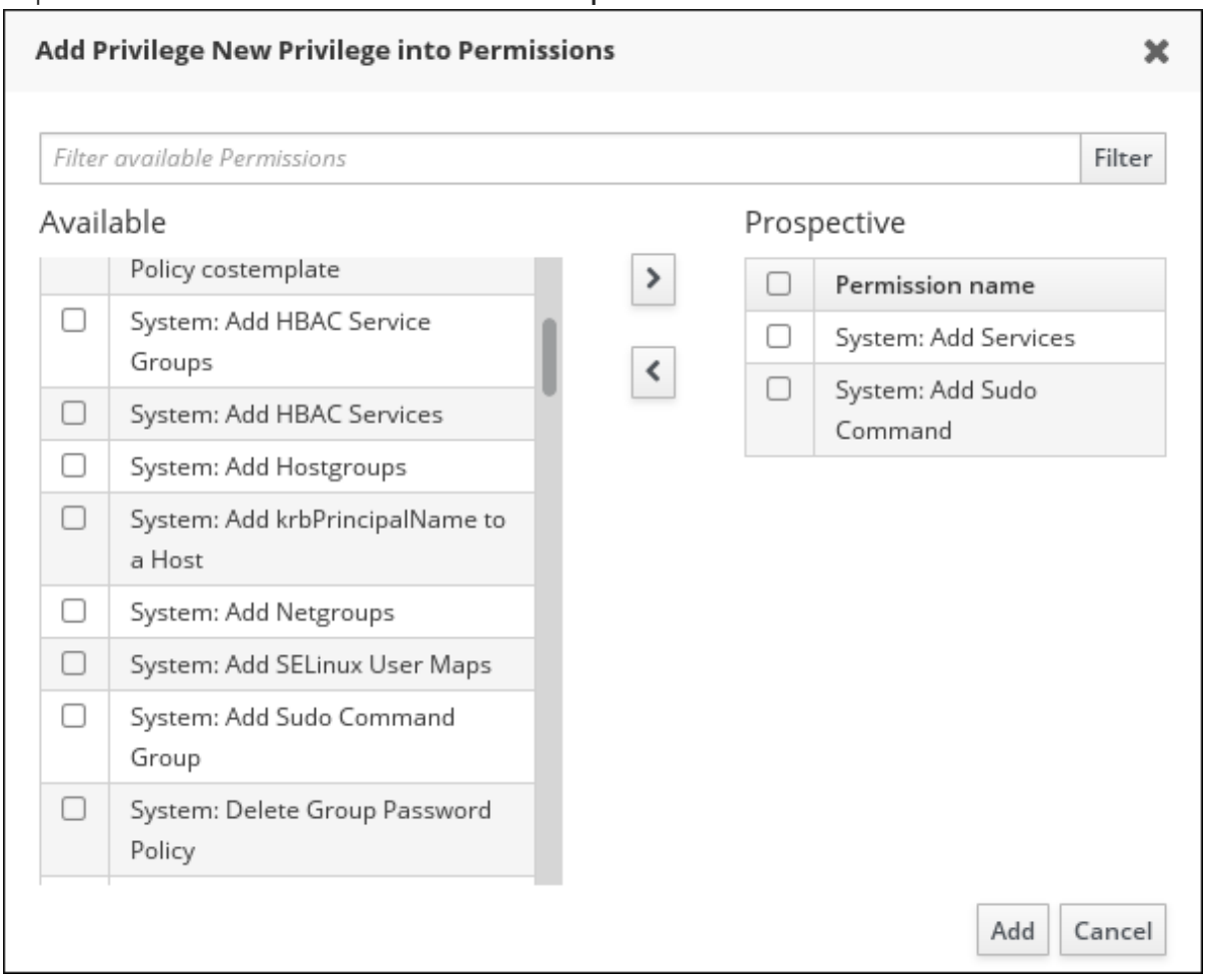
4. Cliquez sur le bouton **Add and Edit** pour enregistrer le nouveau privilège et passer à la page de configuration des privilèges pour ajouter des autorisations.

5. Modifiez les propriétés des privilèges en cliquant sur le nom du privilège dans la liste des privilèges. La page de configuration des privilèges s'ouvre.

6. L'onglet **Permissions** affiche une liste des autorisations incluses dans le privilège sélectionné. Cliquez sur le bouton **Add** en haut de la liste pour ajouter des autorisations au privilège :



7. Cochez la case en regard du nom de chaque autorisation à ajouter et utilisez le bouton > pour déplacer les autorisations dans la colonne **Prospective**:



8. Confirmez en cliquant sur le bouton **Add**.

9. *Optional.* Si vous devez supprimer des autorisations, cliquez sur le bouton **Delete** après avoir coché la case située à côté de l'autorisation concernée : la boîte de dialogue **Remove privileges from permissions** s'ouvre.

10. *Optional.* Si vous devez supprimer un privilège existant, cliquez sur le bouton **Delete** après avoir coché la case située à côté de son nom dans la liste : la boîte de dialogue **Remove privileges** s'ouvre.

29.4. GESTION DES RÔLES DANS L'INTERFACE WEB IDM

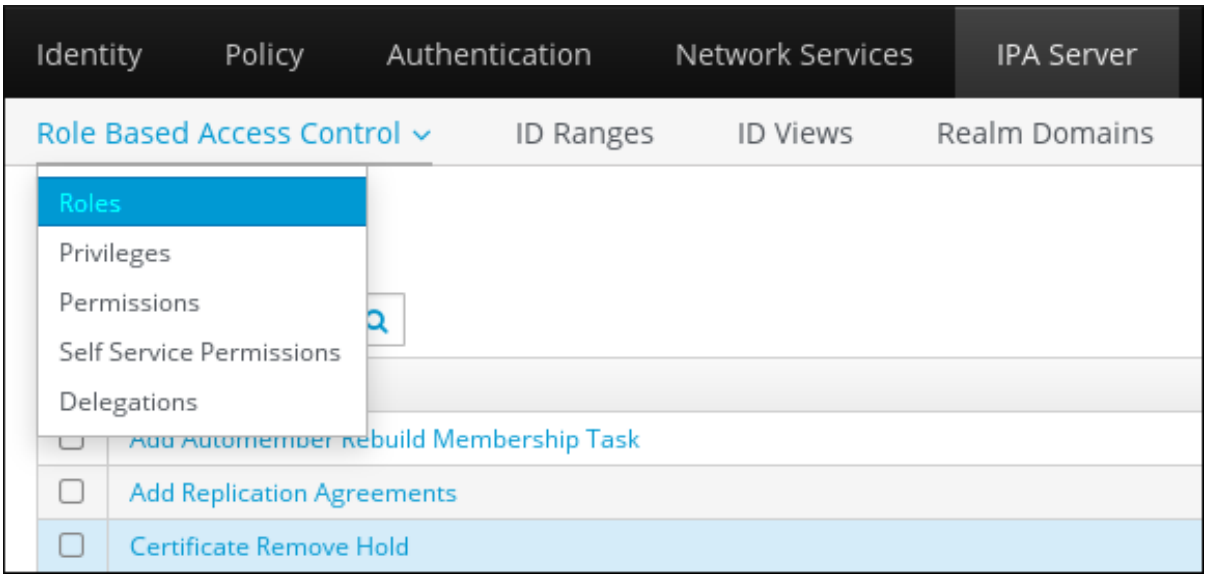
Cette section décrit comment gérer les rôles dans la gestion des identités (IdM) à l'aide de l'interface web (IdM Web UI).

Conditions préalables

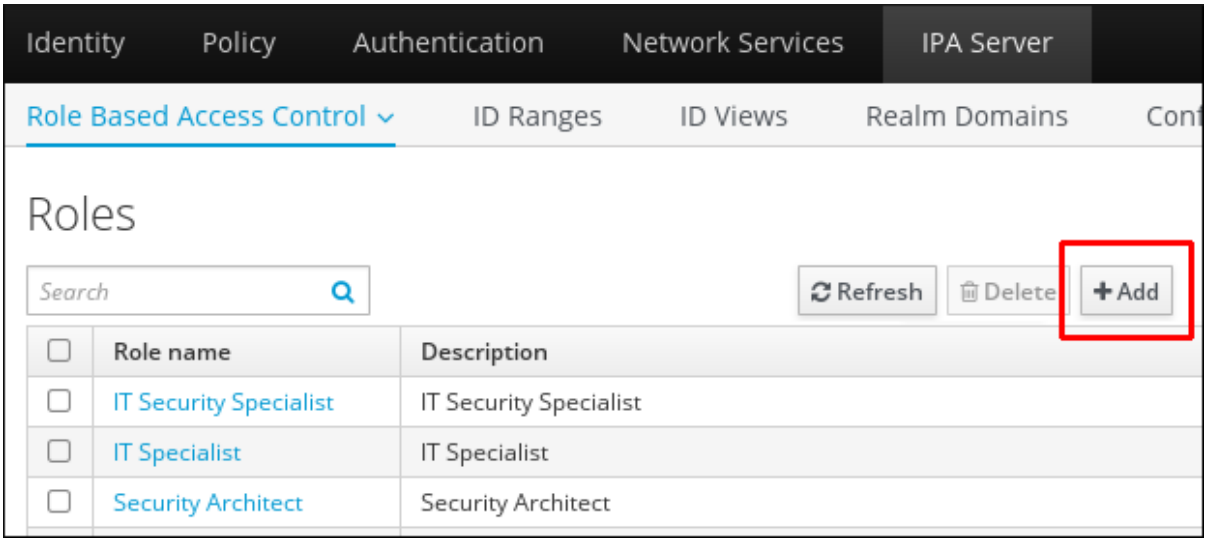
- Privilèges d'administrateur pour la gestion de l'IdM ou le rôle **User Administrator**.
- Vous êtes connecté à l'interface Web IdM. Pour plus de détails, voir Accès à l'IdM Web UI dans un navigateur web.
- Privilèges existants. Pour plus d'informations sur les privilèges, voir Gestion des privilèges dans l'interface Web IdM.

Procédure

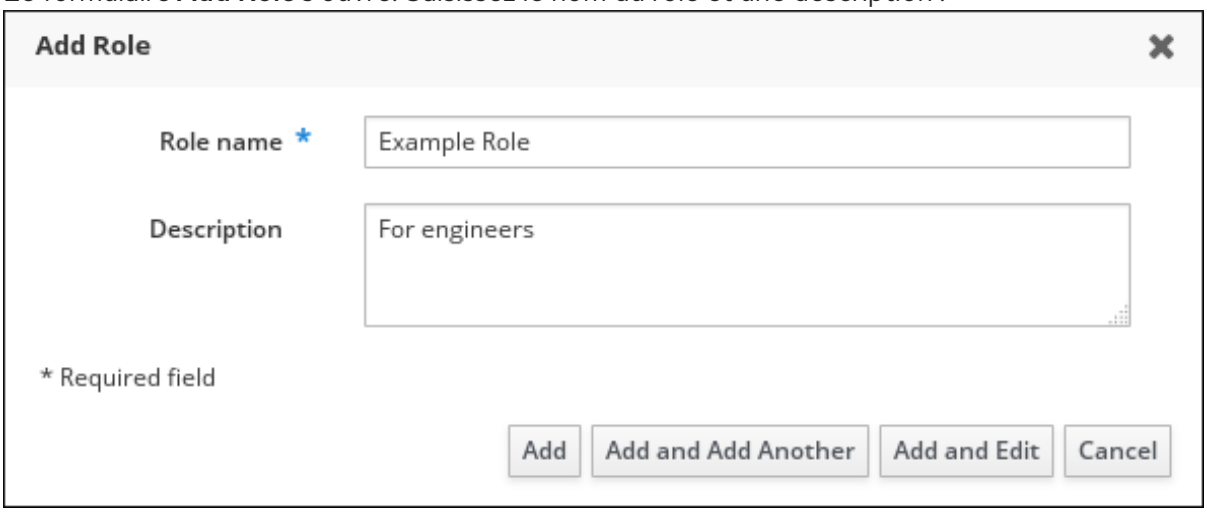
1. Pour ajouter un nouveau rôle, ouvrez le sous-menu **Role-Based Access Control** dans l'onglet **IPA Server** et sélectionnez **Roles**:



2. La liste des rôles s'ouvre. Cliquez sur le bouton **Add** en haut de la liste des instructions de contrôle d'accès basées sur les rôles.



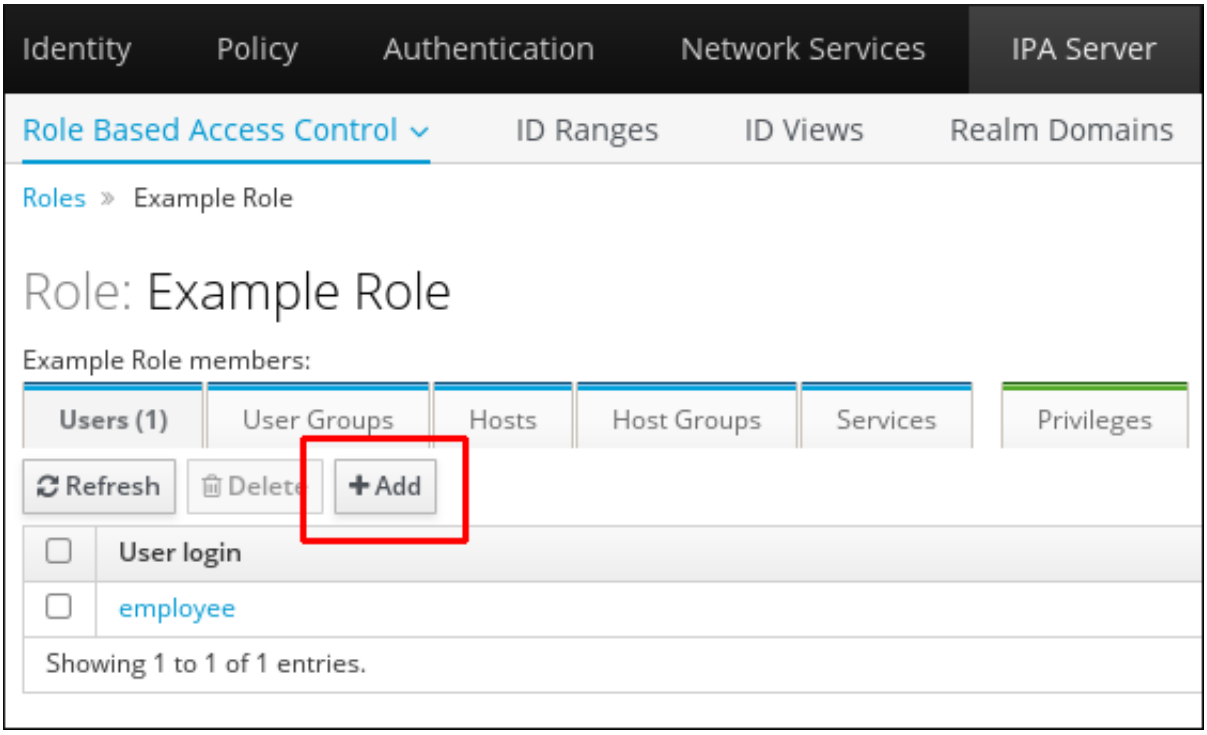
3. Le formulaire **Add Role** s'ouvre. Saisissez le nom du rôle et une description :



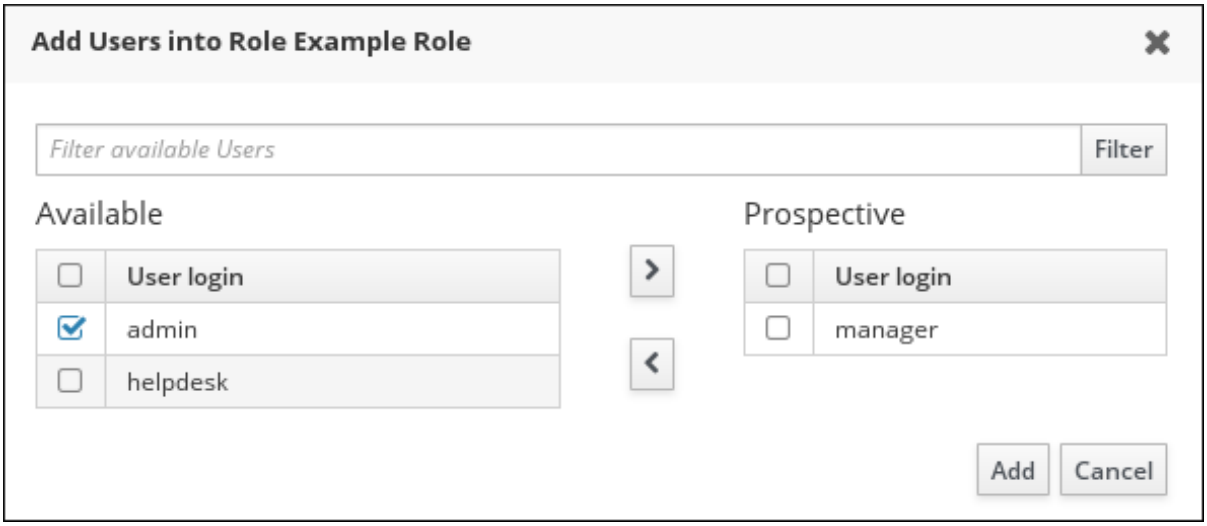
4. Cliquez sur le bouton **Add and Edit** pour enregistrer le nouveau rôle et accéder à la page de configuration des rôles pour ajouter des privilèges et des utilisateurs.

5. Modifiez les propriétés des rôles en cliquant sur le nom du rôle dans la liste des rôles. La page de configuration des rôles s'ouvre.

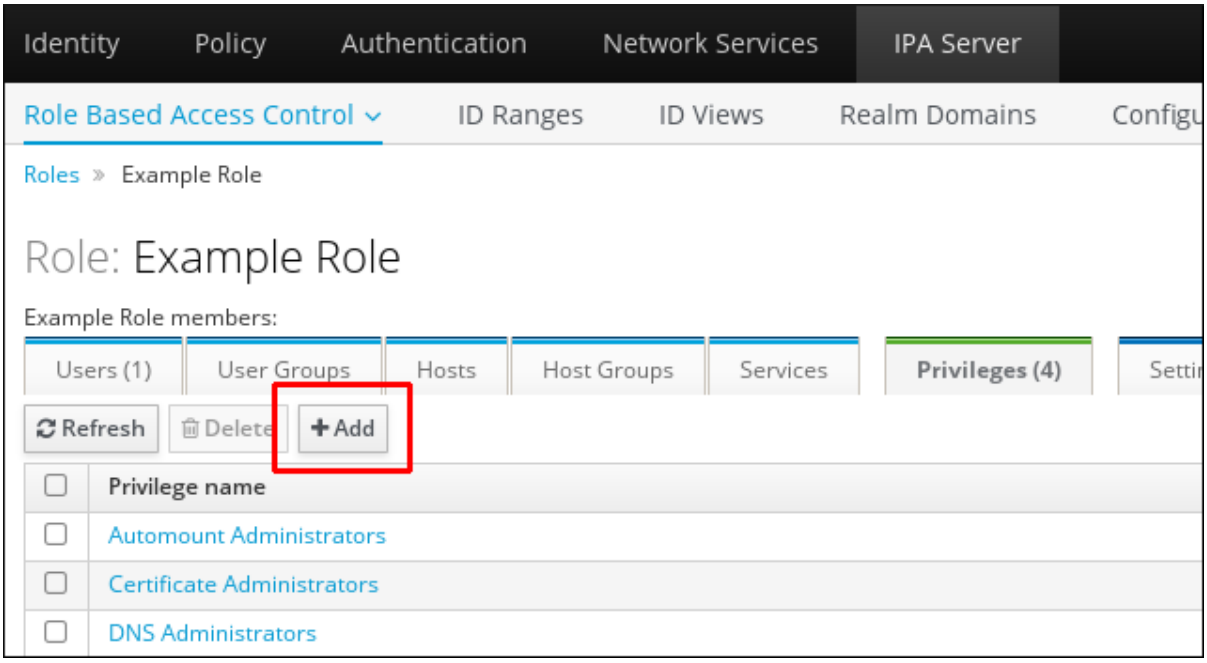
6. Ajoutez des membres en utilisant les onglets **Users**, **Users Groups**, **Hosts**, **Host Groups** ou **Services**, en cliquant sur le bouton **Add** en haut de la ou des listes concernées.



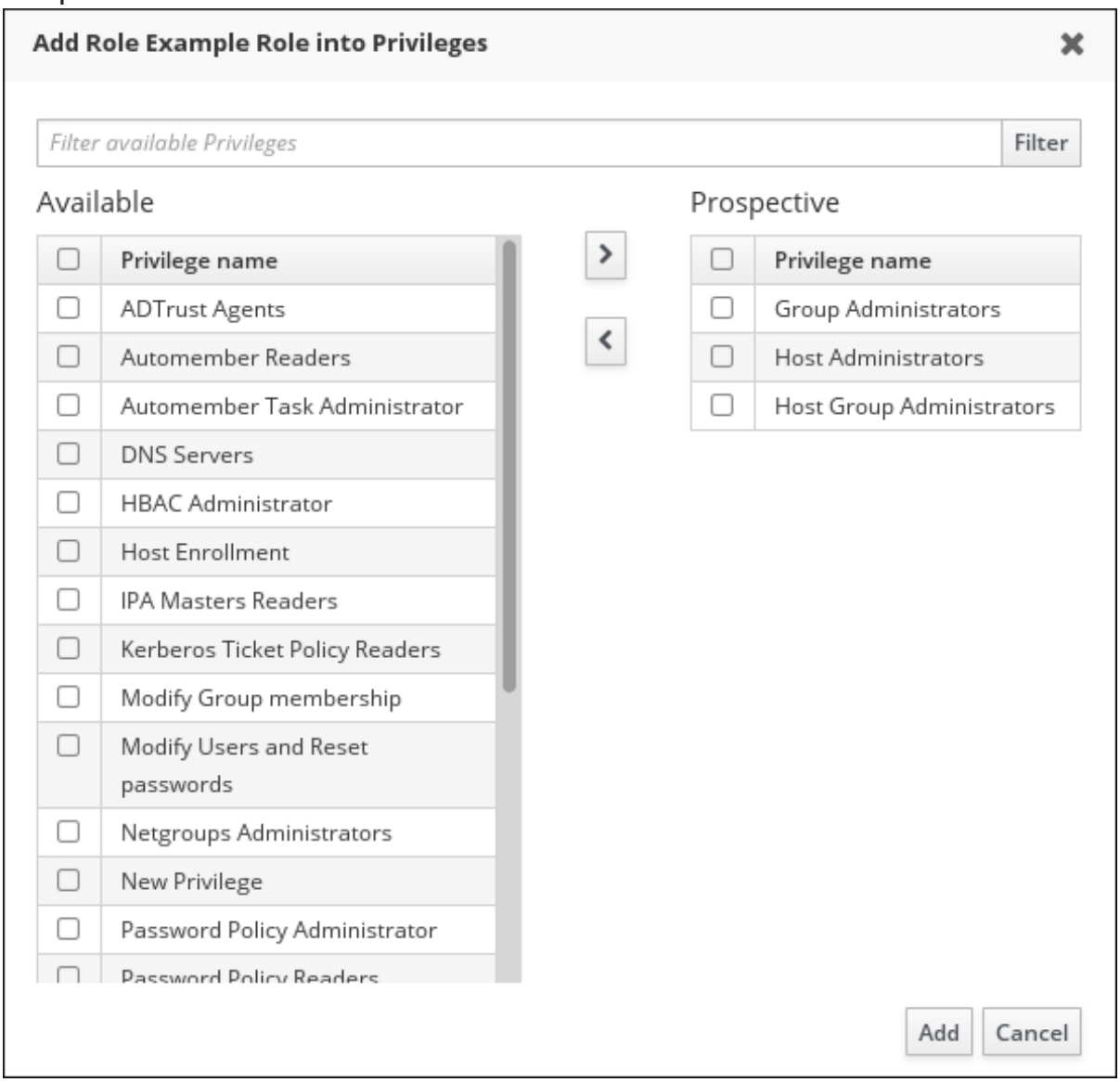
7. Dans la fenêtre qui s'ouvre, sélectionnez les membres de gauche et utilisez le bouton > pour les déplacer dans la colonne **Prospective**.



8. En haut de l'onglet **Privileges**, cliquez sur **Add**.



9. Sélectionnez les privilèges sur la gauche et utilisez le bouton > pour les déplacer dans la colonne **Prospective**.



10. Cliquez sur le bouton **Add** pour enregistrer.

11. *Optional.* Si vous devez supprimer des privilèges ou des membres d'un rôle, cliquez sur le bouton **Delete** après avoir coché la case située à côté du nom de l'entité à supprimer. Une boîte de dialogue s'ouvre.

12. *Optional.* Si vous devez supprimer un rôle existant, cliquez sur le bouton **Delete** après avoir coché la case située à côté de son nom dans la liste, pour afficher la boîte de dialogue **Remove roles**.

CHAPITRE 30. PRÉPARATION DE L'ENVIRONNEMENT POUR LA GESTION DE L'IDM À L'AIDE DES PLAYBOOKS ANSIBLE

En tant qu'administrateur système gérant la gestion des identités (IdM), lorsque vous travaillez avec Red Hat Ansible Engine, il est recommandé de procéder comme suit :

- Créez un sous-répertoire dédié aux playbooks Ansible dans votre répertoire personnel, par exemple **~/MyPlaybooks**.
- Copiez et adaptez les exemples de playbooks Ansible des répertoires et sous-répertoires **/usr/share/doc/ansible-freeipa/*** et **/usr/share/doc/rhel-system-roles/*** dans votre répertoire **~/MyPlaybooks**.
- Incluez votre fichier d'inventaire dans votre répertoire **~/MyPlaybooks**.

Grâce à cette pratique, vous pouvez retrouver tous vos playbooks en un seul endroit et vous pouvez exécuter vos playbooks sans invoquer les privilèges de l'administrateur.

NOTE

Vous n'avez besoin que des privilèges **root** sur les nœuds gérés pour exécuter les rôles **ipaserver**, **ipareplica**, **ipaclient** et **ipabackup ansible-freeipa** . Ces rôles nécessitent un accès privilégié aux répertoires et au gestionnaire de paquets logiciels **dnf**.

Cette section décrit comment créer le répertoire **~/MyPlaybooks** et le configurer de manière à ce que vous puissiez l'utiliser pour stocker et exécuter des playbooks Ansible.

Conditions préalables

- Vous avez installé un serveur IdM sur vos nœuds gérés, ***server.idm.example.com*** et ***replica.idm.example.com***.
- Vous avez configuré le DNS et le réseau pour pouvoir vous connecter aux nœuds gérés, ***server.idm.example.com*** et ***replica.idm.example.com***directement à partir du nœud de contrôle.
- Vous connaissez le mot de passe de l'IdM **admin**.

Procédure

1. Créez un répertoire pour votre configuration Ansible et vos playbooks dans votre répertoire personnel :

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

2. Allez dans le répertoire **~/MyPlaybooks/**:

```
$ cd ~/MyPlaybooks
```

3. Créez le fichier **~/MyPlaybooks/ansible.cfg** avec le contenu suivant :

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

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

4. Créez le fichier **~/MyPlaybooks/inventory** avec le contenu suivant :

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

[us]
replica.idm.example.com

[ipaserver:children]
eu
us
```

Cette configuration définit deux groupes d'hôtes, **eu** et **us**, pour les hôtes de ces sites. En outre, cette configuration définit le groupe d'hôtes **ipaserver**, qui contient tous les hôtes des groupes **eu** et **us**.

5. [Facultatif] Créez une clé publique et une clé privée SSH. Pour simplifier l'accès dans votre environnement de test, ne définissez pas de mot de passe pour la clé privée :

```
$ ssh-keygen
```

6. Copiez la clé publique SSH dans le compte IdM **admin** sur chaque nœud géré :

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

Ces commandes nécessitent la saisie du mot de passe IdM **admin**.

Ressources supplémentaires

- Voir Installation d'un serveur de gestion des identités à l'aide d'un playbook Ansible .
- Voir Comment constituer votre inventaire .

CHAPITRE 31. UTILISER LES PLAYBOOKS ANSIBLE POUR GÉRER LE CONTRÔLE D'ACCÈS BASÉ SUR LES RÔLES DANS IDM

Le contrôle d'accès basé sur les rôles (RBAC) est un mécanisme de contrôle d'accès neutre défini autour des rôles et des privilèges. Les composants du contrôle d'accès basé sur les rôles dans la gestion de l'identité (IdM) sont les rôles, les privilèges et les permissions :

- **Permissions** accorder le droit d'effectuer une tâche spécifique telle que l'ajout ou la suppression d'utilisateurs, la modification d'un groupe, l'activation de l'accès en lecture, etc.
- **Privileges** combiner les autorisations, par exemple toutes les autorisations nécessaires pour ajouter un nouvel utilisateur.
- **Roles** accorder un ensemble de privilèges à des utilisateurs, des groupes d'utilisateurs, des hôtes ou des groupes d'hôtes.

Dans les grandes entreprises en particulier, l'utilisation de RBAC peut aider à créer un système hiérarchique d'administrateurs avec leurs domaines de responsabilité individuels.

Ce chapitre décrit les opérations suivantes effectuées lors de la gestion de RBAC à l'aide des playbooks Ansible :

- Permissions dans l'IdM
- Permissions gérées par défaut
- Privilèges dans l'IdM
- Rôles dans l'IdM
- Rôles prédéfinis dans l'IdM
- Utiliser Ansible pour s'assurer qu'un rôle IdM RBAC avec des privilèges est présent
- Utiliser Ansible pour s'assurer qu'un rôle IdM RBAC est absent
- Utiliser Ansible pour s'assurer qu'un groupe d'utilisateurs est assigné à un rôle IdM RBAC
- Utiliser Ansible pour s'assurer que des utilisateurs spécifiques ne sont pas affectés à un rôle IdM RBAC
- Utiliser Ansible pour s'assurer qu'un service est membre d'un rôle IdM RBAC
- Utiliser Ansible pour s'assurer qu'un hôte est membre d'un rôle IdM RBAC
- Utiliser Ansible pour s'assurer qu'un groupe d'hôtes est membre d'un rôle IdM RBAC

31.1. PERMISSIONS DANS L'IDM

Les autorisations sont l'unité de niveau le plus bas du contrôle d'accès basé sur les rôles. Elles définissent les opérations ainsi que les entrées LDAP auxquelles ces opérations s'appliquent. Comparables à des blocs de construction, les autorisations peuvent être attribuées à autant de privilèges que nécessaire.
Une ou plusieurs adresses **rights** définissent les opérations autorisées :

- **write**
- **read**
- **search**
- **compare**
- **add**
- **delete**
- **all**

Ces opérations s'appliquent à trois sites de base **targets**:

- **subtree**: un nom de domaine (DN) ; la sous-arborescence sous ce DN
- **target filter**un filtre LDAP
- **target**: DN avec des caractères génériques possibles pour spécifier les entrées

En outre, les options de commodité suivantes définissent le(s) attribut(s) correspondant(s) :

- **type**: un type d'objet (utilisateur, groupe, etc.) ; définit **subtree** et **target filter**
- **memberof**: membres d'un groupe ; fixe un **target filter**
- **targetgroup**: accorde l'accès à la modification d'un groupe spécifique (par exemple en accordant les droits de gérer l'appartenance à un groupe) ; définit une valeur de **target**

Grâce aux autorisations IdM, vous pouvez contrôler quels utilisateurs ont accès à quels objets et même à quels attributs de ces objets. L'IdM vous permet d'autoriser ou de bloquer des attributs individuels ou de modifier la visibilité totale d'une fonction IdM spécifique, telle que les utilisateurs, les groupes ou sudo, pour tous les utilisateurs anonymes, tous les utilisateurs authentifiés ou seulement un certain groupe d'utilisateurs privilégiés.
Par exemple, la flexibilité de cette approche des autorisations est utile pour un administrateur qui souhaite limiter l'accès des utilisateurs ou des groupes uniquement aux sections spécifiques auxquelles ces utilisateurs ou groupes ont besoin d'accéder et rendre les autres sections complètement cachées pour eux.

NOTE

Une autorisation ne peut pas contenir d'autres autorisations.

31.2. PERMISSIONS GÉRÉES PAR DÉFAUT

Les autorisations gérées sont des autorisations fournies par défaut avec l'IdM. Elles se comportent comme les autres autorisations créées par l'utilisateur, avec les différences suivantes :

- Vous ne pouvez pas les supprimer ni modifier leur nom, leur emplacement et leurs attributs cibles.
- Ils ont trois séries d'attributs :
  - **Default** l'utilisateur ne peut pas les modifier, car ils sont gérés par IdM

- **Included** les attributs, qui sont des attributs supplémentaires ajoutés par l'utilisateur
- **Excluded** les attributs, qui sont des attributs supprimés par l'utilisateur

Une autorisation gérée s'applique à tous les attributs qui figurent dans les ensembles d'attributs par défaut et inclus, mais pas dans l'ensemble exclu.

NOTE

Bien que vous ne puissiez pas supprimer une autorisation gérée, le fait de définir son type de liaison sur autorisation et de supprimer l'autorisation gérée de tous les privilèges la désactive effectivement.

Les noms de toutes les autorisations gérées commencent par **System:**, par exemple **System: Add Sudo rule** ou **System: Modify Services**. Les versions antérieures de l'IdM utilisaient un schéma différent pour les autorisations par défaut. Par exemple, l'utilisateur ne pouvait pas les supprimer et ne pouvait les attribuer qu'à des privilèges. La plupart de ces autorisations par défaut ont été transformées en autorisations gérées, mais les autorisations suivantes utilisent toujours le schéma précédent :

- Ajouter une tâche de reconstruction de l'adhésion automatique
- Ajouter des sous-enregistrements de configuration
- Ajouter des accords de réplication
- Certificat Supprimer la retenue
- Obtenir l'état des certificats de l'autorité de certification
- Plage de lecture de l'ADN
- Modifier la portée de l'ADN
- Lire la configuration des gestionnaires PassSync
- Modifier la configuration des gestionnaires PassSync
- Lire les accords de réplication
- Modifier les accords de réplication
- Supprimer les accords de réplication
- Lire la configuration de la base de données LDBM
- Demande de certificat
- Demande de certificat ignorant les listes de contrôle de l'autorité de certification
- Demander des certificats à un autre hôte
- Récupérer des certificats auprès de l'autorité de certification
- Révoquer le certificat
- Écriture de la configuration de l'IPA

NOTE

Si vous tentez de modifier une autorisation gérée à partir de la ligne de commande, le système ne vous permet pas de changer les attributs que vous ne pouvez pas modifier, la commande échoue. Si vous tentez de modifier une autorisation gérée à partir de l'interface Web, les attributs que vous ne pouvez pas modifier sont désactivés.

31.3. PRIVILÈGES DANS L'IDM

Un privilège est un groupe de permissions applicables à un rôle.
Alors qu'une autorisation donne le droit d'effectuer une seule opération, certaines tâches de l'IdM nécessitent plusieurs autorisations pour être menées à bien. Par conséquent, un privilège combine les différentes autorisations requises pour effectuer une tâche spécifique.
Par exemple, la création d'un compte pour un nouvel utilisateur IdM nécessite les autorisations suivantes :

- Création d'une nouvelle entrée utilisateur
- Réinitialisation du mot de passe d'un utilisateur
- Ajout du nouvel utilisateur au groupe d'utilisateurs IPA par défaut

La combinaison de ces trois tâches de bas niveau en une tâche de plus haut niveau sous la forme d'un privilège personnalisé nommé, par exemple, **Add User** facilite la gestion des rôles par l'administrateur du système. L'IdM contient déjà plusieurs privilèges par défaut. Outre les utilisateurs et les groupes d'utilisateurs, des privilèges sont également attribués aux hôtes et aux groupes d'hôtes, ainsi qu'aux services de réseau. Cette pratique permet un contrôle fin des opérations effectuées par un ensemble d'utilisateurs sur un ensemble d'hôtes utilisant des services de réseau spécifiques.

NOTE

Un privilège ne peut pas contenir d'autres privilèges.

31.4. RÔLES DANS L'IDM

Un rôle est une liste de privilèges que les utilisateurs spécifiés pour ce rôle possèdent.
En effet, les autorisations permettent d'effectuer certaines tâches de bas niveau (créer une entrée utilisateur, ajouter une entrée à un groupe, etc.), tandis que les privilèges combinent une ou plusieurs de ces autorisations nécessaires à une tâche de plus haut niveau (comme la création d'un nouvel utilisateur dans un groupe donné). Les rôles regroupent les privilèges selon les besoins : par exemple, un administrateur d'utilisateurs peut ajouter, modifier et supprimer des utilisateurs.

IMPORTANT

Les rôles sont utilisés pour classer les actions autorisées. Ils ne sont pas utilisés comme outil pour mettre en œuvre la séparation des privilèges ou pour se protéger contre l'escalade des privilèges.

NOTE

Les rôles ne peuvent pas contenir d'autres rôles.

31.5. RÔLES PRÉDÉFINIS DANS LA GESTION DE L'IDENTITÉ

Red Hat Identity Management fournit la gamme suivante de rôles prédéfinis :

Tableau 31.1. Rôles prédéfinis dans la gestion des identités

Rôle	Privilège	Description
Administrateur des inscriptions	Inscription au programme d'accueil	Responsable de l'inscription du client ou de l'hôte
helpdesk	Modifier les utilisateurs et réinitialiser les mots de passe, Modifier l'appartenance à un groupe	Effectuer des tâches simples d'administration des utilisateurs
Spécialiste de la sécurité informatique	Administrateurs Netgroups, Administrateur HBAC, Administrateur Sudo	Responsable de la gestion de la politique de sécurité telle que les contrôles d'accès basés sur l'hôte, les règles sudo
Spécialiste des technologies de l'information	Administrateurs d'hôtes, administrateurs de groupes d'hôtes, administrateurs de services, administrateurs de montages automatiques	Responsable de la gestion des hôtes
Architecte de sécurité	Administrateur de la délégation, Administrateurs de la réplication, Configuration de l'IPA en écriture, Administrateur de la politique des mots de passe	Responsable de la gestion de l'environnement de gestion des identités, de la création de trusts et d'accords de réplication
Administrateur des utilisateurs	Administrateurs d'utilisateurs, administrateurs de groupes, administrateurs d'utilisateurs de scène	Responsable de la création des utilisateurs et des groupes

31.6. UTILISER ANSIBLE POUR S'ASSURER QU'UN RÔLE IDM RBAC AVEC DES PRIVILÈGES EST PRÉSENT

Pour exercer un contrôle plus granulaire sur l'accès basé sur les rôles (RBAC) aux ressources dans la gestion des identités (IdM) que les rôles par défaut, créez un rôle personnalisé.

La procédure suivante décrit comment utiliser un playbook Ansible pour définir les privilèges d'un nouveau rôle personnalisé IdM et assurer sa présence. Dans l'exemple, le nouveau rôle **user_and_host_administrator** contient une combinaison unique des privilèges suivants qui sont présents dans IdM par défaut :

- **Group Administrators**
- **User Administrators**
- **Stage User Administrators**

- **Group Administrators**

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

Procédure

1. Naviguez jusqu'au répertoire ***~/<MyPlaybooks>/*** répertoire :

```
$ cd ~/<MyPlaybooks>/
```

2. Faites une copie du fichier **role-member-user-present.yml** situé dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/role/**:

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/role/role-member-user-present.yml role-member-user-present-copy.yml
```

3. Ouvrez le fichier **role-member-user-present-copy.yml** Ansible playbook pour l'éditer.
4. Adaptez le fichier en définissant les variables suivantes dans la section **iparole** task :
   - Définissez la variable **ipaadmin_password** avec le mot de passe de l'administrateur IdM.
   - Attribuez à la variable **name** le nom du nouveau rôle.
   - Définissez la liste **privilege** avec les noms des privilèges IdM que vous souhaitez inclure dans le nouveau rôle.
   - Si vous le souhaitez, définissez la variable **user** avec le nom de l'utilisateur auquel vous souhaitez attribuer le nouveau rôle.
   - Si vous le souhaitez, attribuez à la variable **group** le nom du groupe auquel vous souhaitez attribuer le nouveau rôle.

   Il s'agit du fichier playbook Ansible modifié pour l'exemple actuel :

```
---
- name: Playbook to manage IPA role with members.
  hosts: ipaserver
  become: yes
  gather_facts: no
```

```
vars_files:
- /home/user_name/MyPlaybooks/secret.yml
tasks:
- iparole:
  ipaadmin_password: "{{ ipaadmin_password }}"
  name: user_and_host_administrator
  user: idm_user01
  group: idm_group01
  privilege:
  - Group Administrators
  - User Administrators
  - Stage User Administrators
  - Group Administrators
```

5. Enregistrer le fichier.

6. Exécutez le playbook Ansible. Spécifiez le fichier du livre de jeu, le fichier contenant le mot de passe protégeant le fichier **secret.yml** et le fichier d'inventaire :

```
$ ansible-playbook --vault-password-file=password_file -v -i
~/<MyPlaybooks>/inventory role-member-user-present-copy.yml
```

Ressources supplémentaires

- Voir Chiffrer du contenu avec Ansible Vault.
- Voir Rôles dans IdM.
- Voir le fichier **README-role** dans le répertoire **/usr/share/doc/ansible-freeipa/**.
- Voir les exemples de playbooks dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/iparole**.

31.7. UTILISER ANSIBLE POUR S'ASSURER QU'UN RÔLE IDM RBAC EST ABSENT

En tant qu'administrateur système gérant le contrôle d'accès basé sur les rôles (RBAC) dans la gestion des identités (IdM), vous pouvez vouloir garantir l'absence d'un rôle obsolète afin qu'aucun administrateur ne l'attribue accidentellement à un utilisateur.

La procédure suivante décrit comment utiliser un playbook Ansible pour s'assurer de l'absence d'un rôle. L'exemple ci-dessous décrit comment s'assurer que le rôle personnalisé **user_and_host_administrator** n'existe pas dans IdM.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.

- L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
- L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

Procédure

1. Naviguez jusqu'au répertoire ***~/<MyPlaybooks>/*** répertoire :

```
$ cd ~/<MyPlaybooks>/
```

2. Faites une copie du fichier **role-is-absent.yml** situé dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/role/**:

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/role/role-is-absent.yml role-is-absent-copy.yml
```

3. Ouvrez le fichier **role-is-absent-copy.yml** Ansible playbook pour l'éditer.
4. Adaptez le fichier en définissant les variables suivantes dans la section **iparole** task :
   - Définissez la variable **ipaadmin_password** avec le mot de passe de l'administrateur IdM.
   - Définissez la variable **name** avec le nom du rôle.
   - Assurez-vous que la variable **state** est définie sur **absent**.

   Il s'agit du fichier playbook Ansible modifié pour l'exemple actuel :

```
---
- name: Playbook to manage IPA role with members.
  hosts: ipaserver
  become: yes
  gather_facts: no

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - iparole:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: user_and_host_administrator
      state: absent
```

5. Enregistrer le fichier.
6. Exécutez le playbook Ansible. Spécifiez le fichier du livre de jeu, le fichier contenant le mot de passe protégeant le fichier **secret.yml** et le fichier d'inventaire :

```
$ ansible-playbook --vault-password-file=password_file -v -i ~/<MyPlaybooks>/inventory role-is-absent-copy.yml
```

Ressources supplémentaires

- Voir Chiffrer du contenu avec Ansible Vault.
- Voir Rôles dans IdM.
- Voir le fichier Markdown de **README-role** dans le répertoire **/usr/share/doc/ansible-freeipa/**.
- Voir les exemples de playbooks dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/iparole**.

31.8. UTILISER ANSIBLE POUR S'ASSURER QU'UN GROUPE D'UTILISATEURS EST ASSIGNÉ À UN RÔLE IDM RBAC

En tant qu'administrateur système gérant le contrôle d'accès basé sur les rôles (RBAC) dans la gestion des identités (IdM), vous pouvez vouloir attribuer un rôle à un groupe spécifique d'utilisateurs, par exemple les administrateurs juniors.

L'exemple suivant décrit comment utiliser un playbook Ansible pour s'assurer que le rôle IdM RBAC **helpdesk** est attribué à **junior_sysadmins**.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

Procédure

1. Naviguez jusqu'au répertoire ***~/<MyPlaybooks>/*** répertoire :

```
$ cd ~/<MyPlaybooks>/
```

2. Faites une copie du fichier **role-member-group-present.yml** situé dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/role/**:

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/role/role-member-group-present.yml role-member-group-present-copy.yml
```

3. Ouvrez le fichier **role-member-group-present-copy.yml** Ansible playbook pour l'éditer.
4. Adaptez le fichier en définissant les variables suivantes dans la section **iparole** task :
   - Définissez la variable **ipaadmin_password** avec le mot de passe de l'administrateur IdM.
   - Définissez la variable **name** avec le nom du rôle que vous souhaitez attribuer.

- Attribuez à la variable **group** le nom du groupe.
- Fixer la variable **action** à **member**.

Il s'agit du fichier playbook Ansible modifié pour l'exemple actuel :

```
---
- name: Playbook to manage IPA role with members.
  hosts: ipaserver
  become: yes
  gather_facts: no

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - iparole:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: helpdesk
      group: junior_sysadmins
      action: member
```

5. Enregistrer le fichier.
6. Exécutez le playbook Ansible. Spécifiez le fichier du livre de jeu, le fichier contenant le mot de passe protégeant le fichier **secret.yml** et le fichier d'inventaire :

```
$ ansible-playbook --vault-password-file=password_file -v -i
~/<MyPlaybooks>/inventory role-member-group-present-copy.yml
```

Ressources supplémentaires

- Voir Chiffrer du contenu avec Ansible Vault.
- Voir Rôles dans IdM.
- Voir le fichier Markdown de **README-role** dans le répertoire **/usr/share/doc/ansible-freeipa/**.
- Voir les exemples de playbooks dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/iparole**.

31.9. UTILISER ANSIBLE POUR S'ASSURER QUE DES UTILISATEURS SPÉCIFIQUES NE SONT PAS AFFECTÉS À UN RÔLE IDM RBAC

En tant qu'administrateur système gérant le contrôle d'accès basé sur les rôles (RBAC) dans la gestion des identités (IdM), vous voudrez peut-être vous assurer qu'un rôle RBAC n'est pas attribué à des utilisateurs spécifiques après qu'ils ont, par exemple, changé de poste au sein de l'entreprise.

La procédure suivante décrit comment utiliser un playbook Ansible pour s'assurer que les utilisateurs nommés **user_01** et **user_02** ne sont pas affectés au rôle **helpdesk**.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.

- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

Procédure

1. Naviguez jusqu'au répertoire ***~/<MyPlaybooks>/*** répertoire :

```
$ cd ~/<MyPlaybooks>/
```

2. Faites une copie du fichier **role-member-user-absent.yml** situé dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/role/**:

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/role/role-member-user-absent.yml role-member-user-absent-copy.yml
```

3. Ouvrez le fichier **role-member-user-absent-copy.yml** Ansible playbook pour l'éditer.
4. Adaptez le fichier en définissant les variables suivantes dans la section **iparole** task :
   - Définissez la variable **ipaadmin_password** avec le mot de passe de l'administrateur IdM.
   - Définissez la variable **name** avec le nom du rôle que vous souhaitez attribuer.
   - Définissez la liste **user** avec les noms des utilisateurs.
   - Fixer la variable **action** à **member**.
   - Fixer la variable **state** à **absent**.

   Il s'agit du fichier playbook Ansible modifié pour l'exemple actuel :

```
---
- name: Playbook to manage IPA role with members.
  hosts: ipaserver
  become: yes
  gather_facts: no

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - iparole:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: helpdesk
      user
      - user_01
```

```
    - user_02
    action: member
    state: absent
```

5. Enregistrer le fichier.
6. Exécutez le playbook Ansible. Spécifiez le fichier du livre de jeu, le fichier contenant le mot de passe protégeant le fichier **secret.yml** et le fichier d'inventaire :

```
$ ansible-playbook --vault-password-file=password_file -v -i
~/<MyPlaybooks>/inventory role-member-user-absent-copy.yml
```

Ressources supplémentaires

- Voir Chiffrer du contenu avec Ansible Vault.
- Voir Rôles dans IdM.
- Voir le fichier Markdown de **README-role** dans le répertoire **/usr/share/doc/ansible-freeipa/**.
- Voir les exemples de playbooks dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/iparole**.

31.10. UTILISER ANSIBLE POUR S'ASSURER QU'UN SERVICE EST MEMBRE D'UN RÔLE IDM RBAC

En tant qu'administrateur système gérant le contrôle d'accès basé sur les rôles (RBAC) dans la gestion des identités (IdM), vous pouvez vouloir vous assurer qu'un service spécifique inscrit dans l'IdM est membre d'un rôle particulier. L'exemple suivant décrit comment s'assurer que le rôle personnalisé **web_administrator** peut gérer le service **HTTP** qui s'exécute sur le serveur **client01.idm.example.com**.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.
- Le rôle **web_administrator** existe dans IdM.
- Le service **HTTP/client01.idm.example.com@IDM.EXAMPLE.COM** existe dans IdM.

Procédure

1. Naviguez jusqu'au répertoire ***~/<MyPlaybooks>/*** répertoire :

```
$ cd ~/<MyPlaybooks>/
```

2. Faites une copie du fichier **role-member-service-present.yml** situé dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/role/**:

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/role/role-member-service-present-absent.yml role-member-service-present-copy.yml
```

3. Ouvrez le fichier **role-member-service-present-copy.yml** Ansible playbook pour l'éditer.
4. Adaptez le fichier en définissant les variables suivantes dans la section **iparole** task :
   - Définissez la variable **ipaadmin_password** avec le mot de passe de l'administrateur IdM.
   - Définissez la variable **name** avec le nom du rôle que vous souhaitez attribuer.
   - Définissez la liste **service** avec le nom du service.
   - Fixer la variable **action** à **member**.

   Il s'agit du fichier playbook Ansible modifié pour l'exemple actuel :

```
---
- name: Playbook to manage IPA role with members.
  hosts: ipaserver
  become: yes
  gather_facts: no

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - iparole:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: web_administrator
      service:
      - HTTP/client01.idm.example.com
      action: member
```

5. Enregistrer le fichier.
6. Exécutez le playbook Ansible. Spécifiez le fichier du livre de jeu, le fichier contenant le mot de passe protégeant le fichier **secret.yml** et le fichier d'inventaire :

```
$ ansible-playbook --vault-password-file=password_file -v -i ~/<MyPlaybooks>/inventory role-member-service-present-copy.yml
```

Ressources supplémentaires

- Voir Chiffrer du contenu avec Ansible Vault.
- Voir Rôles dans IdM.
- Voir le fichier Markdown de **README-role** dans le répertoire **/usr/share/doc/ansible-freeipa/**.

- Voir les exemples de playbooks dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/iparole**.

31.11. UTILISER ANSIBLE POUR S'ASSURER QU'UN HÔTE EST MEMBRE D'UN RÔLE IDM RBAC

En tant qu'administrateur système gérant le contrôle d'accès basé sur les rôles dans la gestion des identités (IdM), vous pouvez vouloir vous assurer qu'un hôte ou un groupe d'hôtes spécifique est associé à un rôle spécifique. L'exemple suivant décrit comment s'assurer que le rôle personnalisé **web_administrator** peut gérer l'hôte IdM **client01.idm.example.com** sur lequel le service **HTTP** est exécuté.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.
- Le rôle **web_administrator** existe dans IdM.
- L'hôte **client01.idm.example.com** existe dans IdM.

Procédure

1. Naviguez jusqu'au répertoire ***~/<MyPlaybooks>/*** répertoire :

```
$ cd ~/<MyPlaybooks>/
```

2. Faites une copie du fichier **role-member-host-present.yml** situé dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/role/**:

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/role/role-member-host-present.yml role-member-host-present-copy.yml
```

3. Ouvrez le fichier **role-member-host-present-copy.yml** Ansible playbook pour l'éditer.
4. Adaptez le fichier en définissant les variables suivantes dans la section **iparole** task :
   - Définissez la variable **ipaadmin_password** avec le mot de passe de l'administrateur IdM.
   - Définissez la variable **name** avec le nom du rôle que vous souhaitez attribuer.
   - Définissez la liste **host** avec le nom de l'hôte.

Il s'agit du fichier playbook Ansible modifié pour l'exemple actuel :

```
---
- name: Playbook to manage IPA role with members.
  hosts: ipaserver
  become: yes
  gather_facts: no

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - iparole:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: web_administrator
      host:
      - client01.idm.example.com
      action: member
```

5. Enregistrer le fichier.

6. Exécutez le playbook Ansible. Spécifiez le fichier du livre de jeu, le fichier contenant le mot de passe protégeant le fichier **secret.yml** et le fichier d'inventaire :

```
$ ansible-playbook --vault-password-file=password_file -v -i
~/<MyPlaybooks>/inventory role-member-host-present-copy.yml
```

Ressources supplémentaires

- Voir Chiffrer du contenu avec Ansible Vault.
- Voir Rôles dans IdM.
- Voir le fichier Markdown de **README-role** dans le répertoire **/usr/share/doc/ansible-freeipa/**.
- Voir les exemples de playbooks dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/iparole**.

31.12. UTILISER ANSIBLE POUR S'ASSURER QU'UN GROUPE D'HÔTES EST MEMBRE D'UN RÔLE IDM RBAC

En tant qu'administrateur système gérant le contrôle d'accès basé sur les rôles dans la gestion des identités (IdM), vous pouvez vouloir vous assurer qu'un hôte ou un groupe d'hôtes spécifique est associé à un rôle spécifique. L'exemple suivant décrit comment s'assurer que le rôle personnalisé **web_administrator** peut gérer le groupe d'hôtes IdM **web_servers** sur lequel le service **HTTP** est exécuté.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.

- Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
- L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
- L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

- Le rôle **web_administrator** existe dans IdM.
- Le groupe d'hôtes **web_servers** existe dans IdM.

Procédure

1. Naviguez jusqu'au répertoire ***~/<MyPlaybooks>/*** répertoire :

```
$ cd ~/<MyPlaybooks>/
```

2. Faites une copie du fichier **role-member-hostgroup-present.yml** situé dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/role/**:

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/role/role-member-hostgroup-present.yml role-member-hostgroup-present-copy.yml
```

3. Ouvrez le fichier **role-member-hostgroup-present-copy.yml** Ansible playbook pour l'éditer.
4. Adaptez le fichier en définissant les variables suivantes dans la section **iparole** task :
   - Définissez la variable **ipaadmin_password** avec le mot de passe de l'administrateur IdM.
   - Définissez la variable **name** avec le nom du rôle que vous souhaitez attribuer.
   - Définissez la liste **hostgroup** avec le nom du groupe d'hôtes.

   Il s'agit du fichier playbook Ansible modifié pour l'exemple actuel :

```
---
- name: Playbook to manage IPA role with members.
  hosts: ipaserver
  become: yes
  gather_facts: no

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - iparole:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: web_administrator
      hostgroup:
      - web_servers
      action: member
```

5. Enregistrer le fichier.

6. Exécutez le playbook Ansible. Spécifiez le fichier du livre de jeu, le fichier contenant le mot de passe protégeant le fichier **secret.yml** et le fichier d'inventaire :

```
$ ansible-playbook --vault-password-file=password_file -v -i
~/<MyPlaybooks>/inventory role-member-hostgroup-present-copy.yml
```

Ressources supplémentaires

- Voir Chiffrer du contenu avec Ansible Vault.
- Voir Rôles dans IdM.
- Voir le fichier Markdown de **README-role** dans le répertoire **/usr/share/doc/ansible-freeipa/**.
- Voir les exemples de playbooks dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/iparole**.

CHAPITRE 32. UTILISER LES PLAYBOOKS ANSIBLE POUR GÉRER LES PRIVILÈGES RBAC

Le contrôle d'accès basé sur les rôles (RBAC) est un mécanisme de contrôle d'accès neutre défini autour de rôles, de privilèges et de permissions. Dans les grandes entreprises en particulier, l'utilisation du RBAC peut aider à créer un système hiérarchique d'administrateurs avec leurs domaines de responsabilité individuels.

Ce chapitre décrit les opérations suivantes pour utiliser les playbooks Ansible afin de gérer les privilèges RBAC dans la gestion des identités (IdM) :

- Utiliser Ansible pour s'assurer qu'un privilège RBAC personnalisé est présent
- Utiliser Ansible pour s'assurer que les permissions des membres sont présentes dans un privilège IdM RBAC personnalisé
- Utiliser Ansible pour s'assurer qu'un privilège IdM RBAC n'inclut pas une permission
- Utiliser Ansible pour renommer un privilège IdM RBAC personnalisé
- Utiliser Ansible pour s'assurer qu'un privilège IdM RBAC est absent

Conditions préalables

- Vous comprenez les concepts et les principes de RBAC .

32.1. UTILISER ANSIBLE POUR S'ASSURER QU'UN PRIVILÈGE IDM RBAC PERSONNALISÉ EST PRÉSENT

Pour disposer d'un privilège personnalisé pleinement fonctionnel dans le contrôle d'accès basé sur les rôles (RBAC) de la gestion des identités (IdM), vous devez procéder par étapes :

1. Créer un privilège sans aucune autorisation.
2. Ajoutez les autorisations de votre choix au privilège.

La procédure suivante décrit comment créer un privilège vide à l'aide d'une séquence Ansible afin de pouvoir y ajouter ultérieurement des autorisations. L'exemple décrit comment créer un privilège nommé **full_host_administration** destiné à combiner toutes les autorisations IdM liées à l'administration des hôtes.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.

- L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

Procédure

1. Naviguez jusqu'au répertoire ***~/MyPlaybooks/*** répertoire :

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

2. Faites une copie du fichier **privilege-present.yml** situé dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/privilege/**:

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/privilege/privilege-present.yml privilege-present-copy.yml
```

3. Ouvrez le fichier **privilege-present-copy.yml** Ansible playbook pour l'éditer.

4. Adaptez le fichier en définissant les variables suivantes dans la section **ipaprivilege** task :

   - Définissez la variable **ipaadmin_password** avec le mot de passe de l'administrateur IdM.
   - Attribuez à la variable **name** le nom du nouveau privilège, **full_host_administration**.
   - En option, décrivez le privilège à l'aide de la variable **description**.

   Il s'agit du fichier playbook Ansible modifié pour l'exemple actuel :

```
---
- name: Privilege present example
  hosts: ipaserver

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Ensure privilege full_host_administration is present
    ipaprivilege:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: full_host_administration
      description: This privilege combines all IdM permissions related to host administration
```

5. Enregistrer le fichier.

6. Exécutez le playbook Ansible. Spécifiez le fichier du livre de jeu, le fichier contenant le mot de passe protégeant le fichier **secret.yml** et le fichier d'inventaire :

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

32.2. UTILISER ANSIBLE POUR S'ASSURER QUE LES PERMISSIONS DES MEMBRES SONT PRÉSENTES DANS UN PRIVILÈGE IDM RBAC PERSONNALISÉ

Pour disposer d'un privilège personnalisé pleinement fonctionnel dans le contrôle d'accès basé sur les rôles (RBAC) de la gestion des identités (IdM), vous devez procéder par étapes :

1. Créer un privilège sans aucune autorisation.
2. Ajoutez les autorisations de votre choix au privilège.

La procédure suivante explique comment utiliser un cahier de jeu Ansible pour ajouter des autorisations à un privilège créé à l'étape précédente. L'exemple décrit comment ajouter toutes les autorisations IdM liées à l'administration des hôtes à un privilège nommé **full_host_administration**. Par défaut, les autorisations sont réparties entre les privilèges **Host Enrollment**, **Host Administrators** et **Host Group Administrator**.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.
- Le privilège ***full_host_administration*** existe. Pour plus d'informations sur la création d'un privilège à l'aide d'Ansible, voir Utilisation d'Ansible pour garantir la présence d'un privilège IdM RBAC personnalisé.

Procédure

1. Naviguez jusqu'au répertoire ***~/MyPlaybooks/*** répertoire :

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

2. Faites une copie du fichier **privilege-member-present.yml** situé dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/privilege/**:

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/privilege/privilege-member-present.yml
privilege-member-present-copy.yml
```

3. Ouvrez le fichier **privilege-member-present-copy.yml** Ansible playbook pour l'éditer.
4. Adaptez le fichier en définissant les variables suivantes dans la section **ipaprivilege** task :
   - Adaptez le site **name** de la tâche pour qu'il corresponde à votre cas d'utilisation.
   - Définissez la variable **ipaadmin_password** avec le mot de passe de l'administrateur IdM.
   - Attribuez à la variable **name** le nom du privilège.

- Définissez la liste **permission** avec les noms des autorisations que vous souhaitez inclure dans le privilège.
- Assurez-vous que la variable **action** est fixée à **member**.

Il s'agit du fichier playbook Ansible modifié pour l'exemple actuel :

```
---
- name: Privilege member present example
  hosts: ipaserver

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Ensure that permissions are present for the "full_host_administration" privilege
    ipaprivilege:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: full_host_administration
      permission:
      - "System: Add krbPrincipalName to a Host"
      - "System: Enroll a Host"
      - "System: Manage Host Certificates"
      - "System: Manage Host Enrollment Password"
      - "System: Manage Host Keytab"
      - "System: Manage Host Principals"
      - "Retrieve Certificates from the CA"
      - "Revoke Certificate"
      - "System: Add Hosts"
      - "System: Add krbPrincipalName to a Host"
      - "System: Enroll a Host"
      - "System: Manage Host Certificates"
      - "System: Manage Host Enrollment Password"
      - "System: Manage Host Keytab"
      - "System: Manage Host Keytab Permissions"
      - "System: Manage Host Principals"
      - "System: Manage Host SSH Public Keys"
      - "System: Manage Service Keytab"
      - "System: Manage Service Keytab Permissions"
      - "System: Modify Hosts"
      - "System: Remove Hosts"
      - "System: Add Hostgroups"
      - "System: Modify Hostgroup Membership"
      - "System: Modify Hostgroups"
      - "System: Remove Hostgroups"
```

5. Enregistrer le fichier.
6. Exécutez le playbook Ansible. Spécifiez le fichier du livre de jeu, le fichier contenant le mot de passe protégeant le fichier **secret.yml** et le fichier d'inventaire :

```
$ ansible-playbook --vault-password-file=password_file -v -i inventory privilege-member-present-copy.yml
```

32.3. UTILISER ANSIBLE POUR S'ASSURER QU'UN PRIVILÈGE IDM RBAC N'INCLUT PAS UNE PERMISSION

En tant qu'administrateur système de la gestion des identités (IdM), vous pouvez personnaliser le contrôle d'accès basé sur les rôles de l'IdM.

La procédure suivante décrit comment utiliser un livre de jeu Ansible pour supprimer une autorisation d'un privilège. L'exemple décrit comment supprimer l'autorisation **Request Certificates ignoring CA ACLs** du privilège par défaut **Certificate Administrators** parce que, par exemple, l'administrateur considère qu'il s'agit d'un risque de sécurité.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

Procédure

1. Naviguez jusqu'au répertoire ***~/MyPlaybooks/*** répertoire :

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

2. Faites une copie du fichier **privilege-member-present.yml** situé dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/privilege/**:

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/privilege/privilege-member-absent.yml
privilege-member-absent-copy.yml
```

3. Ouvrez le fichier **privilege-member-absent-copy.yml** Ansible playbook pour l'éditer.
4. Adaptez le fichier en définissant les variables suivantes dans la section **ipaprivilege** task :
   - Adaptez le site **name** de la tâche pour qu'il corresponde à votre cas d'utilisation.
   - Définissez la variable **ipaadmin_password** avec le mot de passe de l'administrateur IdM.
   - Attribuez à la variable **name** le nom du privilège.
   - Définissez la liste **permission** sur les noms des autorisations que vous souhaitez supprimer du privilège.
   - Assurez-vous que la variable **action** est fixée à **member**.
   - Assurez-vous que la variable **state** est fixée à **absent**.

Il s'agit du fichier playbook Ansible modifié pour l'exemple actuel :

```
---
- name: Privilege absent example
  hosts: ipaserver

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Ensure that the "Request Certificate ignoring CA ACLs" permission is absent from
the "Certificate Administrators" privilege
    ipaprivilege:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: Certificate Administrators
      permission:
      - "Request Certificate ignoring CA ACLs"
      action: member
      state: absent
```

5. Enregistrer le fichier.

6. Exécutez le playbook Ansible. Spécifiez le fichier du livre de jeu, le fichier contenant le mot de passe protégeant le fichier **secret.yml** et le fichier d'inventaire :

```
$ ansible-playbook --vault-password-file=password_file -v -i inventory privilege-
member-absent-copy.yml
```

32.4. UTILISER ANSIBLE POUR RENOMMER UN PRIVILÈGE IDM RBAC PERSONNALISÉ

En tant qu'administrateur système de la gestion des identités (IdM), vous pouvez personnaliser le contrôle d'accès basé sur les rôles de l'IdM.

La procédure suivante décrit comment renommer un privilège parce que, par exemple, vous lui avez retiré quelques autorisations. Par conséquent, le nom du privilège n'est plus exact. Dans l'exemple, l'administrateur renomme un privilège **full_host_administration** en **limited_host_administration**.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

- Le privilège ***full_host_administration*** existe. Pour plus d'informations sur l'ajout d'un privilège, voir Utilisation d'Ansible pour s'assurer qu'un privilège IdM RBAC personnalisé est présent .

Procédure

1. Naviguez jusqu'au répertoire ***~/MyPlaybooks/*** répertoire :

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

2. Faites une copie du fichier **privilege-present.yml** situé dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/privilege/**:

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/privilege/privilege-present.yml rename-privilege.yml
```

3. Ouvrez le fichier **rename-privilege.yml** Ansible playbook pour l'éditer.

4. Adaptez le fichier en définissant les variables suivantes dans la section **ipaprivilege** task :

   - Définissez la variable **ipaadmin_password** avec le mot de passe de l'administrateur IdM.
   - Définir la variable **name** avec le nom actuel du privilège.
   - Ajoutez la variable **rename** et attribuez-lui le nouveau nom du privilège.
   - Ajoutez la variable **state** et fixez-la à **renamed**.

5. Renommer le playbook lui-même, par exemple :

```
---
- name: Rename a privilege
  hosts: ipaserver
```

6. Renommez la tâche dans le playbook, par exemple :

```
[...]
tasks:
- name: Ensure the full_host_administration privilege is renamed to limited_host_administration
  ipaprivilege:
  [...]
```

Il s'agit du fichier playbook Ansible modifié pour l'exemple actuel :

```
---
- name: Rename a privilege
  hosts: ipaserver

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Ensure the full_host_administration privilege is renamed to limited_host_administration
    ipaprivilege:
      ipaadmin_password: "{{ ipaadmin_password }}"
```

```
    name: full_host_administration
    rename: limited_host_administration
    state: renamed
```

7. Enregistrer le fichier.

8. Exécutez le playbook Ansible. Spécifiez le fichier du livre de jeu, le fichier contenant le mot de passe protégeant le fichier **secret.yml** et le fichier d'inventaire :

```
$ ansible-playbook --vault-password-file=password_file -v -i inventory rename-privilege.yml
```

32.5. UTILISER ANSIBLE POUR S'ASSURER QU'UN PRIVILÈGE IDM RBAC EST ABSENT

En tant qu'administrateur système de la gestion des identités (IdM), vous pouvez personnaliser le contrôle d'accès basé sur les rôles de l'IdM. La procédure suivante décrit comment utiliser un playbook Ansible pour s'assurer qu'un privilège RBAC est absent. L'exemple décrit comment s'assurer que le privilège **CA administrator** est absent. Grâce à cette procédure, l'administrateur de **admin** devient le seul utilisateur capable de gérer les autorités de certification dans IdM.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

Procédure

1. Naviguez jusqu'au répertoire ***~/MyPlaybooks/*** répertoire :

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

2. Faites une copie du fichier **privilege-absent.yml** situé dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/privilege/**:

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/privilege/privilege-absent.yml privilege-absent-copy.yml
```

3. Ouvrez le fichier **privilege-absent-copy.yml** Ansible playbook pour l'éditer.

4. Adaptez le fichier en définissant les variables suivantes dans la section **ipaprivilege** task :

- Définissez la variable **ipaadmin_password** avec le mot de passe de l'administrateur IdM.
- Attribuez à la variable **name** le nom du privilège que vous souhaitez supprimer.
- Assurez-vous que la variable **state** est fixée à **absent**.

5. Renommez la tâche dans le playbook, par exemple :

```
[...]
tasks:
- name: Ensure privilege "CA administrator" is absent
  ipaprivilege:
  [...]
```

Il s'agit du fichier playbook Ansible modifié pour l'exemple actuel :

```
---
- name: Privilege absent example
  hosts: ipaserver

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Ensure privilege "CA administrator" is absent
    ipaprivilege:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: CA administrator
      state: absent
```

6. Enregistrer le fichier.

7. Exécutez le playbook Ansible. Spécifiez le fichier du livre de jeu, le fichier contenant le mot de passe protégeant le fichier **secret.yml** et le fichier d'inventaire :

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

32.6. RESSOURCES SUPPLÉMENTAIRES

- Voir les privilèges dans l'IdM.
- Voir les autorisations dans IdM.
- Voir le fichier **README-privilege** disponible dans le répertoire **/usr/share/doc/ansible-freeipa/**.
- Voir les exemples de playbooks dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/ipaprivilege**.

CHAPITRE 33. UTILISER LES PLAYBOOKS ANSIBLE POUR GÉRER LES PERMISSIONS RBAC DANS IDM

Le contrôle d'accès basé sur les rôles (RBAC) est un mécanisme de contrôle d'accès neutre défini autour de rôles, de privilèges et de permissions. Dans les grandes entreprises en particulier, l'utilisation du RBAC peut aider à créer un système hiérarchique d'administrateurs avec leurs domaines de responsabilité individuels.

Ce chapitre décrit les opérations suivantes effectuées lors de la gestion des autorisations RBAC dans la gestion des identités (IdM) à l'aide des playbooks Ansible :

- Utiliser Ansible pour s'assurer qu'une permission RBAC est présente
- Utiliser Ansible pour s'assurer qu'une permission RBAC avec un attribut est présente
- Utiliser Ansible pour s'assurer qu'une permission RBAC est absente
- Utiliser Ansible pour s'assurer qu'un attribut est membre d'une permission IdM RBAC
- Utiliser Ansible pour s'assurer qu'un attribut n'est pas membre d'une permission RBAC IdM
- Utiliser Ansible pour renommer une permission IdM RBAC

Conditions préalables

- Vous comprenez les concepts et les principes de RBAC .

33.1. UTILISER ANSIBLE POUR S'ASSURER QU'UNE PERMISSION RBAC EST PRÉSENTE

En tant qu'administrateur système de la gestion des identités (IdM), vous pouvez personnaliser le contrôle d'accès basé sur les rôles (RBAC) de l'IdM.

La procédure suivante décrit comment utiliser un playbook Ansible pour s'assurer qu'une permission est présente dans IdM afin qu'elle puisse être ajoutée à un privilège. L'exemple décrit comment garantir l'état cible suivant :

- L'autorisation **MyPermission** existe.
- L'autorisation **MyPermission** ne peut être appliquée qu'aux hôtes.
- Un utilisateur bénéficiant d'un privilège contenant l'autorisation peut effectuer toutes les opérations suivantes sur une entrée :
  - Écrire
  - Lire
  - Recherche
  - Comparer
  - Ajouter
  - Supprimer

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

Procédure

1. Naviguez jusqu'au répertoire ***~/MyPlaybooks/*** répertoire :

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

2. Faites une copie du fichier **permission-present.yml** situé dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/permission/**:

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/permission/permission-present.yml
permission-present-copy.yml
```

3. Ouvrez le fichier **permission-present-copy.yml** Ansible playbook pour l'éditer.
4. Adaptez le fichier en définissant les variables suivantes dans la section **ipapermission** task :
   - Adaptez le site **name** de la tâche pour qu'il corresponde à votre cas d'utilisation.
   - Définissez la variable **ipaadmin_password** avec le mot de passe de l'administrateur IdM.
   - Attribuez à la variable **name** le nom de l'autorisation.
   - Fixer la variable **object_type** à **host**.
   - Fixer la variable **right** à **all**.

   Il s'agit du fichier playbook Ansible modifié pour l'exemple actuel :

```
---
- name: Permission present example
  hosts: ipaserver

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Ensure that the "MyPermission" permission is present
    ipapermission:
      ipaadmin_password: "{{ ipaadmin_password }}"
```

```
name: MyPermission
object_type: host
right: all
```

5. Enregistrer le fichier.

6. Exécutez le playbook Ansible. Spécifiez le fichier du livre de jeu, le fichier contenant le mot de passe protégeant le fichier **secret.yml** et le fichier d'inventaire :

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

33.2. UTILISER ANSIBLE POUR S'ASSURER QU'UNE PERMISSION RBAC AVEC UN ATTRIBUT EST PRÉSENTE

En tant qu'administrateur système de la gestion des identités (IdM), vous pouvez personnaliser le contrôle d'accès basé sur les rôles (RBAC) de l'IdM.

La procédure suivante décrit comment utiliser un playbook Ansible pour s'assurer qu'une permission est présente dans IdM afin qu'elle puisse être ajoutée à un privilège. L'exemple décrit comment garantir l'état cible suivant :

- L'autorisation **MyPermission** existe.
- L'autorisation **MyPermission** ne peut être utilisée que pour ajouter des hôtes.
- Un utilisateur bénéficiant d'un privilège contenant l'autorisation peut effectuer toutes les opérations suivantes sur une entrée d'hôte :
  - Écrire
  - Lire
  - Recherche
  - Comparer
  - Ajouter
  - Supprimer
- Les entrées d'hôte créées par un utilisateur bénéficiant d'un privilège contenant l'autorisation **MyPermission** peuvent avoir une valeur **description**.

NOTE

Le type d'attribut que vous pouvez spécifier lors de la création ou de la modification d'une autorisation n'est pas limité par le schéma LDAP de l'IdM. Toutefois, le fait de spécifier, par exemple, **attrs: car_licence** si **object_type** est **host** plus tard entraîne le message d'erreur **ipa: ERROR: attribute "car-license" not allowed** lorsque vous essayez d'exercer l'autorisation et d'ajouter une valeur de permis de conduire spécifique à un hôte.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

Procédure

1. Naviguez jusqu'au répertoire ***~/MyPlaybooks/*** répertoire :

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

2. Faites une copie du fichier **permission-present.yml** situé dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/permission/**:

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/permission/permission-present.yml
permission-present-with-attribute.yml
```

3. Ouvrez le fichier **permission-present-with-attribute.yml** Ansible playbook pour l'éditer.
4. Adaptez le fichier en définissant les variables suivantes dans la section **ipapermission** task :
   - Adaptez le site **name** de la tâche pour qu'il corresponde à votre cas d'utilisation.
   - Définissez la variable **ipaadmin_password** avec le mot de passe de l'administrateur IdM.
   - Attribuez à la variable **name** le nom de l'autorisation.
   - Fixer la variable **object_type** à **host**.
   - Fixer la variable **right** à **all**.
   - Fixer la variable **attrs** à **description**.

   Il s'agit du fichier playbook Ansible modifié pour l'exemple actuel :

```
---
- name: Permission present example
  hosts: ipaserver

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Ensure that the "MyPermission" permission is present with an attribute
    ipapermission:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: MyPermission
```

```
    object_type: host
    right: all
    attrs: description
```

5. Enregistrer le fichier.

6. Exécutez le playbook Ansible. Spécifiez le fichier du livre de jeu, le fichier contenant le mot de passe protégeant le fichier **secret.yml** et le fichier d'inventaire :

```
$ ansible-playbook --vault-password-file=password_file -v -i inventory permission-present-with-attribute.yml
```

Ressources supplémentaires

- Voir Schéma des utilisateurs et des groupes dans *Linux Domain Identity, Authentication and Policy Guide* dans RHEL 7.

33.3. UTILISER ANSIBLE POUR S'ASSURER QU'UNE PERMISSION RBAC EST ABSENTE

En tant qu'administrateur système de la gestion des identités (IdM), vous pouvez personnaliser le contrôle d'accès basé sur les rôles (RBAC) de l'IdM.

La procédure suivante décrit comment utiliser un manuel de jeu Ansible pour s'assurer qu'une permission est absente dans IdM et qu'elle ne peut pas être ajoutée à un privilège.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

Procédure

1. Naviguez jusqu'au répertoire ***~/MyPlaybooks/*** répertoire :

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

2. Faites une copie du fichier **permission-absent.yml** situé dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/permission/**:

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/permission/permission-absent.yml permission-absent-copy.yml
```

3. Ouvrez le fichier **permission-absent-copy.yml** Ansible playbook pour l'éditer.
4. Adaptez le fichier en définissant les variables suivantes dans la section **ipapermission** task :
   - Adaptez le site **name** de la tâche pour qu'il corresponde à votre cas d'utilisation.
   - Définissez la variable **ipaadmin_password** avec le mot de passe de l'administrateur IdM.
   - Attribuez à la variable **name** le nom de l'autorisation.

   Il s'agit du fichier playbook Ansible modifié pour l'exemple actuel :

```
---
- name: Permission absent example
  hosts: ipaserver

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Ensure that the "MyPermission" permission is absent
    ipapermission:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: MyPermission
      state: absent
```

5. Enregistrer le fichier.
6. Exécutez le playbook Ansible. Spécifiez le fichier du livre de jeu, le fichier contenant le mot de passe protégeant le fichier **secret.yml** et le fichier d'inventaire :

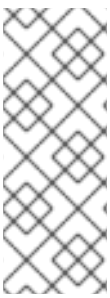
```
$ ansible-playbook --vault-password-file=password_file -v -i inventory permission-absent-copy.yml
```

33.4. UTILISER ANSIBLE POUR S'ASSURER QU'UN ATTRIBUT EST MEMBRE D'UNE PERMISSION IDM RBAC

En tant qu'administrateur système de la gestion des identités (IdM), vous pouvez personnaliser le contrôle d'accès basé sur les rôles (RBAC) de l'IdM.

La procédure suivante décrit comment utiliser un playbook Ansible pour s'assurer qu'un attribut est membre d'une permission RBAC dans IdM. Par conséquent, un utilisateur disposant de la permission peut créer des entrées dotées de l'attribut.

L'exemple décrit comment garantir que les entrées d'hôte créées par un utilisateur disposant d'un privilège contenant l'autorisation **MyPermission** peuvent avoir les valeurs **gecos** et **description**.



NOTE

Le type d'attribut que vous pouvez spécifier lors de la création ou de la modification d'une autorisation n'est pas limité par le schéma LDAP de l'IdM. Toutefois, le fait de spécifier, par exemple, **attrs: car_licence** si **object_type** est **host** plus tard entraîne le message d'erreur **ipa: ERROR: attribute "car-license" not allowed** lorsque vous essayez d'exercer l'autorisation et d'ajouter une valeur de permis de conduire spécifique à un hôte.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.
- L'autorisation **MyPermission** existe.

Procédure

1. Naviguez jusqu'au répertoire ***~/MyPlaybooks/*** répertoire :

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

2. Faites une copie du fichier **permission-member-present.yml** situé dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/permission/**:

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/permission/permission-member-present.yml permission-member-present-copy.yml
```

3. Ouvrez le fichier **permission-member-present-copy.yml** Ansible playbook pour l'éditer.
4. Adaptez le fichier en définissant les variables suivantes dans la section **ipapermission** task :
   - Adaptez le site **name** de la tâche pour qu'il corresponde à votre cas d'utilisation.
   - Définissez la variable **ipaadmin_password** avec le mot de passe de l'administrateur IdM.
   - Attribuez à la variable **name** le nom de l'autorisation.
   - Attribuer la liste **attrs** aux variables **description** et **gecos**.
   - Assurez-vous que la variable **action** est définie sur **member**.

   Il s'agit du fichier playbook Ansible modifié pour l'exemple actuel :

```
---
- name: Permission member present example
  hosts: ipaserver

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Ensure that the "gecos" and "description" attributes are present in "MyPermission"
```

```
ipapermission:
  ipaadmin_password: "{{ ipaadmin_password }}"
  name: MyPermission
  attrs:
  - description
  - gecos
  action: member
```

5. Enregistrer le fichier.

6. Exécutez le playbook Ansible. Spécifiez le fichier du livre de jeu, le fichier contenant le mot de passe protégeant le fichier **secret.yml** et le fichier d'inventaire :

```
$ ansible-playbook --vault-password-file=password_file -v -i inventory permission-member-present-copy.yml
```

33.5. UTILISER ANSIBLE POUR S'ASSURER QU'UN ATTRIBUT N'EST PAS MEMBRE D'UNE PERMISSION RBAC IDM

En tant qu'administrateur système de la gestion des identités (IdM), vous pouvez personnaliser le contrôle d'accès basé sur les rôles (RBAC) de l'IdM.

La procédure suivante décrit comment utiliser un playbook Ansible pour s'assurer qu'un attribut n'est pas membre d'une permission RBAC dans IdM. Par conséquent, lorsqu'un utilisateur disposant de cette permission crée une entrée dans IdM LDAP, cette entrée ne peut pas avoir de valeur associée à l'attribut.

L'exemple décrit comment assurer l'état suivant de la cible :

- L'autorisation **MyPermission** existe.
- Les entrées d'hôte créées par un utilisateur disposant d'un privilège contenant l'autorisation **MyPermission** ne peuvent pas avoir l'attribut **description**.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.
- L'autorisation **MyPermission** existe.

Procédure

1. Naviguez jusqu'au répertoire ***~/MyPlaybooks/*** répertoire :

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

2. Faites une copie du fichier **permission-member-absent.yml** situé dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/permission/**:

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/permission/permission-member-absent.yml permission-member-absent-copy.yml
```

3. Ouvrez le fichier **permission-member-absent-copy.yml** Ansible playbook pour l'éditer.

4. Adaptez le fichier en définissant les variables suivantes dans la section **ipapermission** task :

   - Adaptez le site **name** de la tâche pour qu'il corresponde à votre cas d'utilisation.
   - Définissez la variable **ipaadmin_password** avec le mot de passe de l'administrateur IdM.
   - Attribuez à la variable **name** le nom de l'autorisation.
   - Fixer la variable **attrs** à **description**.
   - Fixer la variable **action** à **member**.
   - Assurez-vous que la variable **state** est définie comme suit **absent**

   Il s'agit du fichier playbook Ansible modifié pour l'exemple actuel :

```
---
- name: Permission absent example
  hosts: ipaserver

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Ensure that an attribute is not a member of "MyPermission"
    ipapermission:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: MyPermission
      attrs: description
      action: member
      state: absent
```

5. Enregistrer le fichier.

6. Exécutez le playbook Ansible. Spécifiez le fichier du livre de jeu, le fichier contenant le mot de passe protégeant le fichier **secret.yml** et le fichier d'inventaire :

```
$ ansible-playbook --vault-password-file=password_file -v -i inventory permission-member-absent-copy.yml
```

33.6. UTILISER ANSIBLE POUR RENOMMER UNE PERMISSION IDM RBAC

En tant qu'administrateur système de la gestion des identités (IdM), vous pouvez personnaliser le contrôle d'accès basé sur les rôles de l'IdM.

La procédure suivante décrit comment utiliser un playbook Ansible pour renommer une autorisation. L'exemple décrit comment renommer **MyPermission** en **MyNewPermission**.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.
- Le site **MyPermission** existe dans l'IdM.
- Le site **MyNewPermission** n'existe pas dans l'IdM.

Procédure

1. Naviguez jusqu'au répertoire ***~/MyPlaybooks/*** répertoire :

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

2. Faites une copie du fichier **permission-renamed.yml** situé dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/permission/**:

```
$ cp /usr/share/doc/ansible-freeipa/playbooks/permission/permission-renamed.yml
permission-renamed-copy.yml
```

3. Ouvrez le fichier **permission-renamed-copy.yml** Ansible playbook pour l'éditer.
4. Adaptez le fichier en définissant les variables suivantes dans la section **ipapermission** task :
   - Adaptez le site **name** de la tâche pour qu'il corresponde à votre cas d'utilisation.
   - Définissez la variable **ipaadmin_password** avec le mot de passe de l'administrateur IdM.
   - Attribuez à la variable **name** le nom de l'autorisation.

   Il s'agit du fichier playbook Ansible modifié pour l'exemple actuel :

```
---
- name: Permission present example
  hosts: ipaserver

  vars_files:
```

```
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Rename the "MyPermission" permission
    ipapermission:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: MyPermission
      rename: MyNewPermission
      state: renamed
```

5. Enregistrer le fichier.

6. Exécutez le playbook Ansible. Spécifiez le fichier du livre de jeu, le fichier contenant le mot de passe protégeant le fichier **secret.yml** et le fichier d'inventaire :

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

33.7. RESSOURCES SUPPLÉMENTAIRES

- Voir les autorisations dans IdM.
- Voir les privilèges dans l'IdM.
- Voir le fichier **README-permission** disponible dans le répertoire **/usr/share/doc/ansible-freeipa/**.
- Voir les exemples de playbooks dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/ipapermission**.

CHAPITRE 34. UTILISATION D'UNE VUE ID POUR REMPLACER UNE VALEUR D'ATTRIBUT UTILISATEUR SUR UN CLIENT IDM

Si un utilisateur de Identity Management (IdM) souhaite remplacer certains de ses attributs d'utilisateur ou de groupe stockés dans le serveur LDAP IdM, par exemple le nom de connexion, le répertoire personnel, le certificat utilisé pour l'authentification ou les clés **SSH**, vous pouvez, en tant qu'administrateur IdM, redéfinir ces valeurs pour un client IdM spécifique, en utilisant les vues IdM ID. Par exemple, vous pouvez spécifier un répertoire personnel différent pour un utilisateur sur le client IdM que l'utilisateur utilise le plus souvent pour se connecter à IdM.

Ce chapitre explique comment redéfinir une valeur d'attribut POSIX associée à un utilisateur IdM sur un hôte inscrit à IdM en tant que client. Plus précisément, ce chapitre explique comment redéfinir le nom de connexion et le répertoire personnel de l'utilisateur.

Ce chapitre comprend les sections suivantes :

- Vues de l'ID
- Impact négatif potentiel des opinions de l'ID sur les performances du SSSD
- Attributs qu'une vue d'identification peut remplacer
- Obtenir de l'aide pour les commandes de la vue ID
- Utilisation d'une vue ID pour remplacer le nom de connexion d'un utilisateur IdM sur un hôte spécifique
- Modification d'une vue IdM ID
- Ajout d'une vue ID pour remplacer le répertoire personnel d'un utilisateur IdM sur un client IdM
- Application d'une vue ID à un groupe d'hôtes IdM

34.1. VUES DE L'ID

Dans le cadre de la gestion des identités (IdM), une vue ID est une vue côté client IdM qui spécifie les informations suivantes :

- Nouvelles valeurs pour les attributs d'utilisateur ou de groupe POSIX définis de manière centralisée
- L'hôte ou les hôtes du client sur lesquels les nouvelles valeurs s'appliquent.

Une vue ID contient une ou plusieurs dérogations. Une dérogation est un remplacement spécifique d'une valeur d'attribut POSIX définie de manière centralisée.

Vous ne pouvez définir une vue ID pour un client IdM que de manière centralisée sur les serveurs IdM. Vous ne pouvez pas configurer localement des dérogations côté client pour un client IdM.

Par exemple, vous pouvez utiliser les vues d'identification pour atteindre les objectifs suivants :

- Définir des valeurs d'attribut différentes pour des environnements différents. Par exemple, vous pouvez permettre à l'administrateur IdM ou à un autre utilisateur IdM d'avoir différents répertoires personnels sur différents clients IdM : vous pouvez configurer **/home/encrypted/username** comme répertoire personnel de cet utilisateur sur un client IdM et **/dropbox/username** sur un autre client. L'utilisation des vues ID dans cette situation est

pratique, car sinon, par exemple, la modification de **fallback_homedir**, **override_homedir** ou d'autres variables de répertoire personnel dans le fichier **/etc/sssd/sssd.conf** du client affecterait tous les utilisateurs. Voir Ajouter une vue ID pour remplacer le répertoire personnel d'un utilisateur IdM sur un client IdM pour un exemple de procédure.

- Remplacer une valeur d'attribut générée précédemment par une valeur différente, par exemple remplacer l'UID d'un utilisateur. Cette possibilité peut s'avérer utile lorsque vous souhaitez effectuer une modification à l'échelle du système qui serait autrement difficile à réaliser du côté de LDAP, par exemple faire de 1009 l'UID d'un utilisateur IdM. Les plages d'ID IdM, qui sont utilisées pour générer l'UID d'un utilisateur IdM, ne commencent jamais aussi bas que 1000 ou même 10000. S'il existe une raison pour qu'un utilisateur IdM se fasse passer pour un utilisateur local avec l'UID 1009 sur tous les clients IdM, vous pouvez utiliser les vues ID pour remplacer l'UID de cet utilisateur IdM qui a été généré lorsque l'utilisateur a été créé dans l'IdM.

IMPORTANT

Vous ne pouvez appliquer les vues ID qu'aux clients IdM, et non aux serveurs IdM.

Ressources supplémentaires

- Utilisation des vues d'identification pour les utilisateurs d'Active Directory
- Vues côté client SSSD

34.2. IMPACT NÉGATIF POTENTIEL DES OPINIONS DE L'ID SUR LES PERFORMANCES DU SSSD

Lorsque vous définissez une vue ID, IdM place la valeur de remplacement souhaitée dans le cache SSSD (System Security Services Daemon) du serveur IdM. Le SSSD fonctionnant sur un client IdM récupère alors la valeur de remplacement dans le cache du serveur.

L'application d'une vue ID peut avoir un impact négatif sur les performances du System Security Services Daemon (SSSD), car certaines optimisations et vues ID ne peuvent pas être exécutées en même temps. Par exemple, les vues ID empêchent SSSD d'optimiser le processus de recherche des groupes sur le serveur :

- Avec les vues ID, SSSD doit vérifier chaque membre de la liste renvoyée des noms des membres du groupe si le nom du groupe est remplacé.
- Sans les vues d'identification, SSSD ne peut collecter les noms d'utilisateur qu'à partir de l'attribut membre de l'objet groupe.

Cet effet négatif est particulièrement visible lorsque le cache SSSD est vide ou après avoir effacé le cache, ce qui rend toutes les entrées invalides.

34.3. ATTRIBUTS QU'UNE VUE D'IDENTIFICATION PEUT REMPLACER

Les vues ID consistent en des substitutions d'ID d'utilisateur et de groupe. Les dérogations définissent les nouvelles valeurs des attributs POSIX.

Les substitutions d'ID d'utilisateur et de groupe peuvent définir de nouvelles valeurs pour les attributs POSIX suivants :

Attributs de l'utilisateur

- Nom d'utilisateur (**uid**)

- Nom d'utilisateur (**uid**)
- Entrée GECOS (**gecos**)
- Numéro UID (**uidNumber**)
- Numéro GID (**gidNumber**)
- Shell de connexion (**loginShell**)
- Répertoire personnel (**homeDirectory**)
- Clés publiques SSH (**ipaSshPubkey**)
- Certificat (**userCertificate**)

Attributs du groupe

- Nom du groupe (**cn**)
- Numéro GID du groupe (**gidNumber**)

34.4. OBTENIR DE L'AIDE POUR LES COMMANDES DE LA VUE ID

Vous pouvez obtenir de l'aide pour les commandes impliquant des vues d'identification de la gestion d'identité (IdM) sur l'interface de ligne de commande (CLI) de l'IdM.

Conditions préalables

- Vous avez obtenu un ticket Kerberos pour un utilisateur IdM.

Procédure

- Pour afficher toutes les commandes utilisées pour gérer les vues d'identification et les dérogations :

```
$ ipa help idviews
ID Views

Manage ID Views

IPA allows to override certain properties of users and groups[...]
[...]
Topic commands:
  idoverridegroup-add          Add a new Group ID override
  idoverridegroup-del          Delete a Group ID override
[...]
```

- Pour afficher l'aide détaillée d'une commande particulière, ajoutez l'option **--help** à la commande :

```
$ ipa idview-add --help
Usage: ipa [global-options] idview-add NAME [options]

Add a new ID View.
```

```
Options:
  -h, --help      show this help message and exit
  --desc=STR      Description
[...]
```

34.5. UTILISATION D'UNE VUE ID POUR REMPLACER LE NOM DE CONNEXION D'UN UTILISATEUR IDM SUR UN HÔTE SPÉCIFIQUE

Cette section décrit comment, en tant qu'administrateur du système de gestion des identités (IdM), vous pouvez créer une vue ID pour un client IdM spécifique qui remplace une valeur d'attribut POSIX associée à un utilisateur IdM spécifique. La procédure utilise l'exemple d'une vue ID qui permet à un utilisateur IdM nommé **idm_user** de se connecter à un client IdM nommé **host1** en utilisant le nom de connexion **user_1234**.

Conditions préalables

- Vous êtes connecté en tant qu'administrateur IdM.

Procédure

1. Créez une nouvelle vue d'identification. Par exemple, pour créer une vue ID nommée **example_for_host1**:

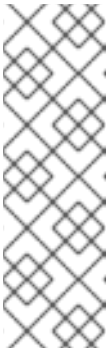
```
$ ipa idview-add example_for_host1
----------------------------
Added ID View "example_for_host1"
----------------------------
  ID View Name: example_for_host1
```

2. Ajoutez une dérogation pour l'utilisateur à la vue **example_for_host1** ID. Pour remplacer le login de l'utilisateur :

   - Entrez la commande **ipa idoverrideuser-add**
   - Ajouter le nom de la vue ID
   - Ajouter le nom d'utilisateur, également appelé l'ancre
   - Ajouter l'option **--login**:

```
$ ipa idoverrideuser-add example_for_host1 idm_user --login=user_1234
----------------------------
Added User ID override "idm_user"
----------------------------
  Anchor to override: idm_user
  User login: user_1234
```

   Pour obtenir la liste des options disponibles, exécutez ipa idoverrideuser-add --help.



NOTE

La commande **ipa idoverrideuser-add --certificate** remplace tous les certificats existants pour le compte dans la vue ID spécifiée. Pour ajouter un certificat supplémentaire, utilisez plutôt la commande **ipa idoverrideuser-add-cert**:

```
$ ipa idoverrideuser-add-cert example_for_host1 user --certificate="MIIEATCC..."
```

3. Facultatif : La commande **ipa idoverrideuser-mod** vous permet de spécifier de nouvelles valeurs d'attribut pour une dérogation utilisateur existante.

4. Appliquer **example_for_host1** à l'hôte **host1.idm.example.com**:

```
$ ipa idview-apply example_for_host1 --hosts=host1.idm.example.com
-----------------------------
Applied ID View "example_for_host1"
-----------------------------
hosts: host1.idm.example.com
---------------------------------------------
Number of hosts the ID View was applied to: 1
---------------------------------------------
```

NOTE

La commande **ipa idview-apply** accepte également l'option **--hostgroups**. Cette option applique la vue ID aux hôtes qui appartiennent au groupe d'hôtes spécifié, mais n'associe pas la vue ID au groupe d'hôtes lui-même. Au lieu de cela, l'option **--hostgroups** développe les membres du groupe d'hôtes spécifié et applique l'option **--hosts** individuellement à chacun d'entre eux.

Cela signifie que si un hôte est ajouté au groupe d'hôtes dans le futur, la vue ID ne s'applique pas au nouvel hôte.

5. Pour appliquer immédiatement la nouvelle configuration au système **host1.idm.example.com**:

   a. Accédez au système par SSH en tant que root :

```
$ ssh root@host1
Password:
```

   b. Effacer le cache SSSD :

```
root@host1 ~]# sss_cache -E
```

   c. Redémarrez le démon SSSD :

```
root@host1 ~]# systemctl restart sssd
```

Verification steps

- Si vous disposez des informations d'identification de **user_1234**, vous pouvez les utiliser pour vous connecter à l'IdM sur **host1**:

  1. SSH à **host1** en utilisant **user_1234** comme nom de connexion :

     ```
     [root@r8server ~]# ssh user_1234@host1.idm.example.com
     Password:

     Last login: Sun Jun 21 22:34:25 2020 from 192.168.122.229
     [user_1234@host1 ~]$
     ```

  2. Affiche le répertoire de travail :

     ```
     [user_1234@host1 ~]$ pwd
     /home/idm_user/
     ```

- Par ailleurs, si vous disposez d'informations d'identification root sur **host1**, vous pouvez les utiliser pour vérifier la sortie de la commande **id** pour **idm_user** et **user_1234**:

  ```
  [root@host1 ~]# id idm_user
  uid=779800003(user_1234) gid=779800003(idm_user) groups=779800003(idm_user)
  [root@host1 ~]# user_1234
  uid=779800003(user_1234) gid=779800003(idm_user) groups=779800003(idm_user)
  ```

34.6. MODIFICATION D'UNE VUE IDM ID

Une vue ID dans Identity Management (IdM) remplace une valeur d'attribut POSIX associée à un utilisateur IdM spécifique. Cette section explique comment modifier une vue d'identification existante. Plus précisément, elle décrit comment modifier une vue ID pour permettre à l'utilisateur nommé **idm_user** d'utiliser le répertoire **/home/user_1234/** comme répertoire personnel de l'utilisateur au lieu de **/home/idm_user/** sur le client IdM **host1.idm.example.com**.

Conditions préalables

- Vous avez un accès root à **host1.idm.example.com**.
- Vous êtes connecté en tant qu'utilisateur disposant des privilèges requis, par exemple **admin**.
- Vous avez une vue ID configurée pour **idm_user** qui s'applique au client IdM **host1**.

Procédure

1. En tant que root, créez le répertoire que vous voulez que **idm_user** utilise sur **host1.idm.example.com** comme répertoire personnel de l'utilisateur :

   ```
   [root@host1 /]# mkdir /home/user_1234/
   ```

2. Modifier la propriété du répertoire :

   ```
   [root@host1 /]# chown idm_user:idm_user /home/user_1234/
   ```

3. Afficher la vue d'identification, y compris les hôtes auxquels la vue d'identification est actuellement appliquée. Pour afficher la vue ID nommée **example_for_host1**:

```
$ ipa idview-show example_for_host1 --all
  dn: cn=example_for_host1,cn=views,cn=accounts,dc=idm,dc=example,dc=com
  ID View Name: example_for_host1
  User object override: idm_user
  Hosts the view applies to: host1.idm.example.com
  objectclass: ipaIDView, top, nsContainer
```

La sortie montre que la vue ID s'applique actuellement à **host1.idm.example.com**.

4. Modifiez l'option de remplacement de l'utilisateur dans la vue **example_for_host1** ID. Pour remplacer le répertoire personnel de l'utilisateur :

   - Entrez la commande **ipa idoverrideuser-add**
   - Ajouter le nom de la vue ID
   - Ajouter le nom d'utilisateur, également appelé l'ancre
   - Ajouter l'option **--homedir**:

     ```
     $ ipa idoverrideuser-mod example_for_host1 idm_user --
     homedir=/home/user_1234
     ----------------------------
     Modified a User ID override "idm_user"
     ----------------------------
       Anchor to override: idm_user
       User login: user_1234
       Home directory: /home/user_1234/
     ```

   Pour obtenir une liste des options disponibles, exécutez **ipa idoverrideuser-mod --help**.

5. Pour appliquer immédiatement la nouvelle configuration au système **host1.idm.example.com**:

   a. Accédez au système par SSH en tant que root :

      ```
      $ ssh root@host1
      Password:
      ```

   b. Effacer le cache SSSD :

      ```
      root@host1 ~]# sss_cache -E
      ```

   c. Redémarrez le démon SSSD :

   ```
   root@host1 ~]# systemctl restart sssd
   ```

Verification steps

1. **SSH** à **host1** comme **idm_user**:

   ```
   [root@r8server ~]# ssh idm_user@host1.idm.example.com
   Password:
   ```

```
Last login: Sun Jun 21 22:34:25 2020 from 192.168.122.229
[user_1234@host1 ~]$
```

2. Imprime le répertoire de travail :

```
[user_1234@host1 ~]$ pwd
/home/user_1234/
```

Ressources supplémentaires

- Définition d'attributs globaux pour un utilisateur AD en modifiant la vue de confiance par défaut

34.7. AJOUT D'UNE VUE ID POUR REMPLACER LE RÉPERTOIRE PERSONNEL D'UN UTILISATEUR IDM SUR UN CLIENT IDM

Une vue ID dans Identity Management (IdM) remplace une valeur d'attribut POSIX associée à un utilisateur IdM spécifique. Cette section décrit comment créer une vue ID qui s'applique à **idm_user** sur un client IdM nommé **host1** pour permettre à l'utilisateur d'utiliser le répertoire **/home/user_1234/** comme répertoire personnel de l'utilisateur au lieu de **/home/idm_user/**.

Conditions préalables

- Vous avez un accès root à **host1.idm.example.com**.
- Vous êtes connecté en tant qu'utilisateur disposant des privilèges requis, par exemple **admin**.

Procédure

1. En tant que root, créez le répertoire que vous voulez que **idm_user** utilise sur **host1.idm.example.com** comme répertoire personnel de l'utilisateur :

```
[root@host1 /]# mkdir /home/user_1234/
```

2. Modifier la propriété du répertoire :

```
[root@host1 /]# chown idm_user:idm_user /home/user_1234/
```

3. Créez une vue d'identification. Par exemple, pour créer une vue ID nommée **example_for_host1**:

```
$ ipa idview-add example_for_host1
---------------------------
Added ID View "example_for_host1"
---------------------------
  ID View Name: example_for_host1
```

4. Ajoutez une dérogation pour l'utilisateur à la vue **example_for_host1** ID. Pour remplacer le répertoire personnel de l'utilisateur :
   - Entrez la commande **ipa idoverrideuser-add**
   - Ajouter le nom de la vue ID
   - Ajouter le nom d'utilisateur, également appelé l'ancre

- Ajouter l'option **--homedir**:

```
$ ipa idoverrideuser-add example_for_host1 idm_user --homedir=/home/user_1234
----------------------------
Added User ID override "idm_user"
----------------------------
  Anchor to override: idm_user
  Home directory: /home/user_1234/
```

5. Appliquer **example_for_host1** à l'hôte **host1.idm.example.com**:

```
$ ipa idview-apply example_for_host1 --hosts=host1.idm.example.com
----------------------------
Applied ID View "example_for_host1"
----------------------------
hosts: host1.idm.example.com
--------------------------------------------
Number of hosts the ID View was applied to: 1
--------------------------------------------
```

NOTE

La commande **ipa idview-apply** accepte également l'option **--hostgroups**. Cette option applique la vue ID aux hôtes qui appartiennent au groupe d'hôtes spécifié, mais n'associe pas la vue ID au groupe d'hôtes lui-même. Au lieu de cela, l'option **--hostgroups** développe les membres du groupe d'hôtes spécifié et applique l'option **--hosts** individuellement à chacun d'entre eux.

Cela signifie que si un hôte est ajouté au groupe d'hôtes dans le futur, la vue ID ne s'applique pas au nouvel hôte.

6. Pour appliquer immédiatement la nouvelle configuration au système **host1.idm.example.com**:

   a. Accédez au système par SSH en tant que root :

```
$ ssh root@host1
Password:
```

   b. Effacer le cache SSSD :

```
root@host1 ~]# sss_cache -E
```

   c. Redémarrez le démon SSSD :

```
root@host1 ~]# systemctl restart sssd
```

Verification steps

1. **SSH** à **host1** comme **idm_user**:

```
[root@r8server ~]# ssh idm_user@host1.idm.example.com
Password:
Activate the web console with: systemctl enable --now cockpit.socket
```

```
Last login: Sun Jun 21 22:34:25 2020 from 192.168.122.229
[idm_user@host1 /]$
```

2. Imprime le répertoire de travail :

```
[idm_user@host1 /]$ pwd
/home/user_1234/
```

Ressources supplémentaires

- Remplacement des attributs de la vue de confiance par défaut pour un utilisateur AD sur un client IdM avec une vue ID

34.8. APPLICATION D'UNE VUE ID À UN GROUPE D'HÔTES IDM

La commande **ipa idview-apply** accepte l'option **--hostgroups**. Cependant, l'option agit comme une opération unique qui applique la vue ID aux hôtes qui appartiennent actuellement au groupe d'hôtes spécifié, mais n'associe pas dynamiquement la vue ID au groupe d'hôtes lui-même. L'option **--hostgroups** étend les membres du groupe d'hôtes spécifié et applique l'option **--hosts** individuellement à chacun d'entre eux.

Si vous ajoutez ultérieurement un nouvel hôte au groupe d'hôtes, vous devez appliquer manuellement la vue ID au nouvel hôte, en utilisant la commande **ipa idview-apply** avec l'option **--hosts**.

De même, si vous supprimez un hôte d'un groupe d'hôtes, la vue ID est toujours affectée à l'hôte après la suppression. Pour annuler l'affectation de la vue ID de l'hôte supprimé, vous devez exécuter la commande **ipa idview-unapply *id_view_name* --hosts=*name_of_the_removed_host*** pour annuler l'application de la vue ID de l'hôte supprimé.

Cette section décrit comment atteindre les objectifs suivants :

1. Comment créer un groupe d'hôtes et y ajouter des hôtes.
2. Comment appliquer une vue d'identification au groupe d'hôtes.
3. Comment ajouter un nouvel hôte au groupe d'hôtes et appliquer la vue ID au nouvel hôte.

Conditions préalables

- Assurez-vous que la vue ID que vous souhaitez appliquer au groupe d'hôtes existe dans IdM. Par exemple, pour créer une vue ID afin de remplacer le GID d'un utilisateur AD, voir Remplacer les attributs de la vue Trust par défaut pour un utilisateur AD sur un client IdM avec une vue ID

Procédure

1. Créez un groupe d'hôtes et ajoutez-y des hôtes :
   a. Créez un groupe d'hôtes. Par exemple, pour créer un groupe d'hôtes nommé **baltimore**:

```
[root@server ~]# ipa hostgroup-add --desc="Baltimore hosts" baltimore
--------------------------
Added hostgroup "baltimore"
```

```
---------------------------
Host-group: baltimore
Description: Baltimore hosts
```

b. Ajoutez des hôtes au groupe d'hôtes. Par exemple, pour ajouter les hôtes **host102** et **host103** au groupe d'hôtes **baltimore**:

```
[root@server ~]# ipa hostgroup-add-member --hosts={host102,host103} baltimore
Host-group: baltimore
Description: Baltimore hosts
Member hosts: host102.idm.example.com, host103.idm.example.com
-------------------------
Number of members added 2
-------------------------
```

2. Appliquer une vue ID aux hôtes du groupe d'hôtes. Par exemple, pour appliquer la vue d'identification **example_for_host1** au groupe d'hôtes **baltimore**:

```
[root@server ~]# ipa idview-apply --hostgroups=baltimore
ID View Name: example_for_host1
----------------------------------------
Applied ID View "example_for_host1"
----------------------------------------
  hosts: host102.idm.example.com, host103.idm.example.com
------------------------------------------
Number of hosts the ID View was applied to: 2
------------------------------------------
```

3. Ajoutez un nouvel hôte au groupe d'hôtes et appliquez la vue ID au nouvel hôte :

a. Ajoutez un nouvel hôte au groupe d'hôtes. Par exemple, pour ajouter l'hôte **somehost.idm.example.com** au groupe d'hôtes **baltimore**:

```
[root@server ~]# ipa hostgroup-add-member --hosts=somehost.idm.example.com
baltimore
  Host-group: baltimore
  Description: Baltimore hosts
  Member hosts:  host102.idm.example.com,
host103.idm.example.com,somehost.idm.example.com
-------------------------
Number of members added 1
-------------------------
```

b. En option, affichez les informations sur la vue d'identification. Par exemple, pour afficher les détails de la vue d'identification **example_for_host1**:

```
[root@server ~]# ipa idview-show example_for_host1 --all
  dn: cn=example_for_host1,cn=views,cn=accounts,dc=idm,dc=example,dc=com
  ID View Name: example_for_host1
[...]
  Hosts the view applies to: host102.idm.example.com, host103.idm.example.com
  objectclass: ipaIDView, top, nsContainer
```

La sortie montre que la vue ID n'est pas appliquée à **somehost.idm.example.com**, l'hôte nouvellement ajouté au groupe d'hôtes **baltimore**.

c. Appliquez la vue ID au nouvel hôte. Par exemple, pour appliquer la vue d'identification **example_for_host1** à **somehost.idm.example.com**:

```
[root@server ~]# ipa idview-apply --host=somehost.idm.example.com
ID View Name: example_for_host1
-----------------------------------------
Applied ID View "example_for_host1"
-----------------------------------------
  hosts: somehost.idm.example.com
-------------------------------------------------
Number of hosts the ID View was applied to: 1
-------------------------------------------
```

Verification steps

- Affichez à nouveau les informations de la vue ID :

```
[root@server ~]# ipa idview-show example_for_host1 --all
  dn: cn=example_for_host1,cn=views,cn=accounts,dc=idm,dc=example,dc=com
  ID View Name: example_for_host1
[...]
  Hosts the view applies to: host102.idm.example.com, host103.idm.example.com,
somehost.idm.example.com
  objectclass: ipaIDView, top, nsContainer
```

La sortie montre que la vue ID est maintenant appliquée à **somehost.idm.example.com**, l'hôte nouvellement ajouté au groupe d'hôtes **baltimore**.

34.9. MIGRATION DES DOMAINES NIS VERS LA GESTION DES IDENTITÉS

Vous pouvez utiliser les vues ID pour définir des UID et des GID spécifiques aux hôtes existants afin d'éviter de modifier les autorisations des fichiers et des répertoires lors de la migration des domaines NIS vers IdM.

Conditions préalables

- Vous vous êtes authentifié en tant qu'administrateur à l'aide de la commande **kinit admin**.

Procédure

1. Ajouter des utilisateurs et des groupes dans le domaine IdM.

   a. Créez des utilisateurs à l'aide de la commande **ipa user-add**. Pour plus d'informations, voir : Ajouter des utilisateurs à IdM.

   b. Créez des groupes à l'aide de la commande **ipa group-add**. Pour plus d'informations, voir : Ajouter des groupes à IdM.

2. Remplacer les identifiants générés par l'IdM lors de la création de l'utilisateur :

   a. Créez une nouvelle vue ID à l'aide de la commande **ipa idview-add**. Pour plus d'informations, voir : Obtenir de l'aide pour les commandes de vues d'identification .

b. Ajoutez des dérogations d'ID pour les utilisateurs et les groupes à la vue ID en utilisant respectivement **ipa idoverrideuser-add** et **idoverridegroup-add**.

3. Attribuer la vue ID aux hôtes spécifiques en utilisant la commande **ipa idview-apply**.

4. Déclasser les domaines NIS.

Vérification

1. Pour vérifier si tous les utilisateurs et groupes ont été correctement ajoutés à la vue ID, utilisez la commande **ipa idview-show**.

```
$ ipa idview-show example-view
  ID View Name: example-view
  User object overrides: example-user1
  Group object overrides: example-group
```

CHAPITRE 35. UTILISATION DES VUES D'IDENTIFICATION POUR LES UTILISATEURS D'ACTIVE DIRECTORY

Vous pouvez utiliser les vues ID pour spécifier de nouvelles valeurs pour les attributs POSIX de vos utilisateurs Active Directory (AD) dans un environnement IdM-AD Trust.

Par défaut, IdM applique le site **Default Trust View** à tous les utilisateurs AD. Vous pouvez configurer des vues ID supplémentaires sur des clients IdM individuels pour ajuster davantage les attributs POSIX que des utilisateurs spécifiques reçoivent.

35.1. FONCTIONNEMENT DE LA VUE FIDUCIAIRE PAR DÉFAUT

L'adresse **Default Trust View** est la vue d'identification par défaut qui est toujours appliquée aux utilisateurs et aux groupes AD dans les configurations basées sur la confiance. Elle est créée automatiquement lorsque vous établissez la confiance à l'aide de la commande **ipa-adtrust-install** et ne peut pas être supprimée.

NOTE

La vue de confiance par défaut n'accepte les dérogations que pour les utilisateurs et les groupes AD, et non pour les utilisateurs et les groupes IdM.

La vue Trust par défaut permet de définir des attributs POSIX personnalisés pour les utilisateurs et les groupes AD, ce qui permet de remplacer les valeurs définies dans AD.

Tableau 35.1. Application de la vue de confiance par défaut

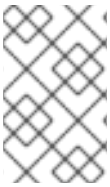
	Valeurs en DA	Vue fiduciaire par défaut	Résultat
Login	ad_user	ad_user	ad_user
UID	111	222	222
GID	111	(sans valeur)	111

Vous pouvez également configurer des vues d'identification supplémentaires pour remplacer la vue de confiance par défaut sur les clients IdM. IdM applique les valeurs de la vue d'identification spécifique à l'hôte en plus de la vue de confiance par défaut :

- Si un attribut est défini dans la vue ID spécifique à l'hôte, l'IdM applique la valeur de cette vue ID.
- Si un attribut n'est pas défini dans la vue d'identification spécifique à l'hôte, l'IdM applique la valeur de la vue de confiance par défaut.

Tableau 35.2. Application d'une vue d'identification spécifique à l'hôte au-dessus de la vue de confiance par défaut

	Valeurs en DA	Vue fiduciaire par défaut	Vue de l'ID spécifique à l'hôte	Résultat
Login	ad_user	ad_user	(sans valeur)	ad_user
UID	111	222	333	333
GID	111	(sans valeur)	333	333



NOTE

Vous ne pouvez appliquer des vues d'identification spécifiques à l'hôte que pour remplacer la vue de confiance par défaut sur les clients IdM. Les serveurs IdM et les répliques appliquent toujours les valeurs de la vue de confiance par défaut.

Ressources supplémentaires

- Utilisation d'une vue ID pour remplacer une valeur d'attribut utilisateur sur un client IdM

35.2. DÉFINITION D'ATTRIBUTS GLOBAUX POUR UN UTILISATEUR AD EN MODIFIANT LA VUE DE CONFIANCE PAR DÉFAUT

Si vous souhaitez remplacer un attribut POSIX pour un utilisateur Active Directory (AD) dans l'ensemble de votre déploiement IdM, modifiez l'entrée de cet utilisateur dans la vue de confiance par défaut. Cette procédure définit le GID de l'utilisateur AD **ad_user@ad.example.com** à 732000006.

Conditions préalables

- Vous vous êtes authentifié en tant qu'administrateur IdM.
- Un groupe doit exister avec le GID ou vous devez définir le GID dans un remplacement d'ID pour un groupe.

Procédure

1. En tant qu'administrateur IdM, créez un remplacement d'ID pour l'utilisateur AD dans la vue de confiance par défaut qui modifie le numéro GID en 732000006 :

```
# ipa idoverrideuser-add 'Default Trust View' ad_user@ad.example.com --
gidnumber=732000006
```

2. Effacez l'entrée de l'utilisateur **ad_user@ad.example.com** du cache SSSD sur tous les serveurs et clients IdM. Cela supprime les données périmées et permet d'appliquer la nouvelle valeur de remplacement.

```
# sssctl cache-expire -u ad_user@ad.example.com
```

Vérification

- Récupérer les informations relatives à l'utilisateur **ad_user@ad.example.com** pour vérifier que le GID reflète la valeur mise à jour.

```
# id ad_user@ad.example.com
uid=702801456(ad_user@ad.example.com) gid=732000006(ad_admins)
groups=732000006(ad_admins),702800513(domain users@ad.example.com)
```

35.3. REMPLACEMENT DES ATTRIBUTS DE LA VUE DE CONFIANCE PAR DÉFAUT POUR UN UTILISATEUR AD SUR UN CLIENT IDM AVEC UNE VUE ID

Il se peut que vous souhaitiez remplacer certains attributs POSIX de la vue de confiance par défaut pour un utilisateur Active Directory (AD). Par exemple, vous pouvez avoir besoin de donner à un utilisateur AD un GID différent sur un client IdM particulier. Vous pouvez utiliser une vue ID pour remplacer une valeur de la vue de confiance par défaut pour un utilisateur AD et l'appliquer à un seul hôte. Cette procédure explique comment définir le GID de l'utilisateur AD **ad_user@ad.example.com** sur le client IdM **host1.idm.example.com** à 732001337.

Conditions préalables

- Vous disposez d'un accès root au client **host1.idm.example.com** IdM.
- Vous êtes connecté en tant qu'utilisateur disposant des privilèges requis, par exemple l'utilisateur **admin**.

Procédure

1. Créez une vue d'identification. Par exemple, pour créer une vue ID nommée **example_for_host1**:

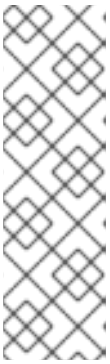
```
$ ipa idview-add example_for_host1
---------------------------
Added ID View "example_for_host1"
---------------------------
  ID View Name: example_for_host1
```

2. Ajoutez une dérogation pour l'utilisateur à la vue **example_for_host1** ID. Pour remplacer le GID de l'utilisateur :
   - Entrez la commande **ipa idoverrideuser-add**
   - Ajouter le nom de la vue ID
   - Ajouter le nom d'utilisateur, également appelé l'ancre
   - Ajouter l'option **--gidnumber=**:

```
$ ipa idoverrideuser-add example_for_host1 ad_user@ad.example.com --
gidnumber=732001337
-----------------------------
Added User ID override "ad_user@ad.example.com"
-----------------------------
  Anchor to override: ad_user@ad.example.com
  GID: 732001337
```

3. Appliquer **example_for_host1** au client **host1.idm.example.com** IdM :

```
$ ipa idview-apply example_for_host1 --hosts=host1.idm.example.com
-----------------------------
Applied ID View "example_for_host1"
-----------------------------
hosts: host1.idm.example.com
---------------------------------------------
Number of hosts the ID View was applied to: 1
---------------------------------------------
```



NOTE

La commande **ipa idview-apply** accepte également l'option **--hostgroups**. Cette option applique la vue ID aux hôtes qui appartiennent au groupe d'hôtes spécifié, mais n'associe pas la vue ID au groupe d'hôtes lui-même. Au lieu de cela, l'option **--hostgroups** développe les membres du groupe d'hôtes spécifié et applique l'option **--hosts** individuellement à chacun d'entre eux.

Cela signifie que si un hôte est ajouté au groupe d'hôtes dans le futur, la vue ID ne s'applique pas au nouvel hôte.

4. Effacez l'entrée de l'utilisateur **ad_user@ad.example.com** du cache SSSD sur le client IdM **host1.idm.example.com**. Cela supprime les données périmées et permet d'appliquer la nouvelle valeur d'annulation.

```
[root@host1 ~]# sssctl cache-expire -u ad_user@ad.example.com
```

Étapes de la vérification

1. **SSH** à **host1** comme **ad_user@ad.example.com**:

```
[root@r8server ~]# ssh ad_user@ad.example.com@host1.idm.example.com
```

2. Récupérer les informations relatives à l'utilisateur **ad_user@ad.example.com** pour vérifier que le GID reflète la valeur mise à jour.

```
[ad_user@ad.example.com@host1 ~]$ id ad_user@ad.example.com
uid=702801456(ad_user@ad.example.com) gid=732001337(admins2)
groups=732001337(admins2),702800513(domain users@ad.example.com)
```

35.4. APPLICATION D'UNE VUE ID À UN GROUPE D'HÔTES IDM

La commande **ipa idview-apply** accepte l'option **--hostgroups**. Cependant, l'option agit comme une opération unique qui applique la vue ID aux hôtes qui appartiennent actuellement au groupe d'hôtes spécifié, mais n'associe pas dynamiquement la vue ID au groupe d'hôtes lui-même. L'option **--hostgroups** étend les membres du groupe d'hôtes spécifié et applique l'option **--hosts** individuellement à chacun d'entre eux.

Si vous ajoutez ultérieurement un nouvel hôte au groupe d'hôtes, vous devez appliquer manuellement la vue ID au nouvel hôte, en utilisant la commande **ipa idview-apply** avec l'option **--hosts**.

De même, si vous supprimez un hôte d'un groupe d'hôtes, la vue ID est toujours affectée à l'hôte après la

suppression. Pour annuler l'affectation de la vue ID de l'hôte supprimé, vous devez exécuter la commande **ipa idview-unapply *id_view_name* --hosts=*name_of_the_removed_host*** pour annuler l'application de la vue ID de l'hôte supprimé.

Cette section décrit comment atteindre les objectifs suivants :

1. Comment créer un groupe d'hôtes et y ajouter des hôtes.
2. Comment appliquer une vue d'identification au groupe d'hôtes.
3. Comment ajouter un nouvel hôte au groupe d'hôtes et appliquer la vue ID au nouvel hôte.

Conditions préalables

- Assurez-vous que la vue ID que vous souhaitez appliquer au groupe d'hôtes existe dans IdM. Par exemple, pour créer une vue ID afin de remplacer le GID d'un utilisateur AD, voir Remplacer les attributs de la vue Trust par défaut pour un utilisateur AD sur un client IdM avec une vue ID

Procédure

1. Créez un groupe d'hôtes et ajoutez-y des hôtes :
   a. Créez un groupe d'hôtes. Par exemple, pour créer un groupe d'hôtes nommé **baltimore**:

```
[root@server ~]# ipa hostgroup-add --desc="Baltimore hosts" baltimore
---------------------------
Added hostgroup "baltimore"
---------------------------
Host-group: baltimore
Description: Baltimore hosts
```

   b. Ajoutez des hôtes au groupe d'hôtes. Par exemple, pour ajouter les hôtes **host102** et **host103** au groupe d'hôtes **baltimore**:

```
[root@server ~]# ipa hostgroup-add-member --hosts={host102,host103} baltimore
Host-group: baltimore
Description: Baltimore hosts
Member hosts: host102.idm.example.com, host103.idm.example.com
-------------------------
Number of members added 2
-------------------------
```

2. Appliquer une vue ID aux hôtes du groupe d'hôtes. Par exemple, pour appliquer la vue d'identification **example_for_host1** au groupe d'hôtes **baltimore**:

```
[root@server ~]# ipa idview-apply --hostgroups=baltimore
ID View Name: example_for_host1
-----------------------------------------
Applied ID View "example_for_host1"
-----------------------------------------
  hosts: host102.idm.example.com, host103.idm.example.com
-------------------------------------------
Number of hosts the ID View was applied to: 2
-------------------------------------------
```

3. Ajoutez un nouvel hôte au groupe d'hôtes et appliquez la vue ID au nouvel hôte :

   a. Ajoutez un nouvel hôte au groupe d'hôtes. Par exemple, pour ajouter l'hôte **somehost.idm.example.com** au groupe d'hôtes **baltimore**:

   ```
   [root@server ~]# ipa hostgroup-add-member --hosts=somehost.idm.example.com
   baltimore
     Host-group: baltimore
     Description: Baltimore hosts
     Member hosts:  host102.idm.example.com,
   host103.idm.example.com,somehost.idm.example.com
   -------------------------
   Number of members added 1
   -------------------------
   ```

   b. En option, affichez les informations sur la vue d'identification. Par exemple, pour afficher les détails de la vue d'identification **example_for_host1**:

   ```
   [root@server ~]# ipa idview-show example_for_host1 --all
     dn: cn=example_for_host1,cn=views,cn=accounts,dc=idm,dc=example,dc=com
     ID View Name: example_for_host1
   [...]
     Hosts the view applies to: host102.idm.example.com, host103.idm.example.com
     objectclass: ipaIDView, top, nsContainer
   ```

   La sortie montre que la vue ID n'est pas appliquée à **somehost.idm.example.com**, l'hôte nouvellement ajouté au groupe d'hôtes **baltimore**.

   c. Appliquez la vue ID au nouvel hôte. Par exemple, pour appliquer la vue d'identification **example_for_host1** à **somehost.idm.example.com**:

   ```
   [root@server ~]# ipa idview-apply --host=somehost.idm.example.com
   ID View Name: example_for_host1
   ----------------------------------------
   Applied ID View "example_for_host1"
   ----------------------------------------
     hosts: somehost.idm.example.com
   ----------------------------------------------
   Number of hosts the ID View was applied to: 1
   ----------------------------------------------
   ```

Verification steps

- Affichez à nouveau les informations de la vue ID :

```
[root@server ~]# ipa idview-show example_for_host1 --all
  dn: cn=example_for_host1,cn=views,cn=accounts,dc=idm,dc=example,dc=com
  ID View Name: example_for_host1
[...]
  Hosts the view applies to: host102.idm.example.com, host103.idm.example.com,
somehost.idm.example.com
  objectclass: ipaIDView, top, nsContainer
```

La sortie montre que la vue ID est maintenant appliquée à **somehost.idm.example.com**, l'hôte nouvellement ajouté au groupe d'hôtes **baltimore**.

CHAPITRE 36. AJUSTEMENT MANUEL DES PLAGES D'IDENTIFICATION

Un serveur IdM génère des numéros d'identification d'utilisateur (UID) et de groupe (GID) uniques. En créant et en attribuant différentes plages d'ID aux répliques, il s'assure également qu'elles ne génèrent jamais les mêmes numéros d'ID. Par défaut, ce processus est automatique. Toutefois, vous pouvez ajuster manuellement la plage d'ID IdM lors de l'installation du serveur IdM, ou définir manuellement la plage d'ID DNA d'un réplica.

36.1. PLAGES D'IDENTIFICATION

Les numéros d'identification sont divisés en *ID ranges*. Le fait de conserver des plages numériques distinctes pour les serveurs et les répliques individuels élimine le risque qu'un numéro d'identification attribué à une entrée soit déjà utilisé par une autre entrée sur un autre serveur ou une autre réplique.

Notez qu'il existe deux types distincts de plages d'identification :

- L'IdM ***ID range***qui est attribué lors de l'installation du premier serveur. Cette plage ne peut pas être modifiée après sa création. Toutefois, vous pouvez créer une nouvelle plage d'identifiants IdM en plus de la plage originale. Pour plus d'informations, voir Attribution automatique de plages d'identif iants et Ajout d'une nouvelle plage d'identifiants IdM .

- Les plages d'ID (ADN) de l ***Distributed Numeric Assignment*** (ADN), qui peuvent être modifiées par l'utilisateur. Elles doivent s'inscrire dans une plage d'identification IdM existante. Pour plus d'informations, voir Attribution manuelle de plages d'identification ADN.
  Les répliques peuvent également se voir attribuer une plage d'ID ADN **next**. Une réplique utilise sa plage suivante lorsqu'elle n'a plus d'ID dans sa plage actuelle. Les plages suivantes ne sont pas attribuées automatiquement lorsqu'un réplica est supprimé et vous devez les attribuer manuellement.

Les plages sont mises à jour et partagées entre le serveur et les répliques par le plug-in DNA, dans le cadre de l'instance de serveur d'annuaire de back-end pour le domaine.

La définition de la gamme d'ADN est déterminée par deux attributs :

- Le prochain numéro disponible du serveur : le bas de la fourchette de l'ADN

- La taille de la plage : le nombre d'ID dans la plage d'ADN

La valeur inférieure initiale est définie lors de la configuration de l'instance du plug-in. Ensuite, le plug-in met à jour la valeur inférieure. Le fait de diviser les numéros disponibles en plages permet aux serveurs d'attribuer continuellement des numéros sans qu'ils ne se chevauchent.

36.2. ATTRIBUTION AUTOMATIQUE DE PLAGES D'IDENTIFICATION

Plages d'IDM

Par défaut, une plage d'ID IdM est automatiquement attribuée lors de l'installation du serveur IdM. La commande **ipa-server-install** sélectionne et attribue de manière aléatoire une plage de 200 000 ID sur un total de 10 000 plages possibles. La sélection d'une plage aléatoire de cette manière réduit considérablement la probabilité de conflits d'ID au cas où vous décideriez de fusionner deux domaines IdM distincts à l'avenir.

NOTE

Cette plage d'identification IdM ne peut pas être modifiée après sa création. Vous ne pouvez ajuster que manuellement les plages d'ID de l'assignation numérique distribuée (ADN), à l'aide des commandes décrites dans Attribution manuelle des plages d'ID ADN. Une plage DNA correspondant à la plage IdM est automatiquement créée lors de l'installation.

Plages d'identification de l'ADN

Si un seul serveur IdM est installé, il contrôle l'ensemble de la plage d'ID ADN. Lorsque vous installez un nouveau réplica et que celui-ci demande sa propre plage d'ID ADN, la plage d'ID initiale du serveur se divise et est répartie entre le serveur et le réplica : le réplica reçoit la moitié de la plage d'ID ADN restante qui est disponible sur le serveur initial. Le serveur et la réplique utilisent ensuite leurs parties respectives de la plage d'identification initiale pour les nouvelles entrées d'utilisateurs ou de groupes. En outre, si la réplique est sur le point d'épuiser la plage d'identifiants qui lui a été attribuée et qu'il lui reste moins de 100 identifiants, elle contacte les autres serveurs disponibles pour demander une nouvelle plage d'identifiants ADN.

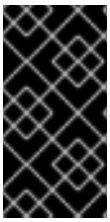
IMPORTANT

Lorsque vous installez un réplica, il **does not** reçoit immédiatement une plage d'identifiants. Un réplica reçoit une plage d'identifiants lors de la première utilisation du plug-in DNA, par exemple lorsque vous ajoutez un utilisateur pour la première fois.

Si le serveur initial cesse de fonctionner avant que le réplica ne lui demande une plage d'ID d'ADN, le réplica ne peut pas contacter le serveur pour demander la plage d'ID. La tentative d'ajout d'un nouvel utilisateur sur le réplica échoue alors. Dans ce cas, vous pouvez déterminer la plage d'identifiants attribuée au serveur désactivé et attribuer manuellement une plage d'identifiants au réplica .

36.3. ATTRIBUTION MANUELLE DE LA PLAGE D'ID IDM LORS DE L'INSTALLATION DU SERVEUR

Vous pouvez ignorer le comportement par défaut et définir manuellement une plage d'IDM au lieu de l'attribuer de manière aléatoire.



IMPORTANT

Ne définissez pas de plages d'identifiants comprenant des valeurs UID de 1000 et inférieures ; ces valeurs sont réservées à l'usage du système. Ne définissez pas non plus une plage d'identifiants qui inclurait la valeur 0 ; le service SSSD ne gère pas la valeur d'identifiant 0.

Procédure

- Vous pouvez définir manuellement la plage d'ID IdM pendant l'installation du serveur en utilisant les deux options suivantes avec **ipa-server-install**:
  - **--idstart** donne la valeur de départ pour les numéros UID et GID.
  - **--idmax** donne le nombre maximum d'UID et de GID ; par défaut, la valeur est la valeur de départ **--idstart** plus 199 999.

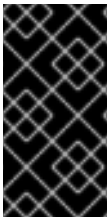
Verification steps

- Pour vérifier si la plage d'identifiants a été correctement attribuée, vous pouvez afficher la plage d'identifiants IdM attribuée à l'aide de la commande **ipa idrange-find**:

```
# ipa idrange-find
----------------
1 range matched
----------------
  Range name: IDM.EXAMPLE.COM_id_range
  First Posix ID of the range: 882200000
  Number of IDs in the range: 200000
  Range type: local domain range
----------------------------
Number of entries returned 1
----------------------------
```

36.4. AJOUT D'UNE NOUVELLE PLAGE D'IDM

Dans certains cas, vous pouvez vouloir créer une nouvelle plage d'ID IdM en plus de la plage d'origine ; par exemple, lorsqu'un réplica n'a plus d'ID et que la plage d'ID IdM d'origine est épuisée.



IMPORTANT

L'ajout d'une nouvelle plage d'ID IdM ne crée pas automatiquement de nouvelles plages d'ID ADN. Vous devez attribuer manuellement de nouvelles plages d'ID ADN aux réplicas, le cas échéant. Pour plus d'informations sur la procédure à suivre, voir Attribution manuelle de plages d'ID ADN.

Procédure

1. Pour créer une nouvelle plage d'ID IdM, utilisez la commande **ipa idrange-add**. Vous devez spécifier le nom de la nouvelle plage, le premier numéro d'identification de la plage et la taille de la plage :

```
# ipa idrange-add IDM.EXAMPLE.COM_new_range --base-id=1000000 --range-size=200000
----------------------------------------
Added ID range "IDM.EXAMPLE.COM_new_range"
----------------------------------------
  Range name: IDM.EXAMPLE.COM_new_range
  First Posix ID of the range: 1000000
  Number of IDs in the range: 200000
  Range type: local domain range
```

2. Facultatif : Mettre à jour la plage d'identifiants immédiatement :

   a. Effacer le cache du System Security Services Daemon (SSSD) :

```
# sss_cache -E
```

   b. Redémarrez le démon SSSD :

```
# systemctl restart sssd
```

NOTE

Si vous n'effacez pas le cache SSSD et ne redémarrez pas le service, SSSD ne détecte la nouvelle plage d'identifiants que lorsqu'il met à jour la liste des domaines et d'autres données de configuration stockées sur le serveur IdM.

Verification steps

- Vous pouvez vérifier si la nouvelle plage est correctement définie en utilisant la commande **ipa idrange-find**:

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

  Range name: IDM.EXAMPLE.COM_new_range
  First Posix ID of the range: 1000000
  Number of IDs in the range: 200000
  Range type: local domain range
----------------------------
Number of entries returned 2
----------------------------
```

36.5. LE RÔLE DE LA SÉCURITÉ ET DES IDENTIFIANTS RELATIFS DANS LES GAMMES D'IDENTIFIANTS IDM

Une plage d'identifiants de gestion d'identité (IdM) est définie par plusieurs paramètres :

- Le nom de la plage
- Le premier ID POSIX de la plage
- La taille de la plage : le nombre d'ID dans la plage
- Le premier **relative identifier** (RID) de l'article correspondant **RID range**
- Le premier RID du **secondary RID range**

Vous pouvez visualiser ces valeurs à l'aide de la commande **ipa idrange-show**:

```
$ ipa idrange-show IDM.EXAMPLE.COM_id_range
  Range name: IDM.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: 1000000
  Range type: local domain range
```

Identifiants de sécurité

Les données des plages d'ID du domaine local sont utilisées par le serveur IdM en interne pour attribuer des **security identifiers** uniques (SID) aux utilisateurs et aux groupes IdM. Les SID sont stockés dans les objets utilisateur et groupe. Le SID d'un utilisateur se compose des éléments suivants :

- Le SID du domaine
- Le **relative identifier** (RID) de l'utilisateur, qui est une valeur à quatre chiffres de 32 bits ajoutée au SID du domaine

Par exemple, si le SID du domaine est S-1-5-21-123-456-789 et que le RID d'un utilisateur de ce domaine est 1008, l'utilisateur a le SID S-1-5-21-123-456-789-1008.

Identifiants relatifs

Le RID lui-même est calculé de la manière suivante :

Soustrayez le premier ID POSIX de la plage de l'UID POSIX de l'utilisateur et ajoutez au résultat le premier RID de la plage RID correspondante. Par exemple, si l'UID de *idmuser* est 196600008, le premier POSIX ID est 196600000, et le premier RID est 1000, alors le RID de *idmuser* est 1008.

NOTE

L'algorithme qui calcule le RID de l'utilisateur vérifie si un ID POSIX donné entre dans la plage d'ID attribuée avant de calculer un RID correspondant. Par exemple, si le premier ID est 196600000 et que la taille de la plage est 200000, l'ID POSIX 1600000 est en dehors de la plage d'ID et l'algorithme ne calcule pas de RID pour lui.

Identifiants relatifs secondaires

Dans IdM, un UID POSIX peut être identique à un GID POSIX. Cela signifie que si *idmuser* existe déjà avec l'UID 196600008, vous pouvez toujours créer un nouveau groupe *idmgroup* avec le GID 196600008.

Cependant, un SID ne peut définir qu'un seul objet, un utilisateur *or* un groupe. Le SID de S-1-5-21-123-456-789-1008 qui a déjà été créé pour *idmuser* ne peut pas être partagé avec *idmgroup*. Un autre SID doit être généré pour *idmgroup*.

L'IdM utilise un **secondary relative identifier**, ou RID secondaire, pour éviter les conflits de SID. Ce RID secondaire se compose des éléments suivants :

- La base secondaire du RID
- Taille de la plage ; par défaut identique à la taille de la plage de base

Dans l'exemple ci-dessus, la base du RID secondaire est fixée à 1000000. Pour calculer le RID de *idmgroup* nouvellement créé : soustrayez le premier ID POSIX de la plage de l'UID POSIX de l'utilisateur, et ajoutez le premier RID de la plage de RID secondaire au résultat. *idmgroup* se voit donc attribuer le RID de 1000008. Par conséquent, le SID de *idmgroup* est S-1-5-21-123-456-789-1000008.

IdM utilise le RID secondaire pour calculer un SID uniquement si un utilisateur ou un objet de groupe a été précédemment créé avec un ID POSIX défini manuellement. Dans le cas contraire, l'attribution automatique empêche d'attribuer deux fois le même ID.

Ressources supplémentaires

- Utilisation d'Ansible pour ajouter une nouvelle plage d'identifiants IdM locaux

36.6. UTILISATION D'ANSIBLE POUR AJOUTER UNE NOUVELLE PLAGE D'IDENTIFIANTS IDM LOCAUX

Dans certains cas, vous pouvez vouloir créer une nouvelle plage d'ID de gestion d'identité (IdM) en plus de la plage d'origine ; par exemple, lorsqu'un réplica n'a plus d'ID et que la plage d'ID IdM d'origine est épuisée. L'exemple suivant décrit comment créer une nouvelle plage d'ID IdM à l'aide d'une séquence Ansible.

NOTE

L'ajout d'une nouvelle plage d'identification IdM ne crée pas automatiquement de nouvelles plages d'identification ADN. Vous devez attribuer manuellement de nouvelles plages d'ID ADN si nécessaire. Pour plus d'informations sur la manière de procéder, voir Affectation manuelle de plages d'ID ADN .

Conditions préalables

- Vous connaissez le mot de passe de l'IdM **admin**.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

Procédure

1. Naviguez jusqu'à votre répertoire ***~/MyPlaybooks/*** répertoire :

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

2. Créez le playbook **idrange-present.yml** avec le contenu suivant :

```
---
- name: Playbook to manage idrange
  hosts: ipaserver
  become: no

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Ensure local idrange is present
    ipaidrange:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: new_id_range
      base_id: 12000000
```

```
    range_size: 200000
    rid_base: 1000000
    secondary_rid_base: 200000000
```

3. Enregistrer le fichier.
4. Exécutez le playbook Ansible. Spécifiez le fichier du livre de jeu, le fichier contenant le mot de passe protégeant le fichier **secret.yml** et le fichier d'inventaire :

```
$ ansible-playbook --vault-password-file=password_file -v -i inventory idrange-present.yml
```

5. Facultatif : Mettre à jour la plage d'identifiants immédiatement :
   a. Effacer le cache du System Security Services Daemon (SSSD) :

```
# sss_cache -E
```

   b. Redémarrez le démon SSSD :

```
# systemctl restart sssd
```

NOTE

Si vous n'effacez pas le cache SSSD et ne redémarrez pas le service, SSSD ne détecte la nouvelle plage d'identifiants que lorsqu'il met à jour la liste des domaines et d'autres données de configuration stockées sur le serveur IdM.

Verification steps

- Vous pouvez vérifier si la nouvelle plage est correctement définie en utilisant la commande **ipa idrange-find**:

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

  Range name: IDM.EXAMPLE.COM_new_id_range
  First Posix ID of the range: 12000000
  Number of IDs in the range: 200000
  Range type: local domain range
----------------------------
Number of entries returned 2
----------------------------
```

Ressources supplémentaires

- Le rôle de la sécurité et des identifiants relatifs dans les gammes d'identifiants IdM

36.7. SUPPRESSION D'UNE PLAGE D'IDENTIFIANTS APRÈS LA SUPPRESSION D'UNE CONFIANCE DANS AD

Si vous avez supprimé la confiance entre vos environnements IdM et Active Directory (AD), il se peut que vous souhaitiez supprimer la plage d'identifiants qui y est associée.

AVERTISSEMENT

Les identifiants attribués aux plages d'identifiants associées aux domaines de confiance peuvent encore être utilisés pour la propriété des fichiers et des répertoires sur les systèmes inscrits dans IdM.

Si vous supprimez la plage d'identifiants correspondant à un groupe AD que vous avez supprimé, vous ne pourrez pas déterminer la propriété des fichiers et des répertoires appartenant à des utilisateurs AD.

Conditions préalables

- Vous avez supprimé une confiance dans un environnement AD.

Procédure

1. Affiche toutes les plages d'identification actuellement utilisées :

```
[root@server ~]# ipa idrange-find
```

2. Identifiez le nom de la plage d'identifiants associée au trust que vous avez supprimé. La première partie du nom de la plage d'identifiants est le nom du trust, par exemple **AD.EXAMPLE.COM_id_range**.

3. Retirer la gamme :

```
[root@server ~]# ipa idrange-del AD.EXAMPLE.COM_id_range
```

4. Redémarrez le service SSSD pour supprimer les références à la plage d'identifiants que vous avez supprimée.

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

Ressources supplémentaires

- Voir Suppression de la confiance à l'aide de la ligne de commande .
- Voir Suppression de la confiance à l'aide de l'interface Web IdM .

36.8. AFFICHAGE DES PLAGES D'IDENTIFICATION D'ADN ACTUELLEMENT ATTRIBUÉES

Vous pouvez afficher la plage d'ID d'assignation numérique distribuée (DNA) actuellement active sur un serveur, ainsi que sa prochaine plage DNA s'il en a une d'attribuée.

Procédure

- Pour afficher les plages d'ID DNA configurées pour les serveurs de la topologie, utilisez les commandes suivantes :
  - **ipa-replica-manage dnarange-show** affiche la plage d'ID ADN actuelle qui est définie sur tous les serveurs ou, si vous spécifiez un serveur, uniquement sur le serveur spécifié, par exemple :

```
# ipa-replica-manage dnarange-show
serverA.example.com: 1001-1500
serverB.example.com: 1501-2000
serverC.example.com: No range set

# ipa-replica-manage dnarange-show serverA.example.com
serverA.example.com: 1001-1500
```

  - **ipa-replica-manage dnanextrange-show** affiche la plage suivante d'ID ADN actuellement définie sur tous les serveurs ou, si vous spécifiez un serveur, uniquement sur le serveur spécifié, par exemple :

```
# ipa-replica-manage dnanextrange-show
serverA.example.com: 2001-2500
serverB.example.com: No on-deck range set
serverC.example.com: No on-deck range set

# ipa-replica-manage dnanextrange-show serverA.example.com
serverA.example.com: 2001-2500
```

36.9. ATTRIBUTION MANUELLE D'UNE PLAGE D'IDENTIFICATION

Dans certaines situations, il est nécessaire d'attribuer manuellement une plage d'ID d'assignation numérique distribuée (ADN), par exemple lorsque.. :

- Une réplique n'a plus d'identifiants et la gamme d'identifiants IdM est épuisée
  Un réplica a épuisé la plage d'ID DNA qui lui a été attribuée et la demande d'ID supplémentaires a échoué parce qu'il n'y a plus d'ID libres dans la plage IdM.

  Pour résoudre cette situation, étendez la plage d'ID ADN attribuée à la réplique. Vous pouvez le faire de deux manières :

  - Raccourcir la plage d'ID ADN attribuée à une autre réplique, puis attribuer les nouvelles valeurs disponibles à la réplique épuisée.

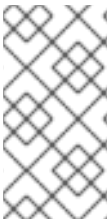
  - Créez une nouvelle plage d'IDM, puis définissez une nouvelle plage d'ID ADN pour le réplica dans cette plage d'IDM créée.
    Pour plus d'informations sur la création d'une nouvelle plage d'ID IdM, voir Ajout d'une nouvelle plage d'ID IdM.

- Une réplique a cessé de fonctionner
  La plage d'ID ADN d'un réplica n'est pas automatiquement récupérée lorsque le réplica cesse de fonctionner et doit être supprimé, ce qui signifie que la plage d'ID ADN précédemment

attribuée au réplica devient indisponible. Vous souhaitez récupérer la plage d'ID ADN et la rendre disponible pour d'autres réplicas.

Pour ce faire, déterminez les valeurs de la plage d'identifiants avant d'affecter manuellement cette plage à un autre serveur. En outre, pour éviter les doublons d'UID ou de GID, assurez-vous qu'aucune valeur d'ID de la plage récupérée n'a été précédemment attribuée à un utilisateur ou à un groupe ; vous pouvez le faire en examinant les UID et les GID des utilisateurs et des groupes existants.

Vous pouvez attribuer manuellement une plage d'ID ADN à une réplique à l'aide des commandes indiquées dans la section Attribution manuelle de plages d'ID ADN .



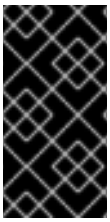
NOTE

Si vous attribuez une nouvelle plage d'ID ADN, les UID des entrées déjà existantes sur le serveur ou la réplique restent les mêmes. Cela ne pose pas de problème car même si vous changez la plage d'ID ADN actuelle, l'IdM conserve un enregistrement des plages qui ont été attribuées dans le passé.

36.10. ATTRIBUTION MANUELLE DE PLAGES D'IDENTIFICATION D'ADN

Dans certains cas, il peut être nécessaire d'attribuer manuellement des plages d'identifiants DNA (Distributed Numeric Assignment) à des répliques existantes, par exemple pour réaffecter une plage d'identifiants DNA attribuée à une réplique qui ne fonctionne pas. Pour plus d'informations, voir Attribution manuelle de plages d'identifiants.

Lorsque vous ajustez manuellement une plage d'identification d'ADN, assurez-vous que la nouvelle plage ajustée est incluse dans la plage d'identification de l'IdM ; vous pouvez le vérifier à l'aide de la commande **ipa idrange-find**. Dans le cas contraire, la commande échoue.



IMPORTANT

Veillez à ne pas créer de plages d'identifiants qui se chevauchent. Si l'une des plages d'identifiants que vous attribuez aux serveurs ou aux répliques se chevauche, deux serveurs différents risquent d'attribuer la même valeur d'identifiant à des entrées différentes.

Conditions préalables

- *Optional.* Si vous récupérez une plage d'identification d'ADN à partir d'une réplique qui ne fonctionne pas, recherchez d'abord la plage d'identification à l'aide des commandes décrites dans la section Affichage des plages d'identification d'ADN actuellement attribuées .

Procédure

- Pour définir la plage d'ID ADN en cours pour un serveur donné, utilisez **ipa-replica-manage dnarange-set**:

```
# ipa-replica-manage dnarange-set serverA.example.com 1250-1499
```

- Pour définir la plage d'ID ADN suivante pour un serveur donné, utilisez **ipa-replica-manage dnanextrange-set**:

```
# ipa-replica-manage dnanextrange-set serverB.example.com 1500-5000
```

Verification steps

- Vous pouvez vérifier que les nouvelles plages d'ADN sont correctement définies en utilisant les commandes décrites dans la section Affichage des plages d'ID ADN actuellement attribuées .

CHAPITRE 37. GESTION MANUELLE DES PLAGES DE SOUS-IDENTIFIANTS

Dans un environnement conteneurisé, il arrive qu'un utilisateur IdM doive attribuer manuellement des plages de sous-identifiants. Les instructions suivantes vous aideront à gérer les plages de sous-identifiants.

37.1. GÉNÉRER DES PLAGES DE SOUS-IDENTIFIANTS À L'AIDE DE L'INTERFACE CLI DE L'IDM

Vous pouvez générer une plage de sous-identifiants et l'attribuer manuellement à un utilisateur. Supposons que le nom d'utilisateur *jsmith* existe sur un serveur **ipa**.

Conditions préalables

- L'utilisateur IdM existe.
- Un ticket Kerberos valide est obtenu. Voir Connexion à IdM dans l'interface Web : Utilisation d'un ticket Kerberos pour plus de détails.
- **root** privilèges.

Procédure

1. Vérifier les plages de sous-identification existantes :
   # **ipa subid-find**
2. Si la plage de sous-identifiants n'existe pas, générez et attribuez la nouvelle plage de sous-identifiants à un utilisateur en entrant la commande suivante :

```
# ipa subid-generate --owner=jsmith

Added subordinate id "359dfcef-6b76-4911-bd37-bb5b66b8c418"

  Unique ID: 359dfcef-6b76-4911-bd37-bb5b66b8c418
  Description: auto-assigned subid
  Owner: jsmith
  SubUID range start: 2147483648
  SubUID range size: 65536
  SubGID range start: 2147483648
  SubGID range size: 65536
```

3. Il est également possible de générer et d'attribuer les nouvelles plages de sous-identifiants à tous les utilisateurs :

```
# /usr/libexec/ipa/ipa-subids --all-users

Found 2 user(s) without subordinate ids
  Processing user 'user4' (1/2)
  Processing user 'user5' (2/2)
Updated 2 user(s)
The ipa-subids command was successful
```

Notez que pour attribuer par défaut des plages de sous-ID aux nouveaux utilisateurs IdM, activez l'option suivante :

```
# ipa config-mod --user-default-subid=True
```

Vérification

1. Pour vérifier si l'utilisateur dispose de la plage de sous-identifiants, entrez la commande suivante :

```
# ipa subid-find --owner=jsmith

1 subordinate id matched

  Unique ID: 359dfcef-6b76-4911-bd37-bb5b66b8c418
  Owner: jsmith
  SubUID range start: 2147483648
  SubUID range size: 65536
  SubGID range start: 2147483648
  SubGID range size: 65536

Number of entries returned 1
```

37.2. GÉNÉRER DES PLAGES DE SOUS-IDENTIFIANTS À L'AIDE DE L'INTERFACE WEBUI DE L'IDM

Vous pouvez générer une plage de sous-identifiants et l'attribuer à un utilisateur dans l'interface WebUI de l'IdM.

Conditions préalables

- Un utilisateur IdM existe.
- Un ticket Kerberos valide est obtenu. Voir Connexion à IdM dans l'interface Web : Utilisation d'un ticket Kerberos pour plus de détails.
- **root** privilèges.

Procédure

1. Dans l'interface IdM WebUI, développez l'onglet **Subordinate IDs** et choisissez l'option **Subordinate IDs**.
2. Lorsque l'interface **Subordinate IDs** apparaît, cliquez sur le bouton **Ajouter** dans le coin supérieur droit de l'interface. La fenêtre **"Add subid"** s'affiche.
3. Dans la fenêtre **"Add subid"**, choisissez un propriétaire, c'est-à-dire l'utilisateur auquel vous souhaitez attribuer une plage de sous-identifiants.
4. Cliquez sur le bouton **Ajouter**.

Vérification

1. Vérifiez le tableau sous l'onglet **Subordinate IDs**. Un nouvel enregistrement devrait apparaître et le propriétaire est l'utilisateur auquel vous avez attribué la plage de sous-identifiants.

37.3. GESTION DES PLAGES DE SOUS-IDENTIFIANTS EXISTANTES À L'AIDE DE LA CLI DE L'IDM

Vous pouvez rechercher des plages de sous-identifiants et afficher des informations sur un sous-identifiant particulier si nécessaire. Supposons que le nom d'utilisateur *jsmith* existe sur un serveur **ipa**.

Conditions préalables

- Un utilisateur IdM existe.

Procédure

1. Pour afficher les détails de la plage de sous-ID lorsque vous connaissez un hachage d'ID unique, entrez la commande suivante :

```
# ipa subid-show 359dfcef-6b76-4911-bd37-bb5b66b8c418

  Unique ID: 359dfcef-6b76-4911-bd37-bb5b66b8c418
  Owner: jsmith
  SubUID range start: 2147483648
  SubUID range size: 65536
  SubGID range start: 2147483648
  SubGID range size: 65536
```

2. Pour trouver les détails de la plage de sous-ID lorsque vous avez un sous-ID de cette plage, vous pouvez utiliser la commande suivante :

```
# ipa subid-match --subuid=2147483648

1 subordinate id matched

  Unique ID: 359dfcef-6b76-4911-bd37-bb5b66b8c418
  Owner: uid=jsmith
  SubUID range start: 2147483648
  SubUID range size: 65536
  SubGID range start: 2147483648
  SubGID range size: 65536

Number of entries returned 1
```

37.4. LISTE DES PLAGES DE SOUS-ID À L'AIDE DE LA COMMANDE GETSUBID

Pour dresser la liste des plages de sous-ID, par exemple pour le site **user1** dans un environnement IdM, suivez les instructions ci-dessous.

Conditions préalables

- Le site **user1** existe dans l'IdM.

- Le paquet **shadow-utils-subid** est installé.

Procédure

1. Inclure l'enregistrement **subid: sss** dans le fichier **/etc/nsswitch.conf**.
   Notez que vous ne pouvez fournir qu'une seule valeur pour le champ **subid**. La valeur **sss** attribuée au champ **subid** indique aux utilitaires qu'ils doivent utiliser les plages de sous-identifiants des paramètres IdM. La valeur **file** ou l'absence de valeur indique aux utilitaires d'utiliser les plages de sous-identifiants des fichiers **/etc/subuid** et **/etc/subgid**.

2. Liste la plage de sous-identifiants d'un utilisateur :

```
# getsubids user1
0: user1 2147483648 65536
```

CHAPITRE 38. GESTION DES HÔTES DANS L'INTERFACE DE GESTION DE L'IDM

Ce chapitre présente les hôtes et les entrées d'hôtes dans la gestion des identités (IdM), ainsi que les opérations suivantes effectuées lors de la gestion des hôtes et des entrées d'hôtes dans le CLI IdM :

- Inscription au programme d'accueil
- Ajout d'entrées d'hôtes IdM
- Suppression des entrées de l'hôte IdM
- Réinscription des hôtes
- Renommer les hôtes
- Désactivation des hôtes
- Réactivation des hôtes

Le chapitre contient également un tableau récapitulatif des conditions préalables, du contexte et des conséquences de ces opérations.

38.1. HÔTES DANS L'IDM

La gestion des identités (IdM) gère ces identités :

- Utilisateurs
- Services
- Hosts

Un hôte représente une machine. En tant qu'identité IdM, un hôte possède une entrée dans le LDAP IdM, c'est-à-dire l'instance 389 Directory Server du serveur IdM.

L'entrée de l'hôte dans IdM LDAP est utilisée pour établir des relations entre d'autres hôtes et même des services au sein du domaine. Ces relations font partie de l'autorisation et du contrôle des hôtes au sein du domaine (*delegating*). Tout hôte peut être utilisé dans les règles **host-based access control** (HBAC).

Le domaine IdM établit une communauté entre les machines, avec des informations d'identité communes, des politiques communes et des services partagés. Toute machine appartenant à un domaine fonctionne comme un client du domaine, ce qui signifie qu'elle utilise les services fournis par le domaine. Le domaine IdM fournit trois services principaux spécifiquement destinés aux machines :

- DNS
- Kerberos
- Gestion des certificats

Dans l'IdM, les hôtes sont étroitement liés aux services qui y sont exécutés :

- Les entrées de service sont associées à un hôte.

- Un hôte stocke les principaux Kerberos de l'hôte et du service.

38.2. INSCRIPTION AU PROGRAMME D'ACCUEIL

Cette section décrit l'enrôlement des hôtes en tant que clients IdM et ce qui se passe pendant et après l'enrôlement. Elle compare l'enrôlement des hôtes IdM et des utilisateurs IdM. Elle décrit également les autres types d'authentification disponibles pour les hôtes.

L'inscription d'un hôte consiste à

- Création d'une entrée d'hôte dans IdM LDAP : éventuellement en utilisant la commande**ipa host-add** dans IdM CLI, ou l'opération équivalente dans IdM Web UI.
- Configuration des services IdM sur l'hôte, par exemple le System Security Services Daemon (SSSD), Kerberos et certmonger, et connexion de l'hôte au domaine IdM.

Les deux actions peuvent être effectuées séparément ou ensemble.

S'ils sont exécutés séparément, ils permettent de répartir les deux tâches entre deux utilisateurs ayant des niveaux de privilèges différents. Cela est utile pour les déploiements en masse.

La commande **ipa-client-install** peut effectuer les deux actions ensemble. La commande crée une entrée d'hôte dans IdM LDAP si cette entrée n'existe pas encore, et configure les services Kerberos et SSSD pour l'hôte. La commande amène l'hôte dans le domaine IdM et lui permet d'identifier le serveur IdM auquel il se connectera. Si l'hôte appartient à une zone DNS gérée par IdM, **ipa-client-install** ajoute également des enregistrements DNS pour l'hôte. La commande doit être exécutée sur le client.

38.3. PRIVILÈGES DE L'UTILISATEUR REQUIS POUR L'INSCRIPTION DE L'HÔTE

L'opération d'enrôlement des hôtes nécessite une authentification afin d'éviter qu'un utilisateur non privilégié n'ajoute des machines indésirables au domaine IdM. Les privilèges requis dépendent de plusieurs facteurs, par exemple :

- Si une entrée d'hôte est créée séparément de l'exécution de **ipa-client-install**
- Si un mot de passe à usage unique (OTP) est utilisé pour l'inscription

Privilèges de l'utilisateur pour la création manuelle facultative d'une entrée d'hôte dans IdM LDAP

Le privilège d'utilisateur requis pour créer une entrée d'hôte dans IdM LDAP à l'aide de la commande CLI **ipa host-add** ou de l'interface Web IdM est **Host Administrators**. Le privilège **Host Administrators** peut être obtenu par le biais du rôle **IT Specialist**.

Privilèges de l'utilisateur pour l'intégration du client dans le domaine IdM

Les hôtes sont configurés en tant que clients IdM lors de l'exécution de la commande **ipa-client-install**. Le niveau d'habilitation requis pour l'exécution de la commande **ipa-client-install** dépend du scénario d'inscription dans lequel vous vous trouvez :

- L'entrée de l'hôte dans IdM LDAP n'existe pas. Pour ce scénario, vous avez besoin des informations d'identification d'un administrateur complet ou du rôle **Host Administrators**. Un administrateur complet est membre du groupe **admins**. Le rôle **Host Administrators** permet

d'ajouter des hôtes et d'enrôler des hôtes. Pour plus d'informations sur ce scénario, voir Installation d'un client à l'aide des informations d'identification de l'utilisateur : installation interactive.

- L'entrée de l'hôte dans IdM LDAP existe. Pour ce scénario, vous avez besoin des informations d'identification d'un administrateur limité pour exécuter **ipa-client-install** avec succès. Dans ce cas, l'administrateur limité a le rôle **Enrollment Administrator**, qui lui confère le privilège **Host Enrollment**. Pour plus d'informations, voir Installation d'un client à l'aide des informations d'identification de l'utilisateur : installation interactive.

- L'entrée de l'hôte dans IdM LDAP existe et un OTP a été généré pour l'hôte par un administrateur complet ou limité. Dans ce cas, vous pouvez installer un client IdM en tant qu'utilisateur ordinaire si vous exécutez la commande **ipa-client-install** avec l'option **--password**, en fournissant l'OTP correct. Pour plus de détails, voir Installation d'un client à l'aide d'un mot de passe à usage unique : installation interactive.

Après l'inscription, les hôtes IdM authentifient chaque nouvelle session pour pouvoir accéder aux ressources IdM. L'authentification de la machine est nécessaire pour que le serveur IdM fasse confiance à la machine et accepte les connexions IdM du logiciel client installé sur cette machine. Après avoir authentifié le client, le serveur IdM peut répondre à ses demandes.

38.4. COMPARAISON ENTRE L'ENRÔLEMENT ET L'AUTHENTIFICATION DES HÔTES ET DES UTILISATEURS DE L'IDM

Il existe de nombreuses similitudes entre les utilisateurs et les hôtes dans l'IdM. Cette section décrit certaines des similitudes qui peuvent être observées au cours de la phase d'inscription ainsi que celles qui concernent l'authentification au cours de la phase de déploiement.

- La phase d'inscription(inscription de l'utilisateur et de l'hôte) :

  - Un administrateur peut créer une entrée LDAP pour un utilisateur et un hôte avant que l'utilisateur ou l'hôte ne rejoigne l'IdM : pour l'utilisateur de l'étape, la commande est **ipa stageuser-add**; pour l'hôte, la commande est **ipa host-add**.

  - Un fichier contenant un *key table* ou, en abrégé, un keytab, une clé symétrique ressemblant dans une certaine mesure à un mot de passe d'utilisateur, est créé lors de l'exécution de la commande **ipa-client-install** sur l'hôte, ce qui permet à l'hôte de rejoindre le domaine IdM. De manière analogue, un utilisateur est invité à créer un mot de passe lorsqu'il active son compte, rejoignant ainsi le royaume IdM.

  - Alors que le mot de passe de l'utilisateur est la méthode d'authentification par défaut pour un utilisateur, le keytab est la méthode d'authentification par défaut pour un hôte. Le keytab est stocké dans un fichier sur l'hôte.

Tableau 38.1. Inscription des utilisateurs et des hôtes

Action	User	Hôte
Préinscription	$ **ipa stageuser-add** ***user_name*** **[--password]**	$ **ipa host-add** ***host_name*** **[--random]**
Activation du compte	$ **ipa stageuser-activate** ***user_name***	**ipa-client install [--password]** (doit être exécuté sur l'hôte lui-même)

- La phase de déploiement(authentification de la session de l'utilisateur et de l'hôte) :
  - Lorsqu'un utilisateur démarre une nouvelle session, il s'authentifie à l'aide d'un mot de passe ; de même, à chaque fois qu'il est mis sous tension, l'hôte s'authentifie en présentant son fichier keytab. Le System Security Services Daemon (SSSD) gère ce processus en arrière-plan.
  - Si l'authentification est réussie, l'utilisateur ou l'hôte obtient un ticket Kerberos (TGT).
  - Le TGT est ensuite utilisé pour obtenir des billets spécifiques pour des services spécifiques.

Tableau 38.2. Authentification de la session de l'utilisateur et de l'hôte

	User	Hôte
Moyens d'authentification par défaut	**Password**	**Keytabs**
Démarrage d'une session (utilisateur ordinaire)	$ **kinit** ***user_name***	*[switch on the host]*
Le résultat de l'authentification réussie	**TGT** à utiliser pour obtenir l'accès à des services spécifiques	**TGT** à utiliser pour obtenir l'accès à des services spécifiques

Les TGT et autres tickets Kerberos sont générés dans le cadre des services et politiques Kerberos définis par le serveur. L'octroi initial d'un ticket Kerberos, le renouvellement des informations d'identification Kerberos et même la destruction de la session Kerberos sont tous gérés automatiquement par les services IdM.

Autres options d'authentification pour les hôtes IdM

Outre les keytabs, IdM prend en charge deux autres types d'authentification des machines :

- Clés SSH. La clé publique SSH de l'hôte est créée et téléchargée dans l'entrée de l'hôte. À partir de là, le System Security Services Daemon (SSSD) utilise IdM comme fournisseur d'identité et peut travailler en conjonction avec OpenSSH et d'autres services pour référencer les clés publiques situées de manière centralisée dans IdM.
- Certificats de machine. Dans ce cas, la machine utilise un certificat SSL délivré par l'autorité de certification du serveur IdM et stocké dans le serveur d'annuaire IdM. Le certificat est ensuite envoyé à la machine pour qu'elle le présente lorsqu'elle s'authentifie auprès du serveur. Sur le client, les certificats sont gérés par un service appelé certmonger.

38.5. OPÉRATIONS D'ACCUEIL

Cette section énumère les opérations les plus courantes liées à l'enrôlement et à l'habilitation des hôtes, et explique les conditions préalables, le contexte et les conséquences de leur exécution.

Tableau 38.3. Opérations d'accueil partie 1

Action	Quelles sont les conditions préalables à l'action ?	Quand est-il judicieux d'exécuter la commande ?	Comment l'action est-elle effectuée par un administrateur système ? Quelle(s) commande(s) exécute-t-il ?
Enrolling a client	voir Préparation du système pour l'installation du client Identity Management dans la section *Installing_Identity_Management*	Lorsque vous souhaitez que l'hôte rejoigne la zone IdM.	L'inscription de machines en tant que clients dans le domaine IdM est un processus en deux parties. Une entrée d'hôte est créée pour le client (et stockée dans l'instance du serveur d'annuaire) lorsque la commande **ipa host-add** est exécutée, puis un fichier keytab est créé pour approvisionner le client. Ces deux étapes sont exécutées automatiquement par la commande **ipa-client-install**. Il est également possible d'exécuter ces étapes séparément, ce qui permet aux administrateurs de préparer les machines et l'IdM avant de configurer les clients. Cela permet des scénarios d'installation plus flexibles, y compris des déploiements en masse.
Disabling a client	L'hôte doit avoir une entrée dans IdM. L'hôte doit avoir un keytab actif.	Lorsque vous souhaitez retirer temporairement l'hôte de la zone IdM, par exemple à des fins de maintenance.	**ipa host-disable host_name**
Enabling a client	L'hôte doit avoir une entrée dans IdM.	Lorsque vous souhaitez que l'hôte temporairement désactivé redevienne actif.	**ipa-getkeytab**
Re-enrolling a client	L'hôte doit avoir une entrée dans IdM.	Lorsque l'hôte d'origine a été perdu mais que vous avez installé un hôte avec le même nom d'hôte.	**ipa-client-install --keytab** ou **ipa-client-install --force-join**
Un-enrolling a client	L'hôte doit avoir une entrée dans IdM.	Lorsque vous souhaitez supprimer définitivement l'hôte de la zone IdM.	**ipa-client-install --uninstall**

Tableau 38.4. Opérations d'accueil partie 2

Action	Sur quelle machine l'administrateur peut-il exécuter la (les) commande(s) ?	Que se passe-t-il lorsque l'action est exécutée ? Quelles sont les conséquences pour le fonctionnement de l'hôte dans l'IdM ? Quelles sont les limitations introduites/supprimées ?
Enrolling a client	Dans le cas d'une inscription en deux étapes : **ipa host-add** peut être exécuté sur n'importe quel client IdM ; la deuxième étape de **ipa-client-install** doit être exécutée sur le client lui-même	Par défaut, SSSD est configuré pour se connecter à un serveur IdM pour l'authentification et l'autorisation. En option, il est possible de configurer le module d'authentification enfichable (PAM) et le service de commutation de noms (NSS) pour qu'ils fonctionnent avec un serveur IdM via Kerberos et LDAP.
Disabling a client	Toute machine dans l'IdM, même l'hôte lui-même	La clé Kerberos et le certificat SSL de l'hôte sont invalidés et tous les services fonctionnant sur l'hôte sont désactivés.
Enabling a client	Toute machine dans IdM. S'il est exécuté sur l'hôte désactivé, les informations d'identification LDAP doivent être fournies.	La clé Kerberos de l'hôte et le certificat SSL sont à nouveau valides, et tous les services IdM fonctionnant sur l'hôte sont réactivés.
Re-enrolling a client	L'hôte à réinscrire. Les informations d'identification LDAP doivent être fournies.	Une nouvelle clé Kerberos est générée pour l'hôte, remplaçant la précédente.
Un-enrolling a client	L'hôte à désinscrire.	La commande déconfigure IdM et tente de ramener la machine à son état antérieur. Une partie de ce processus consiste à désenrôler l'hôte du serveur IdM. La désinscription consiste à désactiver la clé principale sur le serveur IdM. Le principal de la machine dans **/etc/krb5.keytab** (**host/<fqdn>@REALM**) est utilisé pour s'authentifier auprès du serveur IdM afin de se désenrôler. Si ce principal n'existe pas, le désenrôlement échouera et un administrateur devra désactiver le principal de l'hôte (**ipa host-disable <fqdn>**).

38.6. ENTRÉE DE L'HÔTE DANS IDM LDAP

Cette section décrit l'aspect d'une entrée d'hôte dans la gestion des identités (IdM) et les attributs qu'elle peut contenir.

Une entrée d'hôte LDAP contient toutes les informations pertinentes sur le client au sein de l'IdM :

- Entrées de service associées à l'hôte
- Le principal de l'hôte et du service
- Règles de contrôle d'accès
- Informations sur la machine, telles que son emplacement physique et son système d'exploitation

NOTE

Notez que l'onglet **Identity → Hosts** de l'interface Web d'IdM n'affiche pas toutes les informations relatives à un hôte particulier stockées dans le LDAP d'IdM.

Propriétés de configuration de l'entrée hôte

Une entrée d'hôte peut contenir des informations sur l'hôte en dehors de sa configuration système, telles que son emplacement physique, son adresse MAC, ses clés et ses certificats.

Ces informations peuvent être définies lors de la création de l'entrée d'hôte si celle-ci est créée manuellement. Sinon, la plupart de ces informations peuvent être ajoutées à l'entrée d'hôte après l'enregistrement de l'hôte dans le domaine.

Tableau 38.5. Propriétés de configuration de l'hôte

Champ de l'interface utilisateur	Option de la ligne de commande	Description
Description	**--desc**=*description*	Une description de l'hôte.
Localité	**--locality**=*locality*	L'emplacement géographique de l'hôte.
Location	**--location**=*location*	L'emplacement physique de l'hôte, par exemple le rack de son centre de données.
Plate-forme	**--platform**=*string*	Le matériel ou l'architecture de l'hôte.
Système d'exploitation	**--os**=*string*	Le système d'exploitation et la version de l'hôte.
Adresse MAC	**--macaddress**=*address*	L'adresse MAC de l'hôte. Il s'agit d'un attribut à valeurs multiples. L'adresse MAC est utilisée par le plug-in NIS pour créer une carte NIS **ethers** pour l'hôte.

Champ de l'interface utilisateur	Option de la ligne de commande	Description
Clés publiques SSH	**--sshpubkey**=*string*	La clé publique SSH complète de l'hôte. Il s'agit d'un attribut à valeurs multiples, de sorte que plusieurs clés peuvent être définies.
Nom du principal (non modifiable)	**--principalname**=*principal*	Le nom du principal Kerberos pour l'hôte. Par défaut, il s'agit du nom de l'hôte lors de l'installation du client, à moins qu'un autre principal ne soit explicitement défini dans **-p**. Ce nom peut être modifié à l'aide des outils de ligne de commande, mais ne peut pas être modifié dans l'interface utilisateur.
Définition d'un mot de passe à usage unique	**--password**=*string*	Cette option définit un mot de passe pour l'hôte, qui peut être utilisé pour l'inscription en masse.
-	**--random**	Cette option génère un mot de passe aléatoire qui sera utilisé pour l'inscription en bloc.
-	**--certificate**=*string*	Un blob de certificat pour l'hôte.
-	**--updatedns**	Ce paramètre indique si l'hôte peut mettre à jour dynamiquement ses entrées DNS en cas de changement d'adresse IP.

38.7. AJOUT D'ENTRÉES D'HÔTES IDM À PARTIR DE LA CLI IDM

Cette section décrit comment ajouter des entrées d'hôte dans la gestion des identités (IdM) à l'aide de l'interface de ligne de commande (CLI).

Les entrées d'hôte sont créées à l'aide de la commande **host-add**. Cette commande ajoute l'entrée de l'hôte au serveur d'annuaire IdM. Consultez la page de manuel **ipa host** en tapant **ipa help host** dans votre CLI pour obtenir la liste complète des options disponibles avec **host-add**.

L'ajout d'un hôte à l'IdM peut se faire selon différents scénarios :

- Dans sa version la plus simple, il suffit de spécifier le nom d'hôte du client pour l'ajouter au domaine Kerberos et pour créer une entrée dans le serveur LDAP IdM :

```
$ ipa host-add client1.example.com
```

- Si le serveur IdM est configuré pour gérer le DNS, ajoutez l'hôte aux enregistrements de ressources DNS en utilisant l'option **--ip-address**.

  Exemple 38.1. Création d'entrées hôtes avec des adresses IP statiques

  ```
  $ ipa host-add --ip-address=192.168.166.31 client1.example.com
  ```

- Si l'hôte à ajouter n'a pas d'adresse IP statique ou si l'adresse IP n'est pas connue au moment de la configuration du client, utilisez l'option **--force** avec la commande **ipa host-add**.

  Exemple 38.2. Création d'entrées hôtes avec DHCP

  ```
  $ ipa host-add --force client1.example.com
  ```

  Par exemple, les ordinateurs portables peuvent être préconfigurés comme clients IdM, mais ils n'ont pas d'adresse IP au moment où ils sont configurés. L'utilisation de **--force** crée essentiellement une entrée de remplacement dans le service DNS IdM. Lorsque le service DNS met à jour dynamiquement ses enregistrements, l'adresse IP actuelle de l'hôte est détectée et son enregistrement DNS est mis à jour.

38.8. SUPPRESSION DES ENTRÉES D'HÔTES DANS LA CLI DE L'IDM

- Utilisez la commande **host-del** pour supprimer les enregistrements d'hôtes. Si votre domaine IdM dispose d'un DNS intégré, utilisez l'option **--updatedns** pour supprimer du DNS les enregistrements associés à l'hôte, quels qu'ils soient :

  ```
  $ ipa host-del --updatedns client1.example.com
  ```

38.9. RÉINSCRIPTION D'UN CLIENT DE LA GESTION DE L'IDENTITÉ

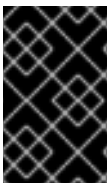
Cette section décrit les différentes façons de réinscrire un client de gestion d'identité.

38.9.1. Réinscription du client à l'IdM

Cette section décrit comment réinscrire un client de gestion d'identité (IdM).

Si une machine cliente a été détruite et a perdu la connexion avec les serveurs IdM, par exemple en raison d'une défaillance matérielle du client, et que vous disposez toujours de son keytab, vous pouvez réinscrire le client. Dans ce scénario, vous souhaitez que le client soit réintégré dans l'environnement IdM avec le même nom d'hôte.

Pendant le réenrôlement, le client génère une nouvelle clé Kerberos et des clés SSH, mais l'identité du client dans la base de données LDAP reste inchangée. Après le réenrôlement, l'hôte a ses clés et d'autres informations dans le même objet LDAP avec le même **FQDN** que précédemment, avant la perte de connexion de la machine avec les serveurs IdM.



IMPORTANT

Vous ne pouvez réinscrire que les clients dont l'entrée de domaine est encore active. Si vous avez désinstallé un client (à l'aide de **ipa-client-install --uninstall**) ou désactivé son entrée d'hôte (à l'aide de **ipa host-disable**), vous ne pouvez pas le réinscrire.

Vous ne pouvez pas réinscrire un client après l'avoir renommé. En effet, dans la gestion de l'identité, l'attribut clé de l'entrée du client dans LDAP est le nom d'hôte du client, son **FQDN**. Contrairement à la réinscription d'un client, au cours de laquelle l'objet LDAP du client reste inchangé, le résultat du renommage d'un client est que ses clés et autres informations se trouvent dans un objet LDAP différent avec un nouveau **FQDN**. La seule façon de renommer un client est donc de désinstaller l'hôte de l'IdM, de changer son nom d'hôte et de l'installer en tant que client IdM avec un nouveau nom. Pour plus de détails sur la manière de renommer un client, voir Renommer les systèmes clients de gestion d'identité .

Que se passe-t-il lors de la réinscription d'un client ?

Lors de la réinscription, Identity Management :

- Révoque le certificat d'origine de l'hôte
- Création de nouvelles clés SSH
- Génère un nouveau keytab

38.9.2. Réinscription d'un client à l'aide des informations d'identification de l'utilisateur : Réinscription interactive

Cette procédure décrit le réenrôlement d'un client de gestion d'identité de manière interactive en utilisant les informations d'identification d'un utilisateur autorisé.

1. Recréez la machine cliente avec le même nom d'hôte.
2. Exécutez la commande **ipa-client-install --force-join** sur la machine cliente :

```
# ipa-client-install --force-join
```

3. Le script demande un utilisateur dont l'identité sera utilisée pour réinscrire le client. Il peut s'agir, par exemple, d'un utilisateur **hostadmin** ayant le rôle d'administrateur d'inscription :

```
User authorized to enroll computers: hostadmin
Password for hostadmin@EXAMPLE.COM:
```

Ressources supplémentaires

- Voir Installation d'un client à l'aide des informations d'identification de l'utilisateur : Installation interactive sur *Installing Identity Management* .

38.9.3. Réinscription d'un client à l'aide de la base de données du client : Réinscription non interactive

Conditions préalables

- Sauvegardez le fichier keytab original du client, par exemple dans le répertoire **/tmp** ou **/root**.

Procédure

Cette procédure décrit le réenrôlement d'un client Identity Management (IdM) de manière non interactive en utilisant le keytab du système client. Par exemple, le réenregistrement à l'aide du keytab du client est approprié pour une installation automatisée.

1. Recréez la machine cliente avec le même nom d'hôte.

2. Copiez le fichier keytab de l'emplacement de sauvegarde dans le répertoire **/etc/** sur l'ordinateur client recréé.

3. Utilisez l'utilitaire **ipa-client-install** pour réinscrire le client et spécifiez l'emplacement du fichier keytab à l'aide de l'option **--keytab**:

```
# ipa-client-install --keytab /etc/krb5.keytab
```

NOTE

Le keytab spécifié dans l'option **--keytab** n'est utilisé que lors de l'authentification pour initier l'inscription. Lors de la réinscription, l'IdM génère un nouveau keytab pour le client.

38.9.4. Test d'un client de gestion d'identité après installation

L'interface de ligne de commande vous informe que l'adresse **ipa-client-install** a été utilisée avec succès, mais vous pouvez également effectuer votre propre test.

Pour vérifier que le client de gestion des identités peut obtenir des informations sur les utilisateurs définis sur le serveur, vérifiez que vous pouvez résoudre un utilisateur défini sur le serveur. Par exemple, pour vérifier l'utilisateur par défaut **admin**:

```
[user@client1 ~]$ id admin
uid=1254400000(admin) gid=1254400000(admins) groups=1254400000(admins)
```

Pour tester que l'authentification fonctionne correctement, **su -** en tant qu'autre utilisateur IdM :

```
[user@client1 ~]$ su - idm_user
Last login: Thu Oct 18 18:39:11 CEST 2018 from 192.168.122.1 on pts/0
[idm_user@client1 ~]$
```

38.10. RENOMMER LES SYSTÈMES CLIENTS DE GESTION DE L'IDENTITÉ

Les sections suivantes décrivent comment modifier le nom d'hôte d'un système client de gestion des identités.

AVERTISSEMENT

Renommer un client est une procédure manuelle. Ne l'effectuez que si la modification du nom d'hôte est absolument nécessaire.

Renommer un client de gestion d'identité implique :

1. Préparation de l'hôte. Pour plus de détails, voir Préparation d'un client IdM pour son renommage .

2. Désinstallation du client IdM de l'hôte. Pour plus de détails, voir Désinstallation d'un client Identity Management.

3. Renommer l'hôte. Pour plus d'informations, voir Renommer le système hôte .

4. Installer le client IdM sur l'hôte avec le nouveau nom. Pour plus de détails, voir Installation d'un client Identity Management à l'adresse *Installing Identity Management* .

5. Configurer l'hôte après l'installation du client IdM. Pour plus d'informations, voir Réajout de services, re-génération de certificats et réajout de groupes d'hôtes.

38.10.1. Préparation d'un client IdM pour son renommage

Avant de désinstaller le client actuel, prenez note de certains paramètres du client. Vous appliquerez cette configuration après avoir réenrôlé la machine avec un nouveau nom d'hôte.

- Identifier les services en cours d'exécution sur la machine :
  - Utilisez la commande **ipa service-find** et identifiez les services avec des certificats dans le résultat :

    ```
    $ ipa service-find old-client-name.example.com
    ```

  - En outre, chaque hôte a un *host service* par défaut qui n'apparaît pas dans la sortie **ipa service-find**. Le principal du service de l'hôte, également appelé *host principal*, est le suivant **host/*old-client-name.example.com***.

- Pour tous les mandants de service affichés par **ipa service-find *old-client-name.example.com***détermine l'emplacement des keytabs correspondants sur le système ***old-client-name.example.com*** système :

  ```
  # find / -name "*.keytab"
  ```

  Chaque service du système client possède un principal Kerberos sous la forme *service_name/host_name@REALM*, tel que **ldap/*old-client-name.example.com@EXAMPLE.COM***.

- Identifier tous les groupes d'hôtes auxquels la machine appartient.

  ```
  # ipa hostgroup-find old-client-name.example.com
  ```

38.10.2. Désinstallation d'un client de gestion d'identité

La désinstallation d'un client supprime ce dernier du domaine de gestion des identités, ainsi que toute la configuration spécifique de gestion des identités des services système, tels que System Security Services Daemon (SSSD). La configuration précédente du système client est ainsi rétablie.

Procédure

1. Exécutez la commande **ipa-client-install --uninstall**:

   ```
   [root@client]# ipa-client-install --uninstall
   ```

2. Supprimez manuellement du serveur les entrées DNS pour l'hôte du client :

```
[root@server]# ipa dnsrecord-del
Record name: old-client-client
Zone name: idm.example.com
No option to delete specific record provided.
Delete all? Yes/No (default No): yes
----------------------
Deleted record "old-client-name"
```

3. Pour chaque keytab identifié autre que **/etc/krb5.keytab**, supprimer les anciens mandants :

```
[root@client ~]# ipa-rmkeytab -k /path/to/keytab -r EXAMPLE.COM
```

4. Sur un serveur IdM, supprimez l'entrée de l'hôte. Cette opération supprime tous les services et révoque tous les certificats émis pour cet hôte :

```
[root@server ~]# ipa host-del client.example.com
```

38.10.3. Renommer le système hôte

Renommez la machine comme vous le souhaitez. Par exemple :

```
[root@client]# hostnamectl set-hostname new-client-name.example.com
```

Vous pouvez maintenant réinstaller le client de gestion des identités sur le domaine de gestion des identités avec le nouveau nom d'hôte.

38.10.4. Réajustement des services, re-génération des certificats et réajustement des groupes d'hôtes

Procédure

1. Sur le serveur de gestion des identités (IdM), ajouter un nouveau keytab pour chaque service identifié dans la préparation d'un client IdM pour son renommage .

```
[root@server ~]# ipa service-add service_name/new-client-name
```

2. Générer des certificats pour les services auxquels un certificat a été attribué dans le cadre de la préparation d'un client IdM à son renommage . Vous pouvez le faire :
   - Utilisation des outils d'administration de l'IdM
   - Utilisation de l'utilitaire **certmonger**
3. Réajoutez le client aux groupes d'hôtes identifiés dans la section Préparation d'un client IdM pour son renommage.

38.11. DÉSACTIVATION ET RÉACTIVATION DES ENTRÉES HÔTES

Cette section décrit comment désactiver et réactiver les hôtes dans la gestion des identités (IdM).

38.11.1. Désactivation des hôtes

Suivez cette procédure pour désactiver une entrée d'hôte dans IdM.

Les services de domaine, les hôtes et les utilisateurs peuvent accéder à un hôte actif. Il peut arriver qu'il soit nécessaire de supprimer temporairement un hôte actif, pour des raisons de maintenance, par exemple. La suppression de l'hôte dans de telles situations n'est pas souhaitable car elle supprime l'entrée de l'hôte et toute la configuration associée de façon permanente. Choisissez plutôt l'option de désactivation de l'hôte.

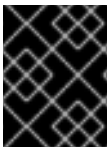
La désactivation d'un hôte empêche les utilisateurs du domaine d'y accéder sans le supprimer définitivement du domaine.

Procédure

- Désactiver un hôte à l'aide de la commande **host-disable**. La désactivation d'un hôte tue les keytabs actifs de l'hôte. Par exemple :

```
$ kinit admin
$ ipa host-disable client.example.com
```

La désactivation d'un hôte entraîne son indisponibilité pour tous les utilisateurs, hôtes et services IdM.



IMPORTANT

La désactivation d'une entrée d'hôte ne désactive pas seulement cet hôte. Elle désactive également tous les services configurés sur cet hôte.

38.11.2. Réactivation des hôtes

Cette section décrit comment réactiver un hôte IdM désactivé.

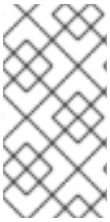
La désactivation d'un hôte a tué ses keytabs actifs, ce qui a supprimé l'hôte du domaine IdM sans toucher autrement à son entrée de configuration.

Procédure

- Pour réactiver un hôte, utilisez la commande **ipa-getkeytab**, en ajoutant :
  - l'option **-s** pour spécifier le serveur IdM auprès duquel le keytab doit être demandé
  - l'option **-p** pour spécifier le nom du principal
  - l'option **-k** pour spécifier le fichier dans lequel enregistrer le keytab.

Par exemple, pour demander un nouveau keytab hôte à **server.example.com** pour **client.example.com**, et stocker le keytab dans le fichier **/etc/krb5.keytab**:

```
$ ipa-getkeytab -s server.example.com -p host/client.example.com -k /etc/krb5.keytab -D
"cn=directory manager" -w password
```



NOTE

Vous pouvez également utiliser les informations d'identification de l'administrateur en spécifiant **-D "uid=admin,cn=users,cn=accounts,dc=example,dc=com"**. Il est important que les informations d'identification correspondent à un utilisateur autorisé à créer le fichier keytab pour l'hôte.

Si la commande **ipa-getkeytab** est exécutée sur un client ou un serveur IdM actif, elle peut être exécutée sans aucun justificatif LDAP (**-D** et **-w**) si l'utilisateur dispose d'un TGT obtenu, par exemple, à l'aide de **kinit admin**. Pour exécuter la commande directement sur l'hôte désactivé, il faut fournir des informations d'identification LDAP pour s'authentifier auprès du serveur IdM.

CHAPITRE 39. AJOUT D'ENTRÉES D'HÔTES À PARTIR DE L'INTERFACE WEB IDM

Ce chapitre présente les hôtes dans la gestion des identités (IdM) et l'opération d'ajout d'une entrée d'hôte dans l'interface Web IdM.

39.1. HÔTES DANS L'IDM

La gestion des identités (IdM) gère ces identités :

- Utilisateurs
- Services
- Hosts

Un hôte représente une machine. En tant qu'identité IdM, un hôte possède une entrée dans le LDAP IdM, c'est-à-dire l'instance 389 Directory Server du serveur IdM.

L'entrée de l'hôte dans IdM LDAP est utilisée pour établir des relations entre d'autres hôtes et même des services au sein du domaine. Ces relations font partie de l'autorisation et du contrôle des hôtes au sein du domaine (*delegating*). Tout hôte peut être utilisé dans les règles **host-based access control** (HBAC).

Le domaine IdM établit une communauté entre les machines, avec des informations d'identité communes, des politiques communes et des services partagés. Toute machine appartenant à un domaine fonctionne comme un client du domaine, ce qui signifie qu'elle utilise les services fournis par le domaine. Le domaine IdM fournit trois services principaux spécifiquement destinés aux machines :

- DNS
- Kerberos
- Gestion des certificats

Dans l'IdM, les hôtes sont étroitement liés aux services qui y sont exécutés :

- Les entrées de service sont associées à un hôte.
- Un hôte stocke les principaux Kerberos de l'hôte et du service.

39.2. INSCRIPTION AU PROGRAMME D'ACCUEIL

Cette section décrit l'enrôlement des hôtes en tant que clients IdM et ce qui se passe pendant et après l'enrôlement. Elle compare l'enrôlement des hôtes IdM et des utilisateurs IdM. Elle décrit également les autres types d'authentification disponibles pour les hôtes.

L'inscription d'un hôte consiste à

- Création d'une entrée d'hôte dans IdM LDAP : éventuellement en utilisant la commande**ipa host-add** dans IdM CLI, ou l'opération équivalente dans IdM Web UI.
- Configuration des services IdM sur l'hôte, par exemple le System Security Services Daemon (SSSD), Kerberos et certmonger, et connexion de l'hôte au domaine IdM.

Les deux actions peuvent être effectuées séparément ou ensemble.

S'ils sont exécutés séparément, ils permettent de répartir les deux tâches entre deux utilisateurs ayant des niveaux de privilèges différents. Cela est utile pour les déploiements en masse.

La commande **ipa-client-install** peut effectuer les deux actions ensemble. La commande crée une entrée d'hôte dans IdM LDAP si cette entrée n'existe pas encore, et configure les services Kerberos et SSSD pour l'hôte. La commande amène l'hôte dans le domaine IdM et lui permet d'identifier le serveur IdM auquel il se connectera. Si l'hôte appartient à une zone DNS gérée par IdM, **ipa-client-install** ajoute également des enregistrements DNS pour l'hôte. La commande doit être exécutée sur le client.

39.3. PRIVILÈGES DE L'UTILISATEUR REQUIS POUR L'INSCRIPTION DE L'HÔTE

L'opération d'enrôlement des hôtes nécessite une authentification afin d'éviter qu'un utilisateur non privilégié n'ajoute des machines indésirables au domaine IdM. Les privilèges requis dépendent de plusieurs facteurs, par exemple :

- Si une entrée d'hôte est créée séparément de l'exécution de **ipa-client-install**
- Si un mot de passe à usage unique (OTP) est utilisé pour l'inscription

Privilèges de l'utilisateur pour la création manuelle facultative d'une entrée d'hôte dans IdM LDAP

Le privilège d'utilisateur requis pour créer une entrée d'hôte dans IdM LDAP à l'aide de la commande CLI **ipa host-add** ou de l'interface Web IdM est **Host Administrators**. Le privilège **Host Administrators** peut être obtenu par le biais du rôle **IT Specialist**.

Privilèges de l'utilisateur pour l'intégration du client dans le domaine IdM

Les hôtes sont configurés en tant que clients IdM lors de l'exécution de la commande **ipa-client-install**. Le niveau d'habilitation requis pour l'exécution de la commande **ipa-client-install** dépend du scénario d'inscription dans lequel vous vous trouvez :

- L'entrée de l'hôte dans IdM LDAP n'existe pas. Pour ce scénario, vous avez besoin des informations d'identification d'un administrateur complet ou du rôle **Host Administrators**. Un administrateur complet est membre du groupe **admins**. Le rôle **Host Administrators** permet d'ajouter des hôtes et d'enrôler des hôtes. Pour plus d'informations sur ce scénario, voir Installation d'un client à l'aide des informations d'identification de l'utilisateur : installation interactive.
- L'entrée de l'hôte dans IdM LDAP existe. Pour ce scénario, vous avez besoin des informations d'identification d'un administrateur limité pour exécuter **ipa-client-install** avec succès. Dans ce cas, l'administrateur limité a le rôle **Enrollment Administrator**, qui lui confère le privilège **Host Enrollment**. Pour plus d'informations, voir Installation d'un client à l'aide des informations d'identification de l'utilisateur : installation interactive.
- L'entrée de l'hôte dans IdM LDAP existe et un OTP a été généré pour l'hôte par un administrateur complet ou limité. Dans ce cas, vous pouvez installer un client IdM en tant qu'utilisateur ordinaire si vous exécutez la commande **ipa-client-install** avec l'option **--password**, en fournissant l'OTP correct. Pour plus de détails, voir Installation d'un client à l'aide d'un mot de passe à usage unique : installation interactive.

Après l'inscription, les hôtes IdM authentifient chaque nouvelle session pour pouvoir accéder aux ressources IdM. L'authentification de la machine est nécessaire pour que le serveur IdM fasse confiance à la machine et accepte les connexions IdM du logiciel client installé sur cette machine. Après avoir

authentifié le client, le serveur IdM peut répondre à ses demandes.

39.4. COMPARAISON ENTRE L'ENRÔLEMENT ET L'AUTHENTIFICATION DES HÔTES ET DES UTILISATEURS DE L'IDM

Il existe de nombreuses similitudes entre les utilisateurs et les hôtes dans l'IdM. Cette section décrit certaines des similitudes qui peuvent être observées au cours de la phase d'inscription ainsi que celles qui concernent l'authentification au cours de la phase de déploiement.

- La phase d'inscription(inscription de l'utilisateur et de l'hôte) :
  - Un administrateur peut créer une entrée LDAP pour un utilisateur et un hôte avant que l'utilisateur ou l'hôte ne rejoigne l'IdM : pour l'utilisateur de l'étape, la commande est **ipa stageuser-add**; pour l'hôte, la commande est **ipa host-add**.
  - Un fichier contenant un *key table* ou, en abrégé, un keytab, une clé symétrique ressemblant dans une certaine mesure à un mot de passe d'utilisateur, est créé lors de l'exécution de la commande **ipa-client-install** sur l'hôte, ce qui permet à l'hôte de rejoindre le domaine IdM. De manière analogue, un utilisateur est invité à créer un mot de passe lorsqu'il active son compte, rejoignant ainsi le royaume IdM.
  - Alors que le mot de passe de l'utilisateur est la méthode d'authentification par défaut pour un utilisateur, le keytab est la méthode d'authentification par défaut pour un hôte. Le keytab est stocké dans un fichier sur l'hôte.

Tableau 39.1. Inscription des utilisateurs et des hôtes

Action	User	Hôte
Préinscription	$ **ipa stageuser-add** ***user_name*** **[--password]**	$ **ipa host-add** ***host_name*** **[--random]**
Activation du compte	$ **ipa stageuser-activate** ***user_name***	**ipa-client install [--password]** (doit être exécuté sur l'hôte lui-même)

- La phase de déploiement(authentification de la session de l'utilisateur et de l'hôte) :
  - Lorsqu'un utilisateur démarre une nouvelle session, il s'authentifie à l'aide d'un mot de passe ; de même, à chaque fois qu'il est mis sous tension, l'hôte s'authentifie en présentant son fichier keytab. Le System Security Services Daemon (SSSD) gère ce processus en arrière-plan.
  - Si l'authentification est réussie, l'utilisateur ou l'hôte obtient un ticket Kerberos (TGT).
  - Le TGT est ensuite utilisé pour obtenir des billets spécifiques pour des services spécifiques.

Tableau 39.2. Authentification de la session de l'utilisateur et de l'hôte

	User	Hôte
Moyens d'authentification par défaut	**Password**	**Keytabs**
Démarrage d'une session (utilisateur ordinaire)	$ **kinit** ***user_name***	*[switch on the host]*
Le résultat de l'authentification réussie	**TGT** à utiliser pour obtenir l'accès à des services spécifiques	**TGT** à utiliser pour obtenir l'accès à des services spécifiques

Les TGT et autres tickets Kerberos sont générés dans le cadre des services et politiques Kerberos définis par le serveur. L'octroi initial d'un ticket Kerberos, le renouvellement des informations d'identification Kerberos et même la destruction de la session Kerberos sont tous gérés automatiquement par les services IdM.

Autres options d'authentification pour les hôtes IdM

Outre les keytabs, IdM prend en charge deux autres types d'authentification des machines :

- Clés SSH. La clé publique SSH de l'hôte est créée et téléchargée dans l'entrée de l'hôte. À partir de là, le System Security Services Daemon (SSSD) utilise IdM comme fournisseur d'identité et peut travailler en conjonction avec OpenSSH et d'autres services pour référencer les clés publiques situées de manière centralisée dans IdM.
- Certificats de machine. Dans ce cas, la machine utilise un certificat SSL délivré par l'autorité de certification du serveur IdM et stocké dans le serveur d'annuaire IdM. Le certificat est ensuite envoyé à la machine pour qu'elle le présente lorsqu'elle s'authentifie auprès du serveur. Sur le client, les certificats sont gérés par un service appelé certmonger.

39.5. ENTRÉE DE L'HÔTE DANS IDM LDAP

Cette section décrit l'aspect d'une entrée d'hôte dans la gestion des identités (IdM) et les attributs qu'elle peut contenir.

Une entrée d'hôte LDAP contient toutes les informations pertinentes sur le client au sein de l'IdM :

- Entrées de service associées à l'hôte
- Le principal de l'hôte et du service
- Règles de contrôle d'accès
- Informations sur la machine, telles que son emplacement physique et son système d'exploitation

NOTE

Notez que l'onglet **Identity → Hosts** de l'interface Web d'IdM n'affiche pas toutes les informations relatives à un hôte particulier stockées dans le LDAP d'IdM.

Propriétés de configuration de l'entrée hôte

Une entrée d'hôte peut contenir des informations sur l'hôte en dehors de sa configuration système, telles que son emplacement physique, son adresse MAC, ses clés et ses certificats.

Ces informations peuvent être définies lors de la création de l'entrée d'hôte si celle-ci est créée manuellement. Sinon, la plupart de ces informations peuvent être ajoutées à l'entrée d'hôte après l'enregistrement de l'hôte dans le domaine.

Tableau 39.3. Propriétés de configuration de l'hôte

Champ de l'interface utilisateur	Option de la ligne de commande	Description
Description	**--desc**=*description*	Une description de l'hôte.
Localité	**--locality**=*locality*	L'emplacement géographique de l'hôte.
Location	**--location**=*location*	L'emplacement physique de l'hôte, par exemple le rack de son centre de données.
Plate-forme	**--platform**=*string*	Le matériel ou l'architecture de l'hôte.
Système d'exploitation	**--os**=*string*	Le système d'exploitation et la version de l'hôte.
Adresse MAC	**--macaddress**=*address*	L'adresse MAC de l'hôte. Il s'agit d'un attribut à valeurs multiples. L'adresse MAC est utilisée par le plug-in NIS pour créer une carte NIS **ethers** pour l'hôte.
Clés publiques SSH	**--sshpubkey**=*string*	La clé publique SSH complète de l'hôte. Il s'agit d'un attribut à valeurs multiples, de sorte que plusieurs clés peuvent être définies.
Nom du principal (non modifiable)	**--principalname**=*principal*	Le nom du principal Kerberos pour l'hôte. Par défaut, il s'agit du nom de l'hôte lors de l'installation du client, à moins qu'un autre principal ne soit explicitement défini dans **-p**. Ce nom peut être modifié à l'aide des outils de ligne de commande, mais ne peut pas être modifié dans l'interface utilisateur.

Champ de l'interface utilisateur	Option de la ligne de commande	Description
Définition d'un mot de passe à usage unique	**--password**=*string*	Cette option définit un mot de passe pour l'hôte, qui peut être utilisé pour l'inscription en masse.
-	**--random**	Cette option génère un mot de passe aléatoire qui sera utilisé pour l'inscription en bloc.
-	**--certificate**=*string*	Un blob de certificat pour l'hôte.
-	**--updatedns**	Ce paramètre indique si l'hôte peut mettre à jour dynamiquement ses entrées DNS en cas de changement d'adresse IP.

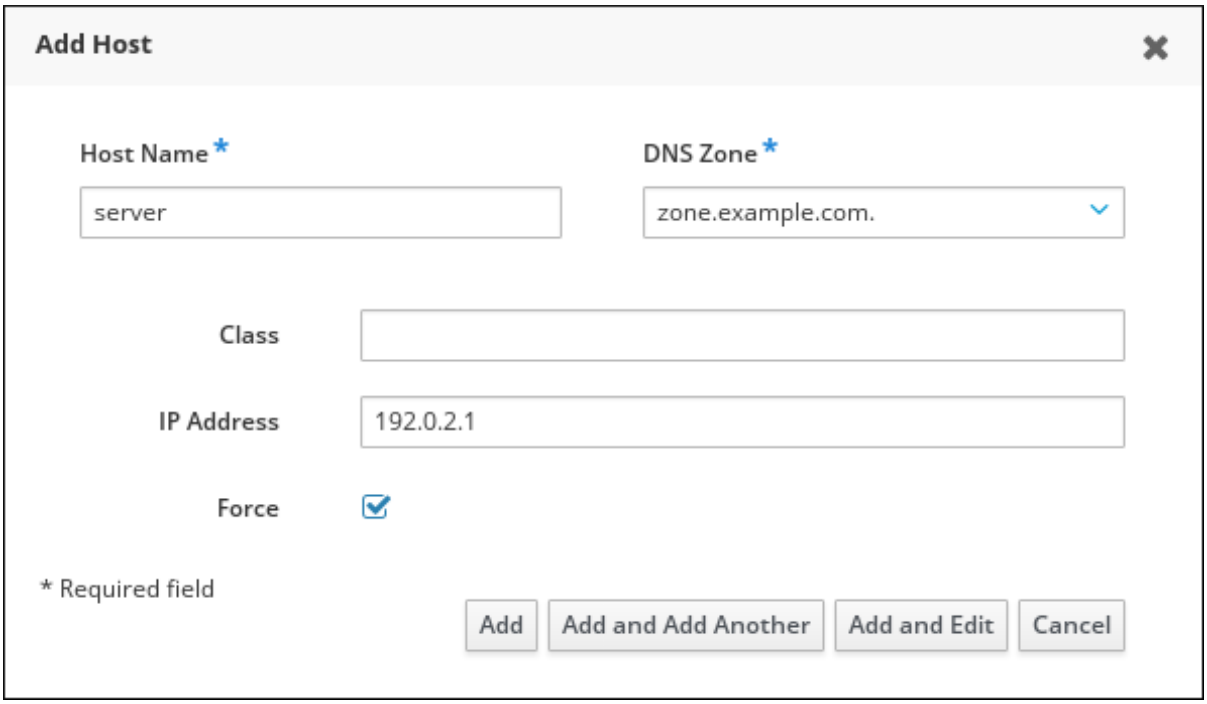
39.6. AJOUTER DES ENTRÉES D'HÔTE À PARTIR DE L'INTERFACE WEB

1. Ouvrez l'onglet **Identity** et sélectionnez le sous-onglet **Hosts**.
2. Cliquez sur **Ajouter** en haut de la liste des hôtes.

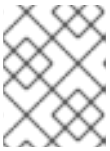
Figure 39.1. Ajout d'entrées d'hôtes

3. Entrez le nom de la machine et sélectionnez le domaine parmi les zones configurées dans la liste déroulante. Si une adresse IP statique a déjà été attribuée à l'hôte, incluez-la dans l'entrée de l'hôte afin que l'entrée DNS soit entièrement créée.
Le champ **Class** n'a pas d'utilité particulière pour le moment.

Figure 39.2. Assistant d'ajout d'hôte



Les zones DNS peuvent être créées dans IdM. Si le serveur IdM ne gère pas le serveur DNS, la zone peut être saisie manuellement dans la zone de menu, comme un champ de texte normal.

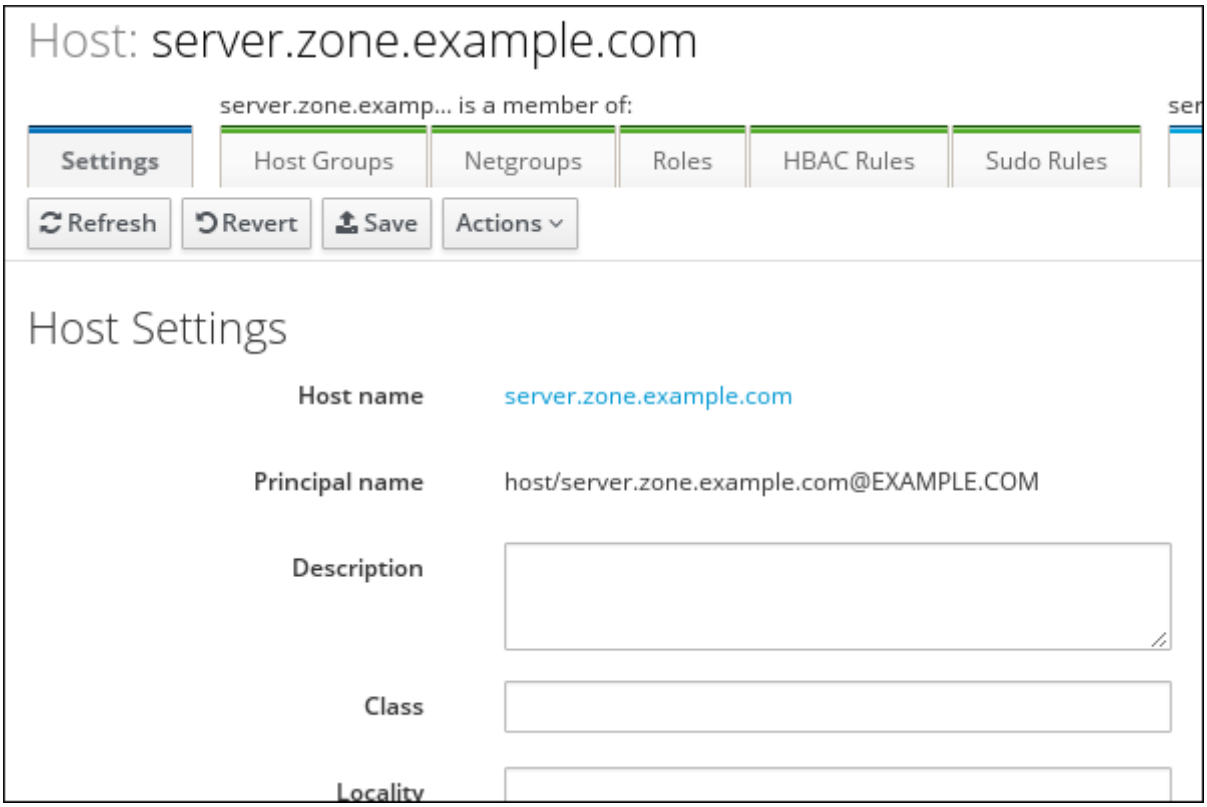


NOTE

Cochez la case **Force** si vous souhaitez ne pas vérifier si l'hôte peut être résolu par DNS.

4. Cliquez sur le bouton **Add and Edit** pour accéder directement à la page d'entrée étendue et saisir davantage d'informations sur les attributs. Des informations sur le matériel et l'emplacement physique de l'hôte peuvent être incluses dans l'entrée de l'hôte.

Figure 39.3. Page d'entrée élargie



CHAPITRE 40. GÉRER LES HÔTES À L'AIDE DES PLAYBOOKS ANSIBLE

Ansible est un outil d'automatisation utilisé pour configurer des systèmes, déployer des logiciels et effectuer des mises à jour continues. Ansible prend en charge la gestion des identités (IdM) et vous pouvez utiliser des modules Ansible pour automatiser la gestion des hôtes.

Ce chapitre décrit les concepts suivants et les opérations effectuées lors de la gestion des hôtes et des entrées d'hôtes à l'aide des playbooks Ansible :

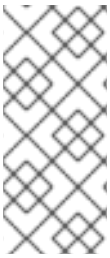
- Assurer la présence d'entrées d'hôtes IdM qui ne sont définies que par leur **FQDNs**
- S'assurer de la présence d'entrées d'hôtes IdM avec des adresses IP
- Assurer la présence de plusieurs entrées d'hôtes IdM avec des mots de passe aléatoires
- Assurer la présence d'une entrée d'hôte IdM avec plusieurs adresses IP
- Garantir l'absence d'entrées d'hôtes IdM

40.1. S'ASSURER DE LA PRÉSENCE D'UNE ENTRÉE D'HÔTE IDM AVEC FQDN À L'AIDE DES PLAYBOOKS ANSIBLE

Cette section décrit comment assurer la présence d'entrées d'hôtes dans la gestion des identités (IdM) à l'aide de playbooks Ansible. Les entrées d'hôte sont uniquement définies par leur **fully-qualified domain names** (FQDN).

Il suffit de spécifier le nom **FQDN** de l'hôte si au moins l'une des conditions suivantes s'applique :

- Le serveur IdM n'est pas configuré pour gérer les DNS.
- L'hôte n'a pas d'adresse IP statique ou l'adresse IP n'est pas connue au moment de la configuration de l'hôte. L'ajout d'un hôte défini uniquement par une adresse **FQDN** crée essentiellement une entrée de remplacement dans le service DNS IdM. Par exemple, des ordinateurs portables peuvent être préconfigurés comme clients IdM, mais ils n'ont pas d'adresse IP au moment de leur configuration. Lorsque le service DNS met à jour dynamiquement ses enregistrements, l'adresse IP actuelle de l'hôte est détectée et son enregistrement DNS est mis à jour.



NOTE

Sans Ansible, les entrées d'hôtes sont créées dans IdM à l'aide de la commande **ipa host-add**. Le résultat de l'ajout d'un hôte à IdM est l'état de l'hôte présent dans IdM. En raison de la dépendance d'Ansible à l'égard de l'idempotence, pour ajouter un hôte à IdM à l'aide d'Ansible, vous devez créer un playbook dans lequel vous définissez l'état de l'hôte comme étant présent : **state: present**.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.

- Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
- L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
- L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

Procédure

1. Créez un fichier d'inventaire, par exemple **inventory.file**, et définissez-y **ipaserver**:

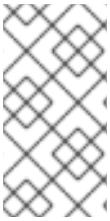
```
[ipaserver]
server.idm.example.com
```

2. Créez un fichier Ansible playbook avec le **FQDN** de l'hôte dont vous voulez assurer la présence dans IdM. Pour simplifier cette étape, vous pouvez copier et modifier l'exemple dans le fichier **/usr/share/doc/ansible-freeipa/playbooks/host/add-host.yml**:

```
---
- name: Host present
  hosts: ipaserver

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Host host01.idm.example.com present
    ipahost:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: host01.idm.example.com
      state: present
      force: yes
```

3. Exécutez le manuel de jeu :

```
$ ansible-playbook --vault-password-file=password_file -v -i
path_to_inventory_directory/inventory.file path_to_playbooks_directory/ensure-host-
is-present.yml
```



NOTE

La procédure aboutit à la création d'une entrée d'hôte dans le serveur LDAP IdM, mais pas à l'enrôlement de l'hôte dans le domaine Kerberos IdM. Pour cela, vous devez déployer l'hôte en tant que client IdM. Pour plus de détails, voir Installation d'un client de gestion d'identité à l'aide d'un playbook Ansible.

Verification steps

1. Connectez-vous à votre serveur IdM en tant qu'administrateur :

```
$ ssh admin@server.idm.example.com
Password:
```

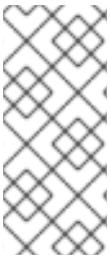
2. Entrez la commande **ipa host-show** et indiquez le nom de l'hôte :

```
$ ipa host-show host01.idm.example.com
  Host name: host01.idm.example.com
  Principal name: host/host01.idm.example.com@IDM.EXAMPLE.COM
  Principal alias: host/host01.idm.example.com@IDM.EXAMPLE.COM
  Password: False
  Keytab: False
  Managed by: host01.idm.example.com
```

Le résultat confirme que **host01.idm.example.com** existe dans IdM.

40.2. ASSURER LA PRÉSENCE D'UNE ENTRÉE D'HÔTE IDM AVEC DES INFORMATIONS DNS EN UTILISANT LES PLAYBOOKS ANSIBLE

Cette section décrit comment assurer la présence d'entrées d'hôtes dans la gestion des identités (IdM) à l'aide des playbooks Ansible. Les entrées d'hôte sont définies par leur **fully-qualified domain names** (FQDN) et leurs adresses IP.



NOTE

Sans Ansible, les entrées d'hôtes sont créées dans IdM à l'aide de la commande **ipa host-add**. Le résultat de l'ajout d'un hôte à IdM est l'état de l'hôte présent dans IdM. En raison de la dépendance d'Ansible à l'égard de l'idempotence, pour ajouter un hôte à IdM à l'aide d'Ansible, vous devez créer un playbook dans lequel vous définissez l'état de l'hôte comme étant présent : **state: present**.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

Procédure

1. Créez un fichier d'inventaire, par exemple **inventory.file**, et définissez-y **ipaserver**:

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

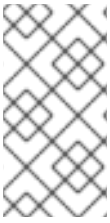
2. Créez un fichier playbook Ansible avec le **fully-qualified domain name** (FQDN) de l'hôte dont vous voulez assurer la présence dans l'IdM. En outre, si le serveur IdM est configuré pour gérer le DNS et que vous connaissez l'adresse IP de l'hôte, spécifiez une valeur pour le paramètre **ip_address**. L'adresse IP est nécessaire pour que l'hôte existe dans les enregistrements de

ressources DNS. Pour simplifier cette étape, vous pouvez copier et modifier l'exemple dans le fichier **/usr/share/doc/ansible-freeipa/playbooks/host/host-present.yml**. Vous pouvez également inclure d'autres informations supplémentaires :

```
---
- name: Host present
  hosts: ipaserver

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Ensure host01.idm.example.com is present
    ipahost:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: host01.idm.example.com
      description: Example host
      ip_address: 192.168.0.123
      locality: Lab
      ns_host_location: Lab
      ns_os_version: CentOS 7
      ns_hardware_platform: Lenovo T61
      mac_address:
      - "08:00:27:E3:B1:2D"
      - "52:54:00:BD:97:1E"
      state: present
```

3. Exécutez le manuel de jeu :

```
$ ansible-playbook --vault-password-file=password_file -v -i
path_to_inventory_directory/inventory.file path_to_playbooks_directory/ensure-host-
is-present.yml
```



NOTE

La procédure aboutit à la création d'une entrée d'hôte dans le serveur LDAP IdM, mais pas à l'enrôlement de l'hôte dans le domaine Kerberos IdM. Pour cela, vous devez déployer l'hôte en tant que client IdM. Pour plus de détails, voir Installation d'un client de gestion d'identité à l'aide d'un playbook Ansible.

Verification steps

1. Connectez-vous à votre serveur IdM en tant qu'administrateur :

```
$ ssh admin@server.idm.example.com
Password:
```

2. Entrez la commande **ipa host-show** et indiquez le nom de l'hôte :

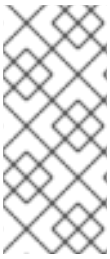
```
$ ipa host-show host01.idm.example.com
  Host name: host01.idm.example.com
  Description: Example host
  Locality: Lab
  Location: Lab
  Platform: Lenovo T61
```

```
Operating system: CentOS 7
Principal name: host/host01.idm.example.com@IDM.EXAMPLE.COM
Principal alias: host/host01.idm.example.com@IDM.EXAMPLE.COM
MAC address: 08:00:27:E3:B1:2D, 52:54:00:BD:97:1E
Password: False
Keytab: False
Managed by: host01.idm.example.com
```

La sortie confirme que **host01.idm.example.com** existe dans IdM.

40.3. ASSURER LA PRÉSENCE DE PLUSIEURS ENTRÉES D'HÔTES IDM AVEC DES MOTS DE PASSE ALÉATOIRES À L'AIDE DES PLAYBOOKS ANSIBLE

Le module **ipahost** permet à l'administrateur système de s'assurer de la présence ou de l'absence d'entrées d'hôtes multiples dans IdM en utilisant une seule tâche Ansible. Cette section décrit comment assurer la présence de plusieurs entrées d'hôtes qui ne sont définies que par leur **fully-qualified domain names** (FQDN). L'exécution du playbook Ansible génère des mots de passe aléatoires pour les hôtes.



NOTE

Sans Ansible, les entrées d'hôtes sont créées dans IdM à l'aide de la commande **ipa host-add**. Le résultat de l'ajout d'un hôte à IdM est l'état de l'hôte présent dans IdM. En raison de la dépendance d'Ansible à l'égard de l'idempotence, pour ajouter un hôte à IdM à l'aide d'Ansible, vous devez créer un playbook dans lequel vous définissez l'état de l'hôte comme étant présent : **state: present**.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

Procédure

1. Créez un fichier d'inventaire, par exemple **inventory.file**, et définissez-y **ipaserver**:

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

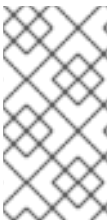
2. Créez un fichier Ansible playbook avec les **fully-qualified domain name** (FQDN) des hôtes dont vous voulez assurer la présence dans IdM. Pour que le playbook Ansible génère un mot de passe aléatoire pour chaque hôte même si l'hôte existe déjà dans IdM et que **update_password**

est limité à **on_create**, ajoutez les options **random: yes** et **force: yes**. Pour simplifier cette étape, vous pouvez copier et modifier l'exemple du fichier Markdown **/usr/share/doc/ansible-freeipa/README-host.md**:

```
---
- name: Ensure hosts with random password
  hosts: ipaserver

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Hosts host01.idm.example.com and host02.idm.example.com present with random
passwords
    ipahost:
      ipaadmin_password: "{{ ipaadmin_password }}"
      hosts:
      - name: host01.idm.example.com
        random: yes
        force: yes
      - name: host02.idm.example.com
        random: yes
        force: yes
    register: ipahost
```

3. Exécutez le manuel de jeu :

```
$ ansible-playbook --vault-password-file=password_file -v -i
path_to_inventory_directory/inventory.file path_to_playbooks_directory/ensure-hosts-
are-present.yml
[...]
TASK [Hosts host01.idm.example.com and host02.idm.example.com present with random
passwords]
changed: [r8server.idm.example.com] => {"changed": true, "host":
{"host01.idm.example.com": {"randompassword": "0HoIRvjUdH0Ycbf6uYdWTxH"},
"host02.idm.example.com": {"randompassword": "5VdLgrf3wvojmACdHC3uA3s"}}}
```



NOTE

Pour déployer les hôtes en tant que clients IdM à l'aide de mots de passe aléatoires à usage unique (OTP), voir Options d'autorisation pour l'inscription d'un client IdM à l'aide d'un playbook Ansible ou Installation d'un client à l'aide d'un mot de passe à usage unique : installation interactive.

Verification steps

1. Connectez-vous à votre serveur IdM en tant qu'administrateur :

```
$ ssh admin@server.idm.example.com
Password:
```

2. Entrez la commande **ipa host-show** et indiquez le nom de l'un des hôtes :

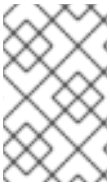
```
$ ipa host-show host01.idm.example.com
  Host name: host01.idm.example.com
```

```
Password: True
Keytab: False
Managed by: host01.idm.example.com
```

La sortie confirme que **host01.idm.example.com** existe dans IdM avec un mot de passe aléatoire.

40.4. ASSURER LA PRÉSENCE D'UNE ENTRÉE D'HÔTE IDM AVEC PLUSIEURS ADRESSES IP EN UTILISANT LES PLAYBOOKS ANSIBLE

Cette section décrit comment assurer la présence d'une entrée d'hôte dans la gestion des identités (IdM) à l'aide des playbooks Ansible. L'entrée d'hôte est définie par son **fully-qualified domain name** (FQDN) et ses multiples adresses IP.



NOTE

Contrairement à l'utilitaire **ipa host**, le module Ansible **ipahost** permet de s'assurer de la présence ou de l'absence de plusieurs adresses IPv4 et IPv6 pour un hôte. La commande **ipa host-mod** ne peut pas gérer les adresses IP.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

Procédure

1. Créez un fichier d'inventaire, par exemple **inventory.file**, et définissez-y **ipaserver**:

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

2. Créez un fichier playbook Ansible. Spécifiez, en tant que **name** de la variable **ipahost**, le **fully-qualified domain name** (FQDN) de l'hôte dont vous voulez assurer la présence dans l'IdM. Spécifiez chacune des multiples valeurs IPv4 et IPv6 **ip_address** sur une ligne séparée en utilisant la syntaxe **- *ip_address*** pour spécifier chacune des valeurs IPv4 et IPv6 sur une ligne séparée. Pour simplifier cette étape, vous pouvez copier et modifier l'exemple dans le fichier **/usr/share/doc/ansible-freeipa/playbooks/host/host-member-ipaddresses-present.yml**. Vous pouvez également inclure des informations supplémentaires :

```
---
- name: Host member IP addresses present
  hosts: ipaserver
```

```
vars_files:
- /home/user_name/MyPlaybooks/secret.yml
tasks:
- name: Ensure host101.example.com IP addresses present
  ipahost:
    ipaadmin_password: "{{ ipaadmin_password }}"
    name: host01.idm.example.com
    ip_address:
    - 192.168.0.123
    - fe80::20c:29ff:fe02:a1b3
    - 192.168.0.124
    - fe80::20c:29ff:fe02:a1b4
    force: yes
```

3. Exécutez le manuel de jeu :

```
$ ansible-playbook --vault-password-file=password_file -v -i
path_to_inventory_directory/inventory.file path_to_playbooks_directory/ensure-host-
with-multiple-IP-addreses-is-present.yml
```

NOTE

La procédure crée une entrée d'hôte dans le serveur LDAP IdM mais n'inscrit pas l'hôte dans le domaine Kerberos IdM. Pour cela, vous devez déployer l'hôte en tant que client IdM. Pour plus de détails, voir Installation d'un client de gestion d'identité à l'aide d'un playbook Ansible.

Verification steps

1. Connectez-vous à votre serveur IdM en tant qu'administrateur :

```
$ ssh admin@server.idm.example.com
Password:
```

2. Entrez la commande **ipa host-show** et indiquez le nom de l'hôte :

```
$ ipa host-show host01.idm.example.com
  Principal name: host/host01.idm.example.com@IDM.EXAMPLE.COM
  Principal alias: host/host01.idm.example.com@IDM.EXAMPLE.COM
  Password: False
  Keytab: False
  Managed by: host01.idm.example.com
```

Le résultat confirme que **host01.idm.example.com** existe dans IdM.

3. Pour vérifier que les adresses IP multiples de l'hôte existent dans les enregistrements DNS IdM, entrez la commande **ipa dnsrecord-show** et spécifiez les informations suivantes :

- Nom du domaine IdM
- The name of the host

```
$ ipa dnsrecord-show idm.example.com host01
```

```
[...]
  Record name: host01
  A record: 192.168.0.123, 192.168.0.124
  AAAA record: fe80::20c:29ff:fe02:a1b3, fe80::20c:29ff:fe02:a1b4
```

La sortie confirme que toutes les adresses IPv4 et IPv6 spécifiées dans le manuel de jeu sont correctement associées à l'entrée d'hôte **host01.idm.example.com**.

40.5. S'ASSURER DE L'ABSENCE D'UNE ENTRÉE D'HÔTE IDM À L'AIDE DES PLAYBOOKS ANSIBLE

Cette section décrit comment garantir l'absence d'entrées d'hôtes dans la gestion des identités (IdM) à l'aide des playbooks Ansible.

Conditions préalables

- Informations d'identification de l'administrateur IdM

Procédure

1. Créez un fichier d'inventaire, par exemple **inventory.file**, et définissez-y **ipaserver**:

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

2. Créez un fichier playbook Ansible avec le **fully-qualified domain name** (FQDN) de l'hôte dont vous voulez garantir l'absence de l'IdM. Si votre domaine IdM a un DNS intégré, utilisez l'option **updatedns: yes** pour supprimer du DNS les enregistrements associés à l'hôte, quels qu'ils soient.
   Pour simplifier cette étape, vous pouvez copier et modifier l'exemple dans le fichier **/usr/share/doc/ansible-freeipa/playbooks/host/delete-host.yml**:

```
---
- name: Host absent
  hosts: ipaserver

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Host host01.idm.example.com absent
    ipahost:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: host01.idm.example.com
      updatedns: yes
      state: absent
```

3. Exécutez le manuel de jeu :

```
$ ansible-playbook --vault-password-file=password_file -v -i
path_to_inventory_directory/inventory.file path_to_playbooks_directory/ensure-host-
absent.yml
```

NOTE

La procédure aboutit à :

- L'hôte n'est pas présent dans le royaume IdM Kerberos.
- L'entrée de l'hôte n'est pas présente dans le serveur LDAP IdM.

Pour supprimer la configuration IdM spécifique des services système, tels que System Security Services Daemon (SSSD), de l'hôte client lui-même, vous devez exécuter la commande **ipa-client-install --uninstall** sur le client. Pour plus de détails, voir Désinstallation d'un client IdM .

Verification steps

1. Connectez-vous à **ipaserver** en tant qu'administrateur :

```
$ ssh admin@server.idm.example.com
Password:
[admin@server /]$
```

2. Afficher des informations sur *host01.idm.example.com*:

```
$ ipa host-show host01.idm.example.com
ipa: ERROR: host01.idm.example.com: host not found
```

Le résultat confirme que l'hôte n'existe pas dans IdM.

40.6. RESSOURCES SUPPLÉMENTAIRES

- Voir le fichier Markdown de **/usr/share/doc/ansible-freeipa/README-host.md**.
- Voir les playbooks supplémentaires dans le répertoire **/usr/share/doc/ansible-freeipa/playbooks/host**.

CHAPITRE 41. GESTION DES GROUPES D'HÔTES À L'AIDE DE LA CLI IDM

Ce chapitre présente les groupes d'hôtes dans la gestion des identités (IdM) et décrit les opérations suivantes pour gérer les groupes d'hôtes et leurs membres dans l'interface de ligne de commande (CLI) :

- Visualisation des groupes d'accueil et de leurs membres
- Création de groupes d'accueil
- Suppression de groupes d'hôtes
- Ajouter des membres à un groupe d'hôtes
- Suppression de membres d'un groupe d'hôtes
- Ajout de gestionnaires de groupes d'hôtes
- Suppression des gestionnaires de groupes d'hôtes

41.1. GROUPES D'ACCUEIL DANS L'IDM

Les groupes d'hôtes IdM peuvent être utilisés pour centraliser le contrôle des tâches de gestion importantes, en particulier le contrôle d'accès.

Définition des groupes d'accueil

Un groupe d'hôtes est une entité qui contient un ensemble d'hôtes IdM avec des règles de contrôle d'accès communes et d'autres caractéristiques. Par exemple, vous pouvez définir des groupes d'hôtes en fonction des départements de l'entreprise, des emplacements physiques ou des exigences en matière de contrôle d'accès.

Un groupe d'hôtes dans IdM peut comprendre

- Serveurs et clients IdM
- Autres groupes d'accueil IdM

Groupes d'hôtes créés par défaut

Par défaut, le serveur IdM crée le groupe d'hôtes **ipaservers** pour tous les hôtes du serveur IdM.

Membres directs et indirects du groupe

Les attributs de groupe dans IdM s'appliquent à la fois aux membres directs et indirects : lorsque le groupe d'hôtes B est membre du groupe d'hôtes A, tous les membres du groupe d'hôtes B sont considérés comme des membres indirects du groupe d'hôtes A.

41.2. VISUALISATION DES GROUPES D'HÔTES IDM À L'AIDE DE LA CLI

Cette section décrit comment visualiser les groupes d'hôtes IdM à l'aide de l'interface de ligne de commande (CLI).

Conditions préalables

- Privilèges d'administrateur pour la gestion de l'IdM ou rôle d'administrateur des utilisateurs.
- Un ticket Kerberos actif. Pour plus de détails, voir Utilisation de kinit pour se connecter manuellement à IdM.

Procédure

1. Recherchez tous les groupes d'hôtes à l'aide de la commande **ipa hostgroup-find**.

```
$ ipa hostgroup-find
------------------
1 hostgroup matched
------------------
  Host-group: ipaservers
  Description: IPA server hosts
----------------------------
Number of entries returned 1
----------------------------
```

Pour afficher tous les attributs d'un groupe d'hôtes, ajoutez l'option **--all**. Par exemple :

```
$ ipa hostgroup-find --all
------------------
1 hostgroup matched
------------------
  dn: cn=ipaservers,cn=hostgroups,cn=accounts,dc=idm,dc=local
  Host-group: ipaservers
  Description: IPA server hosts
  Member hosts: xxx.xxx.xxx.xxx
  ipauniqueid: xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
  objectclass: top, groupOfNames, nestedGroup, ipaobject, ipahostgroup
----------------------------
Number of entries returned 1
----------------------------
```

41.3. CRÉATION DE GROUPES D'HÔTES IDM À L'AIDE DE L'INTERFACE DE PROGRAMMATION

Cette section décrit comment créer des groupes d'hôtes IdM à l'aide de l'interface de ligne de commande (CLI).

Conditions préalables

- Privilèges d'administrateur pour la gestion de l'IdM ou rôle d'administrateur des utilisateurs.
- Un ticket Kerberos actif. Pour plus de détails, voir Utilisation de kinit pour se connecter manuellement à IdM.

Procédure

1. Ajoutez un groupe d'hôtes à l'aide de la commande **ipa hostgroup-add**.
Par exemple, pour créer un groupe d'hôtes IdM nommé *group_name* et lui donner une description :

```
$ ipa hostgroup-add --desc 'My new host group' group_name
---------------------
Added hostgroup "group_name"
---------------------
  Host-group: group_name
  Description: My new host group
---------------------
```

41.4. SUPPRESSION DES GROUPES D'HÔTES IDM À L'AIDE DE LA CLI

Cette section décrit comment supprimer les groupes d'hôtes IdM à l'aide de l'interface de ligne de commande (CLI).

Conditions préalables

- Privilèges d'administrateur pour la gestion de l'IdM ou rôle d'administrateur des utilisateurs.
- Un ticket Kerberos actif. Pour plus de détails, voir Utilisation de kinit pour se connecter manuellement à IdM.

Procédure

1. Supprimez un groupe d'hôtes à l'aide de la commande **ipa hostgroup-del**.
   Par exemple, pour supprimer le groupe d'hôtes IdM nommé *group_name*:

```
$ ipa hostgroup-del group_name
--------------------------
Deleted hostgroup "group_name"
--------------------------
```

NOTE

La suppression d'un groupe ne supprime pas les membres du groupe de l'IdM.

41.5. AJOUT DE MEMBRES DE GROUPES D'HÔTES IDM À L'AIDE DE LA CLI

Vous pouvez ajouter des hôtes et des groupes d'hôtes en tant que membres d'un groupe d'hôtes IdM à l'aide d'une seule commande.

Conditions préalables

- Privilèges d'administrateur pour la gestion de l'IdM ou rôle d'administrateur des utilisateurs.
- Un ticket Kerberos actif. Pour plus de détails, voir Utilisation de kinit pour se connecter manuellement à IdM.
- *Optional*. Utilisez la commande **ipa hostgroup-find** pour trouver les hôtes et les groupes d'hôtes.

Procédure

1. Pour ajouter un membre à un groupe d'hôtes, utilisez le site **ipa hostgroup-add-member** et fournissez les informations nécessaires. Vous pouvez spécifier le type de membre à ajouter à l'aide de ces options :

   - Utilisez l'option **--hosts** pour ajouter un ou plusieurs hôtes à un groupe d'hôtes IdM.
     Par exemple, pour ajouter l'hôte nommé *example_member* au groupe nommé *group_name*:

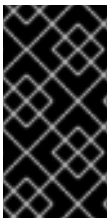
     ```
     $ ipa hostgroup-add-member group_name --hosts example_member
     Host-group: group_name
     Description: My host group
     Member hosts: example_member
     -------------------------
     Number of members added 1
     -------------------------
     ```

   - Utilisez l'option **--hostgroups** pour ajouter un ou plusieurs groupes d'hôtes à un groupe d'hôtes IdM.
     Par exemple, pour ajouter le groupe d'hôtes nommé *nested_group* au groupe nommé *group_name*:

     ```
     $ ipa hostgroup-add-member group_name --hostgroups nested_group
     Host-group: group_name
     Description: My host group
     Member host-groups: nested_group
     -------------------------
     Number of members added 1
     -------------------------
     ```

   - Vous pouvez ajouter plusieurs hôtes et plusieurs groupes d'hôtes à un groupe d'hôtes IdM en une seule commande en utilisant la syntaxe suivante :

     ```
     $ ipa hostgroup-add-member group_name --hosts={host1,host2} --hostgroups=
     {group1,group2}
     ```



IMPORTANT

Lorsque vous ajoutez un groupe d'hôtes en tant que membre d'un autre groupe d'hôtes, ne créez pas de groupes récursifs. Par exemple, si le groupe A est membre du groupe B, n'ajoutez pas le groupe B comme membre du groupe A. Les groupes récursifs peuvent entraîner un comportement imprévisible.

41.6. SUPPRESSION DES MEMBRES DU GROUPE D'HÔTES IDM À L'AIDE DE LA CLI

Vous pouvez supprimer des hôtes ainsi que des groupes d'hôtes d'un groupe d'hôtes IdM à l'aide d'une seule commande.

Conditions préalables

- Privilèges d'administrateur pour la gestion de l'IdM ou rôle d'administrateur des utilisateurs.
- Un ticket Kerberos actif. Pour plus de détails, voir Utilisation de kinit pour se connecter manuellement à IdM.

- *Optional*. Utilisez la commande **ipa hostgroup-find** pour confirmer que le groupe inclut le membre que vous souhaitez supprimer.

Procédure

1. Pour supprimer un membre d'un groupe d'hôtes, utilisez la commande **ipa hostgroup-remove-member** et fournissez les informations nécessaires. Vous pouvez spécifier le type de membre à supprimer à l'aide des options suivantes :

   - Utilisez l'option **--hosts** pour supprimer un ou plusieurs hôtes d'un groupe d'hôtes IdM.
     Par exemple, pour supprimer l'hôte nommé *example_member* du groupe nommé *group_name*:

     ```
     $ ipa hostgroup-remove-member group_name --hosts example_member
     Host-group: group_name
     Description: My host group
     -------------------------
     Number of members removed 1
     -------------------------
     ```

   - Utilisez l'option **--hostgroups** pour supprimer un ou plusieurs groupes d'hôtes d'un groupe d'hôtes IdM.
     Par exemple, pour retirer le groupe d'hôtes nommé *nested_group* du groupe nommé *group_name*:

     ```
     $ ipa hostgroup-remove-member group_name --hostgroups example_member
     Host-group: group_name
     Description: My host group
     -------------------------
     Number of members removed 1
     -------------------------
     ```

NOTE

La suppression d'un groupe ne supprime pas les membres du groupe de l'IdM.

- Vous pouvez supprimer plusieurs hôtes et plusieurs groupes d'hôtes d'un groupe d'hôtes IdM en une seule commande en utilisant la syntaxe suivante :

  ```
  $ ipa hostgroup-remove-member group_name --hosts={host1,host2} --hostgroups={group1,group2}
  ```

41.7. AJOUT DE GESTIONNAIRES MEMBRES DE GROUPES D'HÔTES IDM À L'AIDE DE LA CLI

Vous pouvez ajouter des hôtes ainsi que des groupes d'hôtes en tant que gestionnaires membres à un groupe d'hôtes IdM à l'aide d'une seule commande. Les gestionnaires membres peuvent ajouter des hôtes ou des groupes d'hôtes à des groupes d'hôtes IdM mais ne peuvent pas modifier les attributs d'un groupe d'hôtes.

Conditions préalables

- Privilèges d'administrateur pour la gestion de l'IdM ou rôle d'administrateur des utilisateurs.

- Un ticket Kerberos actif. Pour plus de détails, voir Utilisation de kinit pour se connecter manuellement à IdM.
- Vous devez avoir le nom de l'hôte ou du groupe d'hôtes que vous ajoutez en tant que membres managers et le nom du groupe d'hôtes que vous voulez qu'ils gèrent.

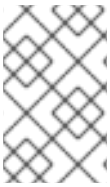
Procédure

1. *Optional*. Utilisez la commande **ipa hostgroup-find** pour trouver les hôtes et les groupes d'hôtes.
2. Pour ajouter un gestionnaire membre à un groupe d'hôtes, utilisez la commande **ipa hostgroup-add-member-manager**.
   Par exemple, pour ajouter l'utilisateur nommé *example_member* en tant que membre manager au groupe nommé *group_name*:

```
$ ipa hostgroup-add-member-manager group_name --user example_member
Host-group: group_name
Member hosts: server.idm.example.com
Member host-groups: project_admins
Member of netgroups: group_name
Membership managed by users: example_member
-------------------------
Number of members added 1
-------------------------
```

3. Utilisez l'option **--groups** pour ajouter un ou plusieurs groupes d'hôtes en tant que gestionnaire membre d'un groupe d'hôtes IdM.
   Par exemple, pour ajouter le groupe d'hôtes nommé *admin_group* en tant que gestionnaire membre du groupe nommé *group_name*:

```
$ ipa hostgroup-add-member-manager group_name --groups admin_group
Host-group: group_name
Member hosts: server.idm.example.com
Member host-groups: project_admins
Member of netgroups: group_name
Membership managed by groups: admin_group
Membership managed by users: example_member
-------------------------
Number of members added 1
-------------------------
```



NOTE

Après avoir ajouté un gestionnaire membre à un groupe d'hôtes, la mise à jour peut prendre un certain temps avant de se propager à tous les clients de votre environnement de gestion des identités.

Verification steps

- Utiliser la commande **ipa group-show** pour vérifier que l'utilisateur et le groupe d'hôtes ont été ajoutés en tant que gestionnaires membres.

```
$ ipa hostgroup-show group_name
```

```
Host-group: group_name
Member hosts: server.idm.example.com
Member host-groups: project_admins
Membership managed by groups: admin_group
Membership managed by users: example_member
```

Ressources supplémentaires

- Voir **ipa hostgroup-add-member-manager --help** pour plus de détails.
- Voir **ipa hostgroup-show --help** pour plus de détails.

41.8. SUPPRESSION DES GESTIONNAIRES MEMBRES DU GROUPE D'HÔTES IDM À L'AIDE DE LA CLI

Vous pouvez supprimer des hôtes ainsi que des groupes d'hôtes en tant que gestionnaires membres d'un groupe d'hôtes IdM à l'aide d'une seule commande. Les gestionnaires membres peuvent supprimer des groupes d'hôtes des groupes d'hôtes IdM mais ne peuvent pas modifier les attributs d'un groupe d'hôtes.

Conditions préalables

- Privilèges d'administrateur pour la gestion de l'IdM ou rôle d'administrateur des utilisateurs.
- Un ticket Kerberos actif. Pour plus de détails, voir Utilisation de kinit pour se connecter manuellement à IdM.
- Vous devez disposer du nom du groupe d'hôtes du gestionnaire existant que vous supprimez et du nom du groupe d'hôtes qu'il gère.

Procédure

1. *Optional*. Utilisez la commande **ipa hostgroup-find** pour trouver les hôtes et les groupes d'hôtes.
2. Pour supprimer un gestionnaire membre d'un groupe d'hôtes, utilisez la commande **ipa hostgroup-remove-member-manager**.
   Par exemple, pour supprimer l'utilisateur nommé *example_member* en tant que membre manager du groupe nommé *group_name*:

   ```
   $ ipa hostgroup-remove-member-manager group_name --user example_member
   Host-group: group_name
   Member hosts: server.idm.example.com
   Member host-groups: project_admins
   Member of netgroups: group_name
   Membership managed by groups: nested_group
   -------------------------
   Number of members removed 1
   -------------------------
   ```

3. Utilisez l'option **--groups** pour supprimer un ou plusieurs groupes d'hôtes en tant que gestionnaire membre d'un groupe d'hôtes IdM.
   Par exemple, pour supprimer le groupe d'hôtes nommé *nested_group* en tant que gestionnaire membre du groupe nommé *group_name*:

```
$ ipa hostgroup-remove-member-manager group_name --groups nested_group
Host-group: group_name
Member hosts: server.idm.example.com
Member host-groups: project_admins
Member of netgroups: group_name
-------------------------
Number of members removed 1
-------------------------
```

NOTE

Après avoir supprimé un gestionnaire membre d'un groupe hôte, la mise à jour peut prendre un certain temps avant de se propager à tous les clients de votre environnement de gestion des identités.

Verification steps

- Utilisez la commande **ipa group-show** pour vérifier que l'utilisateur et le groupe d'hôtes ont été supprimés en tant que gestionnaires membres.

```
$ ipa hostgroup-show group_name
Host-group: group_name
Member hosts: server.idm.example.com
Member host-groups: project_admins
```

Ressources supplémentaires

- Voir **ipa hostgroup-remove-member-manager --help** pour plus de détails.
- Voir **ipa hostgroup-show --help** pour plus de détails.

CHAPITRE 42. GESTION DES GROUPES D'HÔTES À L'AIDE DE L'INTERFACE WEB IDM

Ce chapitre présente les groupes d'hôtes dans la gestion des identités (IdM) et décrit les opérations suivantes pour gérer les groupes d'hôtes et leurs membres dans l'interface Web (Web UI) :

- Visualisation des groupes d'accueil et de leurs membres
- Création de groupes d'accueil
- Suppression de groupes d'hôtes
- Ajouter des membres à un groupe d'hôtes
- Suppression de membres d'un groupe d'hôtes
- Ajout de gestionnaires de groupes d'hôtes
- Suppression des gestionnaires de groupes d'hôtes

42.1. GROUPES D'ACCUEIL DANS L'IDM

Les groupes d'hôtes IdM peuvent être utilisés pour centraliser le contrôle des tâches de gestion importantes, en particulier le contrôle d'accès.

Définition des groupes d'accueil

Un groupe d'hôtes est une entité qui contient un ensemble d'hôtes IdM avec des règles de contrôle d'accès communes et d'autres caractéristiques. Par exemple, vous pouvez définir des groupes d'hôtes en fonction des départements de l'entreprise, des emplacements physiques ou des exigences en matière de contrôle d'accès.

Un groupe d'hôtes dans IdM peut comprendre

- Serveurs et clients IdM
- Autres groupes d'accueil IdM

Groupes d'hôtes créés par défaut

Par défaut, le serveur IdM crée le groupe d'hôtes **ipaservers** pour tous les hôtes du serveur IdM.

Membres directs et indirects du groupe

Les attributs de groupe dans IdM s'appliquent à la fois aux membres directs et indirects : lorsque le groupe d'hôtes B est membre du groupe d'hôtes A, tous les membres du groupe d'hôtes B sont considérés comme des membres indirects du groupe d'hôtes A.

42.2. VISUALISATION DES GROUPES D'HÔTES DANS L'INTERFACE WEB IDM

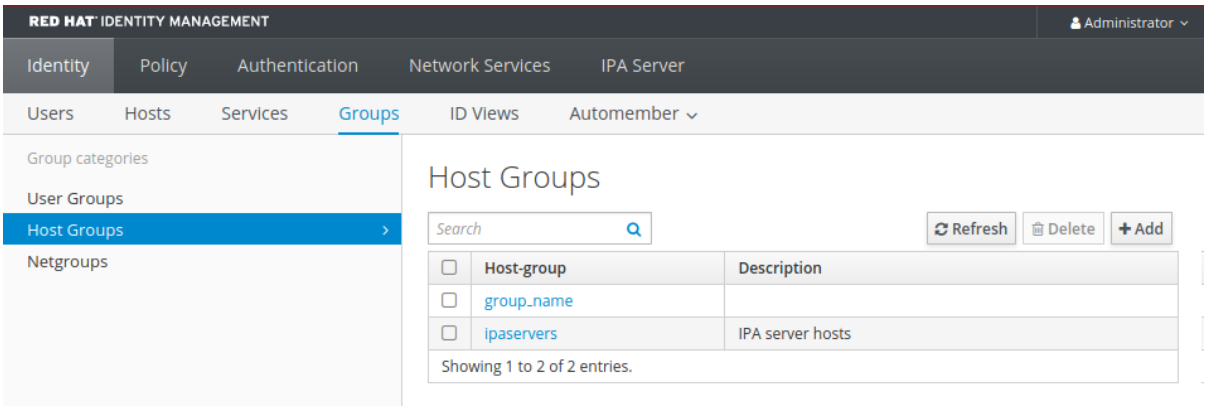
Cette section décrit comment visualiser les groupes d'hôtes IdM à l'aide de l'interface Web (Web UI).

Conditions préalables

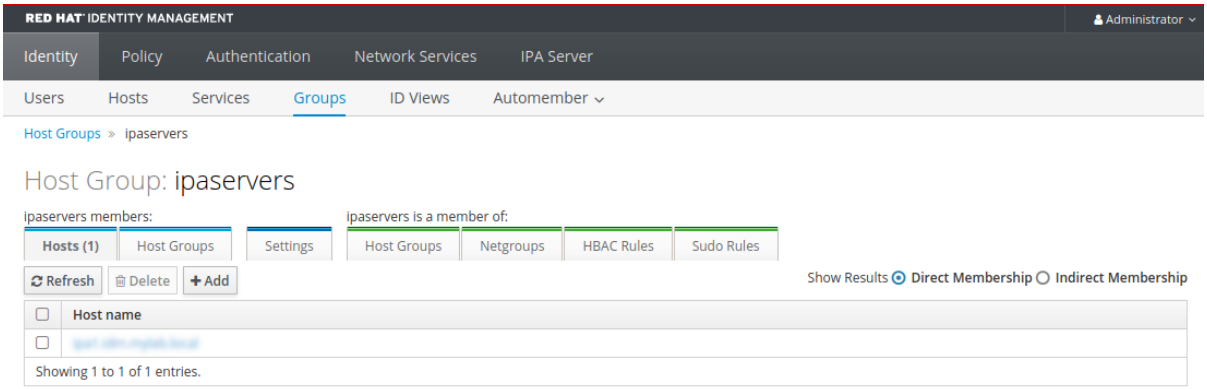
- Privilèges d'administrateur pour la gestion de l'IdM ou rôle d'administrateur des utilisateurs.
- Vous êtes connecté à l'interface Web IdM. Pour plus de détails, voir Accès à l'IdM Web UI dans un navigateur web.

Procédure

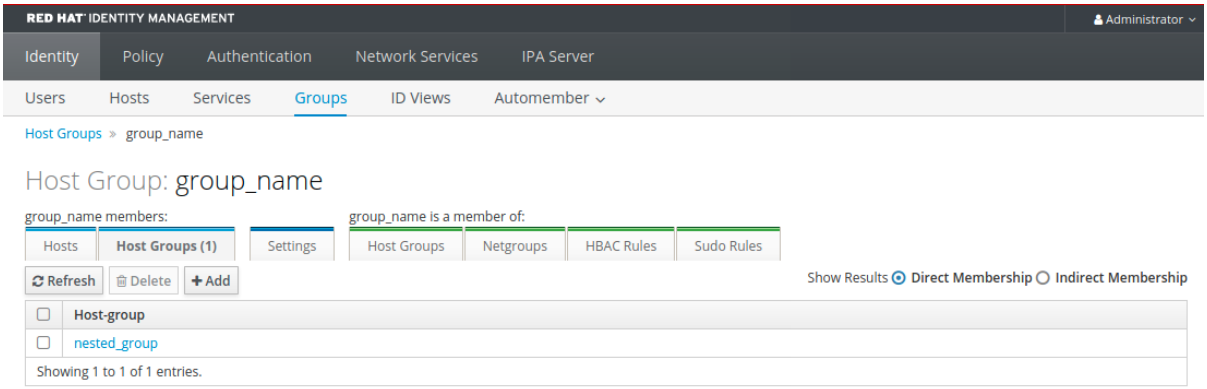
1. Cliquez sur **Identity → Groups** et sélectionnez l'onglet **Host Groups**.
   - La page répertorie les groupes d'hôtes existants et leur description.
   - Vous pouvez rechercher un groupe d'hôtes spécifique.



2. Cliquez sur un groupe dans la liste pour afficher les hôtes qui appartiennent à ce groupe. Vous pouvez limiter les résultats aux membres directs ou indirects.



3. Sélectionnez l'onglet **Host Groups** pour afficher les groupes d'hôtes qui appartiennent à ce groupe (groupes d'hôtes imbriqués). Vous pouvez limiter les résultats aux membres directs ou indirects.



42.3. CRÉATION DE GROUPES D'HÔTES DANS L'INTERFACE WEB IDM

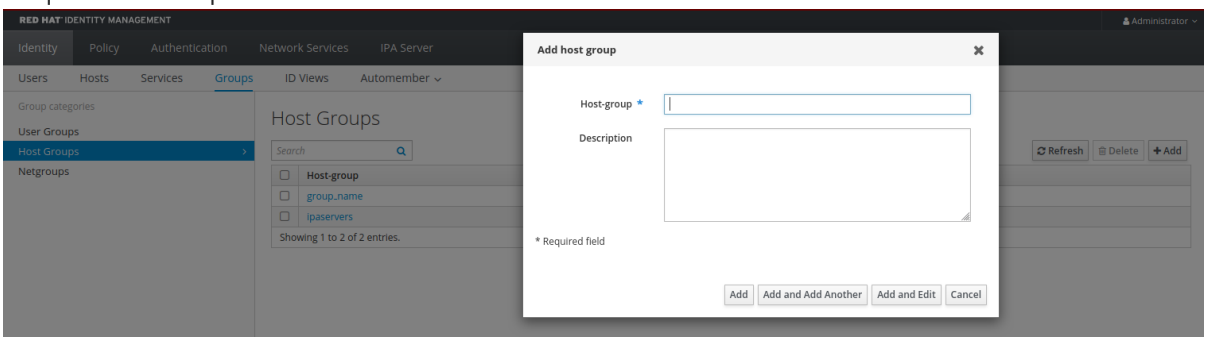
Cette section décrit comment créer des groupes d'hôtes IdM à l'aide de l'interface Web (Web UI).

Conditions préalables

- Privilèges d'administrateur pour la gestion de l'IdM ou rôle d'administrateur des utilisateurs.
- Vous êtes connecté à l'interface Web IdM. Pour plus de détails, voir Accès à l'IdM Web UI dans un navigateur web.

Procédure

1. Cliquez sur **Identity → Groups** et sélectionnez l'onglet **Host Groups**.
2. Cliquez sur **Add**. La boîte de dialogue **Add host group** apparaît.
3. Fournir les informations sur le groupe : nom (obligatoire) et description (facultative).
4. Cliquez sur **Add** pour confirmer.



42.4. SUPPRESSION DE GROUPES D'HÔTES DANS L'INTERFACE WEB IDM

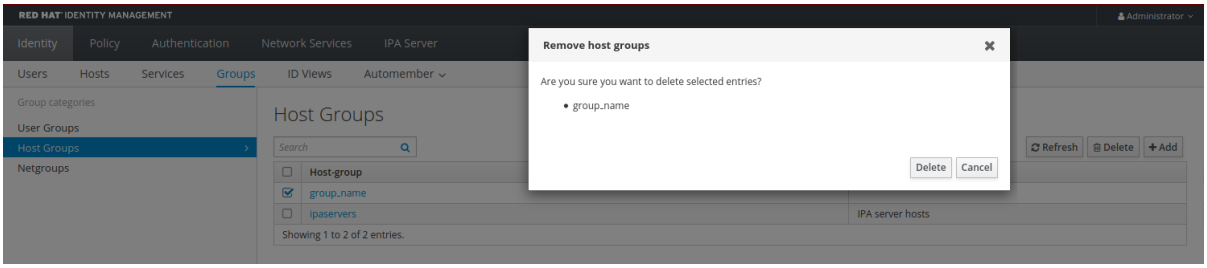
Cette section décrit comment supprimer les groupes d'hôtes IdM à l'aide de l'interface Web (Web UI).

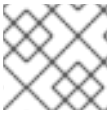
Conditions préalables

- Privilèges d'administrateur pour la gestion de l'IdM ou rôle d'administrateur des utilisateurs.
- Vous êtes connecté à l'interface Web IdM. Pour plus de détails, voir Accès à l'IdM Web UI dans un navigateur web.

Procédure

1. Cliquez sur **Identity → Groups** et sélectionnez l'onglet **Host Groups**.
2. Sélectionnez le groupe d'hôtes IdM à supprimer et cliquez sur **Delete**. Une boîte de dialogue de confirmation apparaît.
3. Cliquez sur **Delete** pour confirmer





NOTE

La suppression d'un groupe d'hôtes ne supprime pas les membres du groupe de l'IdM.

42.5. AJOUT DE MEMBRES DE GROUPES D'HÔTES DANS L'INTERFACE WEB IDM

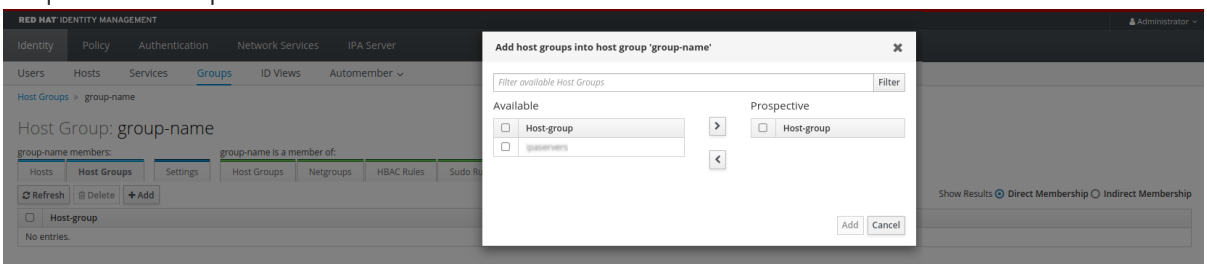
Cette section décrit comment ajouter des membres de groupes d'hôtes dans IdM en utilisant l'interface web (Web UI).

Conditions préalables

- Privilèges d'administrateur pour la gestion de l'IdM ou rôle d'administrateur des utilisateurs.
- Vous êtes connecté à l'interface Web IdM. Pour plus de détails, voir Accès à l'IdM Web UI dans un navigateur web.

Procédure

1. Cliquez sur **Identity → Groups** et sélectionnez l'onglet **Host Groups**.
2. Cliquez sur le nom du groupe auquel vous souhaitez ajouter des membres.
3. Cliquez sur l'onglet **Hosts** ou **Host groups** selon le type de membres que vous souhaitez ajouter. La boîte de dialogue correspondante apparaît.
4. Sélectionnez les hôtes ou les groupes d'hôtes à ajouter, et cliquez sur la flèche > pour les déplacer dans la colonne **Prospective**.
5. Cliquez sur **Add** pour confirmer.



42.6. SUPPRESSION DES MEMBRES D'UN GROUPE D'HÔTES DANS L'INTERFACE WEB IDM

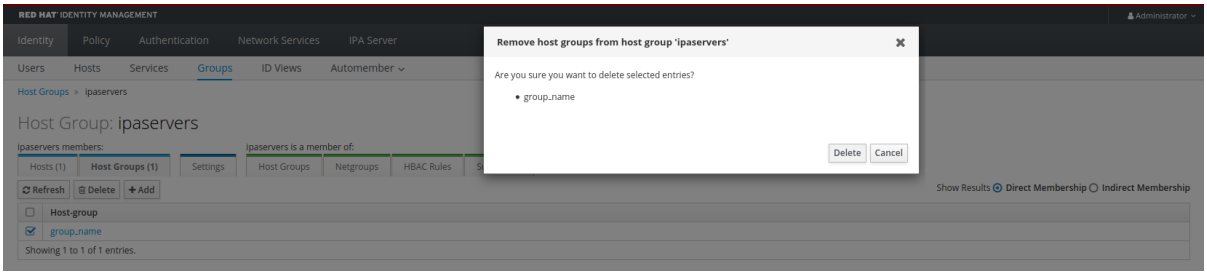
Cette section décrit comment supprimer des membres d'un groupe d'hôtes dans IdM à l'aide de l'interface web (Web UI).

Conditions préalables

- Privilèges d'administrateur pour la gestion de l'IdM ou rôle d'administrateur des utilisateurs.
- Vous êtes connecté à l'interface Web IdM. Pour plus de détails, voir Accès à l'IdM Web UI dans un navigateur web.

Procédure

1. Cliquez sur **Identity → Groups** et sélectionnez l'onglet **Host Groups**.
2. Cliquez sur le nom du groupe dont vous souhaitez retirer des membres.
3. Cliquez sur l'onglet **Hosts** ou **Host groups** en fonction du type de membres que vous souhaitez supprimer.
4. Cochez la case en regard du membre que vous souhaitez supprimer.
5. Cliquez sur Supprimer. Une boîte de dialogue de confirmation apparaît.



6. Cliquez sur Supprimer pour confirmer. Les membres sélectionnés sont supprimés.

42.7. AJOUT DE GESTIONNAIRES MEMBRES DE GROUPES D'HÔTES IDM À L'AIDE DE L'INTERFACE WEB

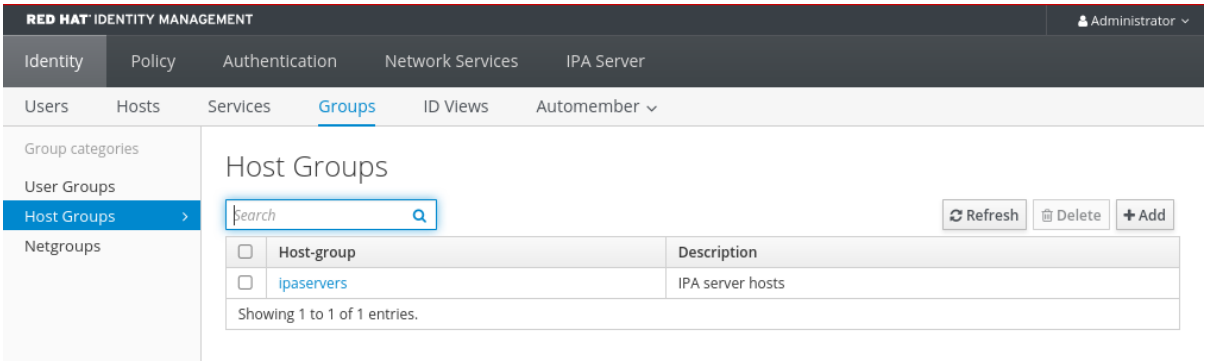
Cette section décrit comment ajouter des utilisateurs ou des groupes d'utilisateurs en tant que gestionnaires membres de groupes d'hôtes dans IdM en utilisant l'interface web (Web UI). Les gestionnaires de membres peuvent ajouter des gestionnaires de membres de groupes d'hôtes aux groupes d'hôtes IdM mais ne peuvent pas modifier les attributs d'un groupe d'hôtes.

Conditions préalables

- Privilèges d'administrateur pour la gestion de l'IdM ou rôle d'administrateur des utilisateurs.
- Vous êtes connecté à l'interface Web IdM. Pour plus de détails, voir Accès à l'IdM Web UI dans un navigateur web.
- Vous devez avoir le nom du groupe d'hôtes que vous ajoutez en tant que membres managers et le nom du groupe d'hôtes que vous voulez qu'ils gèrent.

Procédure

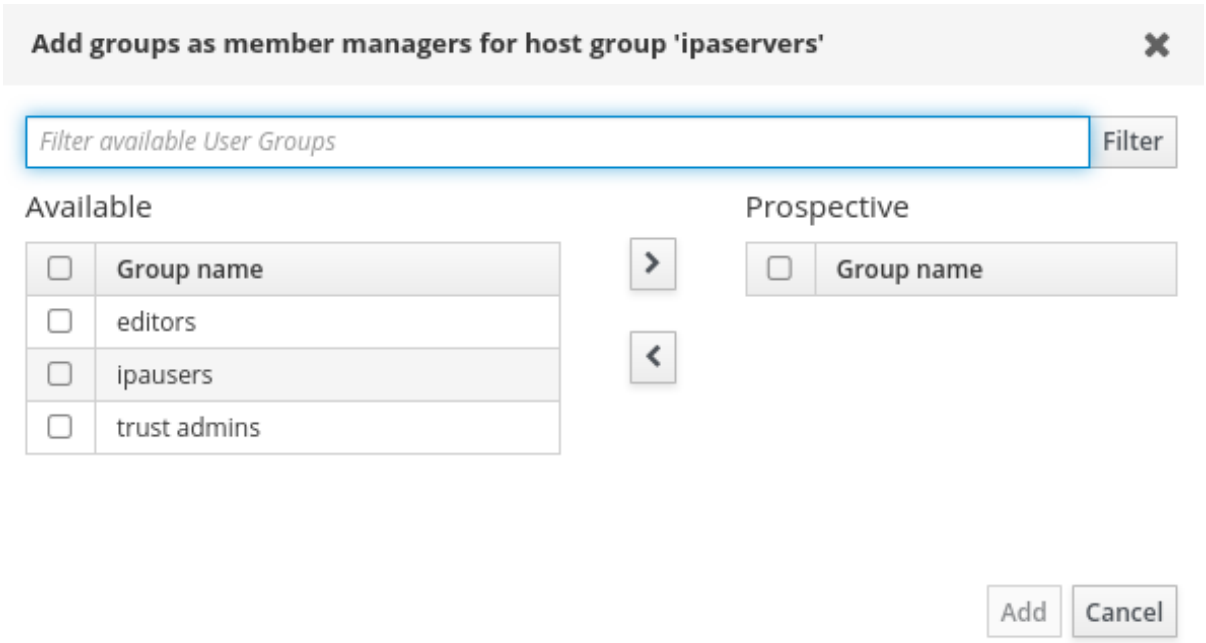
1. Cliquez sur **Identity → Groups** et sélectionnez l'onglet **Host Groups**.



2. Cliquez sur le nom du groupe auquel vous souhaitez ajouter des gestionnaires membres.

3. Cliquez sur l'onglet gestionnaires de membres **User Groups** ou **Users** selon le type de gestionnaires de membres que vous souhaitez ajouter. La boîte de dialogue correspondante apparaît.

4. Cliquez sur **Add**.



5. Sélectionnez les utilisateurs ou les groupes d'utilisateurs à ajouter et cliquez sur la flèche > pour les déplacer dans la colonne **Prospective**.

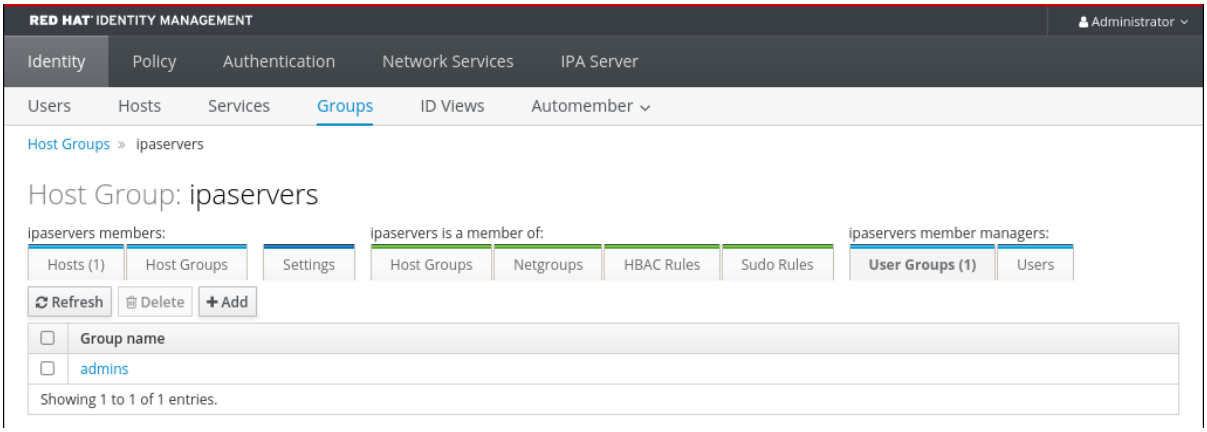
6. Cliquez sur **Add** pour confirmer.

NOTE

Après avoir ajouté un gestionnaire membre à un groupe d'hôtes, la mise à jour peut prendre un certain temps avant de se propager à tous les clients de votre environnement de gestion des identités.

Verification steps

- Dans la boîte de dialogue Groupe hôte, vérifiez que le groupe d'utilisateurs ou l'utilisateur a été ajouté à la liste des groupes ou des utilisateurs des gestionnaires membres.



42.8. SUPPRESSION DES GESTIONNAIRES MEMBRES DU GROUPE D'HÔTES IDM À L'AIDE DE L'INTERFACE WEB

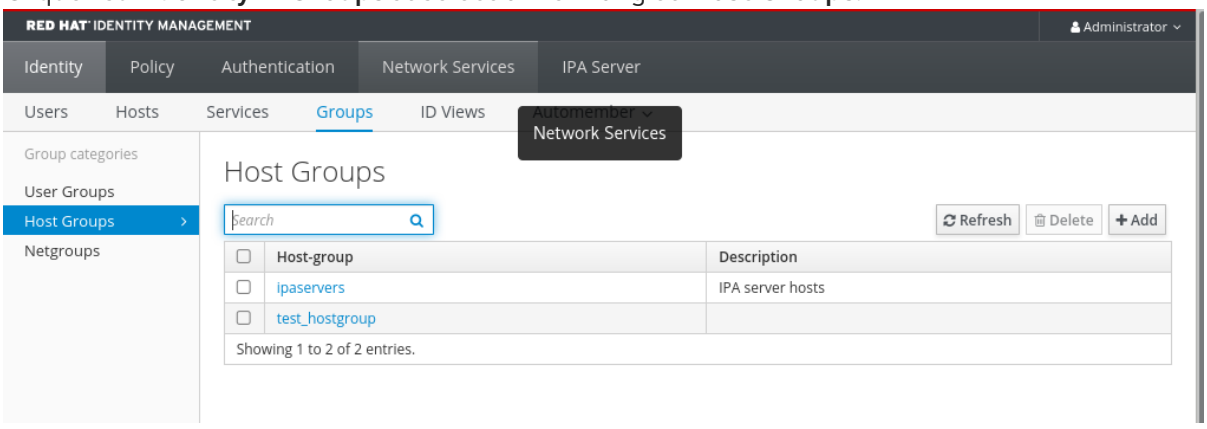
Cette section décrit comment supprimer des utilisateurs ou des groupes d'utilisateurs en tant que gestionnaires de groupes d'hôtes dans IdM à l'aide de l'interface web (Web UI). Les gestionnaires de membres peuvent supprimer des gestionnaires de membres de groupes d'hôtes d'IdM, mais ne peuvent pas modifier les attributs d'un groupe d'hôtes.

Conditions préalables

- Privilèges d'administrateur pour la gestion de l'IdM ou rôle d'administrateur des utilisateurs.
- Vous êtes connecté à l'interface Web IdM. Pour plus de détails, voir Accès à l'IdM Web UI dans un navigateur web.
- Vous devez disposer du nom du groupe d'hôtes du gestionnaire existant que vous supprimez et du nom du groupe d'hôtes qu'il gère.

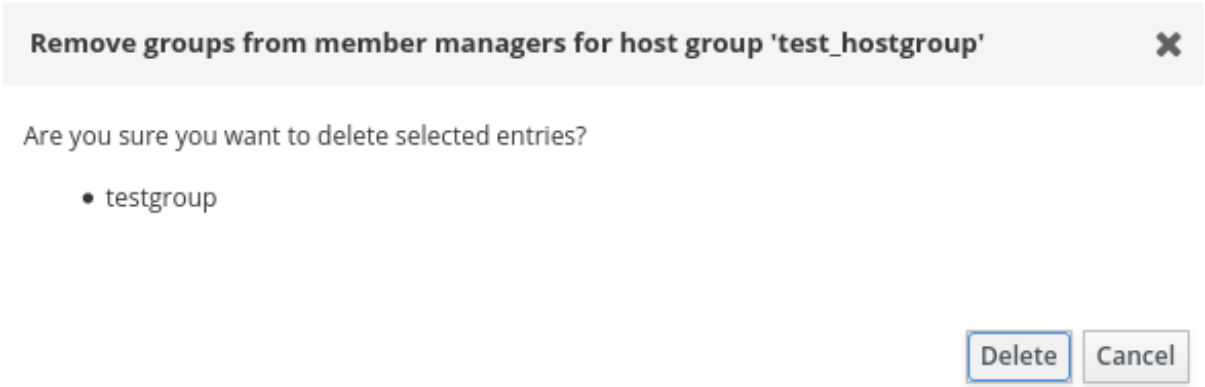
Procédure

1. Cliquez sur **Identity → Groups** et sélectionnez l'onglet **Host Groups**.



2. Cliquez sur le nom du groupe dont vous souhaitez supprimer les gestionnaires membres.
3. Cliquez sur l'onglet des gestionnaires de membres **User Groups** ou **Users** selon le type de gestionnaires de membres que vous souhaitez supprimer. La boîte de dialogue correspondante apparaît.
4. Sélectionnez l'utilisateur ou les groupes d'utilisateurs à supprimer et cliquez sur **Delete**.

5. Cliquez sur **Delete** pour confirmer.

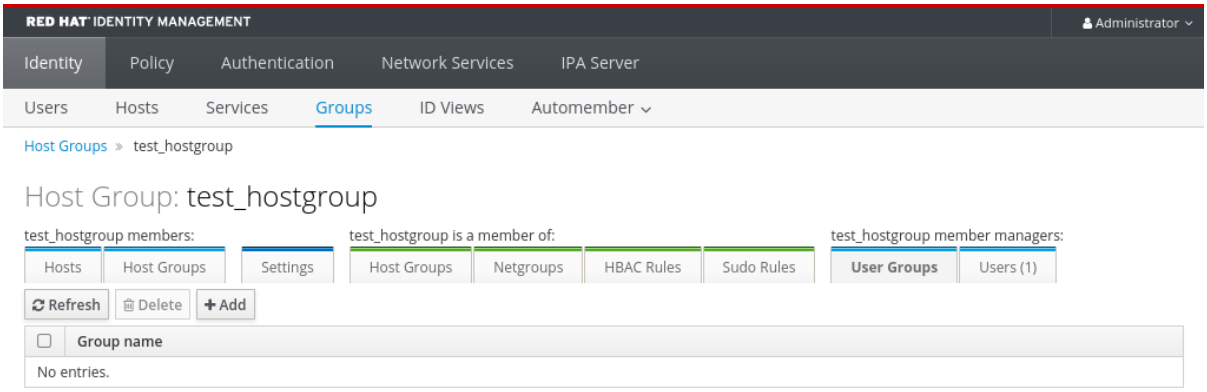


NOTE

Après avoir supprimé un gestionnaire membre d'un groupe hôte, la mise à jour peut prendre un certain temps avant de se propager à tous les clients de votre environnement de gestion des identités.

Verification steps

- Dans la boîte de dialogue Groupe hôte, vérifiez que le groupe d'utilisateurs ou l'utilisateur a été supprimé de la liste des groupes ou des utilisateurs des gestionnaires membres.



CHAPITRE 43. GÉRER LES GROUPES D'HÔTES À L'AIDE DES PLAYBOOKS ANSIBLE

Ce chapitre présente les groupes d'hôtes dans la gestion des identités (IdM) et décrit l'utilisation d'Ansible pour effectuer les opérations suivantes impliquant des groupes d'hôtes dans la gestion des identités (IdM) :

- Groupes d'accueil dans l'IdM
- Assurer la présence de groupes d'hôtes IdM
- Assurer la présence d'hôtes dans les groupes d'hôtes IdM
- Imbrication des groupes d'hôtes IdM
- Assurer la présence des gestionnaires membres dans les groupes d'accueil de l'IdM
- Garantir l'absence d'hôtes dans les groupes d'hôtes IdM
- Garantir l'absence de groupes d'hôtes imbriqués dans les groupes d'hôtes IdM
- S'assurer de l'absence de membres gestionnaires dans les groupes d'accueil de l'IdM

43.1. GROUPES D'ACCUEIL DANS L'IDM

Les groupes d'hôtes IdM peuvent être utilisés pour centraliser le contrôle des tâches de gestion importantes, en particulier le contrôle d'accès.

Définition des groupes d'accueil

Un groupe d'hôtes est une entité qui contient un ensemble d'hôtes IdM avec des règles de contrôle d'accès communes et d'autres caractéristiques. Par exemple, vous pouvez définir des groupes d'hôtes en fonction des départements de l'entreprise, des emplacements physiques ou des exigences en matière de contrôle d'accès.

Un groupe d'hôtes dans IdM peut comprendre

- Serveurs et clients IdM
- Autres groupes d'accueil IdM

Groupes d'hôtes créés par défaut

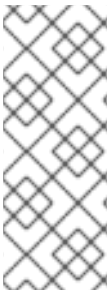
Par défaut, le serveur IdM crée le groupe d'hôtes **ipaservers** pour tous les hôtes du serveur IdM.

Membres directs et indirects du groupe

Les attributs de groupe dans IdM s'appliquent à la fois aux membres directs et indirects : lorsque le groupe d'hôtes B est membre du groupe d'hôtes A, tous les membres du groupe d'hôtes B sont considérés comme des membres indirects du groupe d'hôtes A.

43.2. ASSURER LA PRÉSENCE DES GROUPES D'HÔTES IDM À L'AIDE DES PLAYBOOKS ANSIBLE

Cette section décrit comment assurer la présence de groupes d'hôtes dans la gestion des identités (IdM) à l'aide des playbooks Ansible.



NOTE

Sans Ansible, les entrées de groupes d'hôtes sont créées dans IdM à l'aide de la commande **ipa hostgroup-add**. Le résultat de l'ajout d'un groupe d'hôtes à IdM est l'état du groupe d'hôtes présent dans IdM. En raison de la dépendance d'Ansible à l'égard de l'idempotence, pour ajouter un groupe d'hôtes à IdM à l'aide d'Ansible, vous devez créer un playbook dans lequel vous définissez l'état du groupe d'hôtes comme étant présent : **state: present**.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

Procédure

1. Créez un fichier d'inventaire, par exemple **inventory.file**, et définissez-y **ipaserver** avec la liste des serveurs IdM à cibler :

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

2. Créez un fichier playbook Ansible avec les informations nécessaires sur le groupe d'hôtes. Par exemple, pour garantir la présence d'un groupe d'hôtes nommé **databases**, spécifiez **name: databases** dans la tâche **- ipahostgroup**. Pour simplifier cette étape, vous pouvez copier et modifier l'exemple dans le fichier **/usr/share/doc/ansible-freeipa/playbooks/user/ensure-hostgroup-is-present.yml**.

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

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  # Ensure host-group databases is present
  - ipahostgroup:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: databases
      state: present
```

Dans le playbook, **state: present** signifie une demande d'ajout du groupe d'hôtes à IdM, à moins qu'il n'y existe déjà.

3. Exécutez le manuel de jeu :

```
$ ansible-playbook --vault-password-file=password_file -v -i
path_to_inventory_directory/inventory.file path_to_playbooks_directory/ensure-
hostgroup-is-present.yml
```

Verification steps

1. Connectez-vous à **ipaserver** en tant qu'administrateur :

```
$ ssh admin@server.idm.example.com
Password:
[admin@server /]$
```

2. Demander un ticket Kerberos pour l'administrateur :

```
$ kinit admin
Password for admin@IDM.EXAMPLE.COM:
```

3. Affichez les informations sur le groupe d'hôtes dont vous voulez assurer la présence dans l'IdM :

```
$ ipa hostgroup-show databases
  Host-group: databases
```

Le groupe d'hôtes **databases** existe dans IdM.

43.3. ASSURER LA PRÉSENCE D'HÔTES DANS LES GROUPES D'HÔTES IDM À L'AIDE DES PLAYBOOKS ANSIBLE

Cette section décrit comment assurer la présence des hôtes dans les groupes d'hôtes dans la gestion des identités (IdM) à l'aide des playbooks Ansible.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.
- Les hôtes que vous souhaitez référencer dans votre livre de programmation Ansible existent dans IdM. Pour plus de détails, voir Assurer la présence d'une entrée d'hôte IdM à l'aide des carnets de commande Ansible.

- Les groupes d'hôtes que vous avez référencés dans le fichier du livre de jeu Ansible ont été ajoutés à l'IdM. Pour plus de détails, voir Assurer la présence des groupes d'hôtes IdM à l'aide des playbooks Ansible.

Procédure

1. Créez un fichier d'inventaire, par exemple **inventory.file**, et définissez-y **ipaserver** avec la liste des serveurs IdM à cibler :

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

2. Créez un fichier playbook Ansible avec les informations nécessaires sur l'hôte. Spécifiez le nom du groupe d'hôtes à l'aide du paramètre **name** de la variable **ipahostgroup**. Spécifiez le nom de l'hôte à l'aide du paramètre **host** de la variable **ipahostgroup**. Pour simplifier cette étape, vous pouvez copier et modifier les exemples du fichier **/usr/share/doc/ansible-freeipa/playbooks/hostgroup/ensure-hosts-and-hostgroups-are-present-in-hostgroup.yml**:

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

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  # Ensure host-group databases is present
  - ipahostgroup:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: databases
      host:
      - db.idm.example.com
      action: member
```

Ce livre de jeu ajoute l'hôte **db.idm.example.com** au groupe d'hôtes **databases**. La ligne **action: member** indique que lors de l'exécution de la séquence, aucune tentative n'est faite pour ajouter le groupe **databases** lui-même. Au lieu de cela, seule une tentative d'ajout de **db.idm.example.com** à **databases** est effectuée.

3. Exécutez le manuel de jeu :

```
$ ansible-playbook --vault-password-file=password_file -v -i
path_to_inventory_directory/inventory.file path_to_playbooks_directory/ensure-hosts-
or-hostgroups-are-present-in-hostgroup.yml
```

Verification steps

1. Connectez-vous à **ipaserver** en tant qu'administrateur :

```
$ ssh admin@server.idm.example.com
Password:
[admin@server /]$
```

2. Demander un ticket Kerberos pour l'administrateur :

```
$ kinit admin
Password for admin@IDM.EXAMPLE.COM:
```

3. Afficher des informations sur un groupe d'hôtes pour savoir quels sont les hôtes présents dans ce groupe :

```
$ ipa hostgroup-show databases
  Host-group: databases
  Member hosts: db.idm.example.com
```

L'hôte **db.idm.example.com** est présent en tant que membre du groupe d'hôtes **databases**.

43.4. IMBRICATION DES GROUPES D'HÔTES IDM À L'AIDE DES PLAYBOOKS ANSIBLE

Cette section décrit comment assurer la présence de groupes d'hôtes imbriqués dans les groupes d'hôtes de gestion des identités (IdM) à l'aide des playbooks Ansible.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.
- Les groupes d'hôtes auxquels vous faites référence dans le fichier du livre de jeu Ansible existent dans IdM. Pour plus de détails, voir Assurer la présence des groupes d'hôtes IdM à l'aide des playbooks Ansible.

Procédure

1. Créez un fichier d'inventaire, par exemple **inventory.file**, et définissez-y **ipaserver** avec la liste des serveurs IdM à cibler :

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

2. Créez un fichier playbook Ansible avec les informations nécessaires sur les groupes d'hôtes. Pour s'assurer qu'un groupe d'hôtes imbriqué *A* existe dans un groupe d'hôtes *B*: dans le manuel de jeu Ansible, spécifiez, parmi les variables **- ipahostgroup**, le nom du groupe d'hôtes *B* à l'aide de la variable **name**. Spécifiez le nom du groupe d'hôtes imbriqué *A* avec la variable **hostgroup**. Pour simplifier cette étape, vous pouvez copier et modifier les exemples dans le fichier **/usr/share/doc/ansible-freeipa/playbooks/hostgroup/ensure-hosts-and-hostgroups-are-present-in-hostgroup.yml**:

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

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  # Ensure hosts and hostgroups are present in existing databases hostgroup
  - ipahostgroup:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: databases
      hostgroup:
      - mysql-server
      - oracle-server
      action: member
```

Ce playbook Ansible assure la présence des groupes d'hôtes **myqsl-server** et **oracle-server** dans le groupe d'hôtes **databases**. La ligne **action: member** indique que lorsque le playbook est exécuté, aucune tentative n'est faite pour ajouter le groupe **databases** lui-même à l'IdM.

3. Exécutez le manuel de jeu :

```
$ ansible-playbook --vault-password-file=password_file -v -i
path_to_inventory_directory/inventory.file path_to_playbooks_directory/ensure-hosts-
or-hostgroups-are-present-in-hostgroup.yml
```

Verification steps

1. Connectez-vous à **ipaserver** en tant qu'administrateur :

```
$ ssh admin@server.idm.example.com
Password:
[admin@server /]$
```

2. Demander un ticket Kerberos pour l'administrateur :

```
$ kinit admin
Password for admin@IDM.EXAMPLE.COM:
```

3. Affiche des informations sur le groupe d'hôtes dans lequel se trouvent des groupes d'hôtes imbriqués :

```
$ ipa hostgroup-show databases
  Host-group: databases
  Member hosts: db.idm.example.com
  Member host-groups: mysql-server, oracle-server
```

Les groupes d'hôtes **mysql-server** et **oracle-server** existent dans le groupe d'hôtes **databases**.

43.5. ASSURER LA PRÉSENCE DE GESTIONNAIRES MEMBRES DANS LES GROUPES D'HÔTES IDM À L'AIDE DES PLAYBOOKS ANSIBLE

La procédure suivante décrit comment assurer la présence des gestionnaires membres dans les hôtes IdM et les groupes d'hôtes à l'aide d'un livre de jeu Ansible.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.
- Vous devez avoir le nom de l'hôte ou du groupe d'hôtes que vous ajoutez en tant que membres managers et le nom du groupe d'hôtes que vous voulez qu'ils gèrent.

Procédure

1. Créez un fichier d'inventaire, par exemple **inventory.file**, et définissez-y **ipaserver**:

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

2. Créer un fichier Ansible playbook avec les informations nécessaires à la gestion des hôtes et des membres du groupe d'hôtes :

```
---

- name: Playbook to handle host group membership management
  hosts: ipaserver

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Ensure member manager user example_member is present for group_name
      ipahostgroup:
        ipaadmin_password: "{{ ipaadmin_password }}"
        name: group_name
        membermanager_user: example_member

  - name: Ensure member manager group project_admins is present for group_name
      ipahostgroup:
        ipaadmin_password: "{{ ipaadmin_password }}"
        name: group_name
        membermanager_group: project_admins
```

3. Exécutez le manuel de jeu :

```
$ ansible-playbook --vault-password-file=password_file -v -i
path_to_inventory_directory/inventory.file path_to_playbooks_directory/add-member-
managers-host-groups.yml
```

Verification steps

Vous pouvez vérifier si le groupe **group_name** contient **example_member** et **project_admins** en tant que gestionnaires membres en utilisant la commande **ipa group-show**:

1. Connectez-vous à **ipaserver** en tant qu'administrateur :

```
$ ssh admin@server.idm.example.com
Password:
[admin@server /]$
```

2. Afficher des informations sur *testhostgroup*:

```
ipaserver]$ ipa hostgroup-show group_name
  Host-group: group_name
  Member hosts: server.idm.example.com
  Member host-groups: testhostgroup2
  Membership managed by groups: project_admins
  Membership managed by users: example_member
```

Ressources supplémentaires

- Voir **ipa hostgroup-add-member-manager --help**.
- Voir la page de manuel **ipa**.

43.6. GARANTIR L'ABSENCE D'HÔTES DANS LES GROUPES D'HÔTES IDM À L'AIDE DES PLAYBOOKS ANSIBLE

Cette section décrit comment garantir l'absence d'hôtes dans les groupes d'hôtes de la gestion des identités (IdM) à l'aide des playbooks Ansible.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

- Les hôtes que vous souhaitez référencer dans votre livre de programmation Ansible existent dans IdM. Pour plus de détails, voir Assurer la présence d'une entrée d'hôte IdM à l'aide des carnets de commande Ansible.
- Les groupes d'hôtes auxquels vous faites référence dans le fichier du livre de jeu Ansible existent dans IdM. Pour plus de détails, voir Assurer la présence des groupes d'hôtes IdM à l'aide des playbooks Ansible.

Procédure

1. Créez un fichier d'inventaire, par exemple **inventory.file**, et définissez-y **ipaserver** avec la liste des serveurs IdM à cibler :

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

2. Créez un fichier playbook Ansible avec les informations nécessaires sur l'hôte et le groupe d'hôtes. Spécifiez le nom du groupe d'hôtes en utilisant le paramètre **name** de la variable **ipahostgroup**. Spécifiez le nom de l'hôte dont vous voulez garantir l'absence dans le groupe d'hôtes en utilisant le paramètre **host** de la variable **ipahostgroup**. Pour simplifier cette étape, vous pouvez copier et modifier les exemples du fichier **/usr/share/doc/ansible-freeipa/playbooks/hostgroup/ensure-hosts-and-hostgroups-are-absent-in-hostgroup.yml**:

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

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  # Ensure host-group databases is absent
  - ipahostgroup:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: databases
      host:
      - db.idm.example.com
      action: member
      state: absent
```

   Ce livre de jeu garantit l'absence de l'hôte **db.idm.example.com** du groupe d'hôtes **databases**. La ligne **action: member** indique que lors de l'exécution du livre de jeu, aucune tentative n'est faite pour supprimer le groupe **databases** lui-même.

3. Exécutez le manuel de jeu :

```
$ ansible-playbook --vault-password-file=password_file -v -i
path_to_inventory_directory/inventory.file path_to_playbooks_directory/ensure-hosts-
or-hostgroups-are-absent-in-hostgroup.yml
```

Verification steps

1. Connectez-vous à **ipaserver** en tant qu'administrateur :

```
$ ssh admin@server.idm.example.com
Password:
[admin@server /]$
```

2. Demander un ticket Kerberos pour l'administrateur :

```
$ kinit admin
Password for admin@IDM.EXAMPLE.COM:
```

3. Affiche des informations sur le groupe d'hôtes et les hôtes qu'il contient :

```
$ ipa hostgroup-show databases
  Host-group: databases
  Member host-groups: mysql-server, oracle-server
```

L'hôte **db.idm.example.com** n'existe pas dans le groupe d'hôtes **databases**.

43.7. GARANTIR L'ABSENCE DE GROUPES D'HÔTES IMBRIQUÉS À PARTIR DES GROUPES D'HÔTES IDM À L'AIDE DES PLAYBOOKS ANSIBLE

Cette section décrit comment garantir l'absence de groupes d'hôtes imbriqués dans les groupes d'hôtes externes dans la gestion des identités (IdM) à l'aide des playbooks Ansible.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.
- Les groupes d'hôtes auxquels vous faites référence dans le fichier du livre de jeu Ansible existent dans IdM. Pour plus de détails, voir Assurer la présence des groupes d'hôtes IdM à l'aide des playbooks Ansible.

Procédure

1. Créez un fichier d'inventaire, par exemple **inventory.file**, et définissez-y **ipaserver** avec la liste des serveurs IdM à cibler :

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

2. Créez un fichier playbook Ansible avec les informations nécessaires sur le groupe d'hôtes. Spécifiez, parmi les variables **- ipahostgroup**, le nom du groupe d'hôtes extérieur à l'aide de la variable **name**. Spécifiez le nom du groupe d'hôtes imbriqué à l'aide de la variable **hostgroup**. Pour simplifier cette étape, vous pouvez copier et modifier les exemples dans le fichier **/usr/share/doc/ansible-freeipa/playbooks/hostgroup/ensure-hosts-and-hostgroups-are-absent-in-hostgroup.yml**:

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

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  # Ensure hosts and hostgroups are absent in existing databases hostgroup
  - ipahostgroup:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: databases
      hostgroup:
      - mysql-server
      - oracle-server
      action: member
      state: absent
```

Ce playbook s'assure que les groupes d'hôtes **mysql-server** et **oracle-server** sont absents du groupe d'hôtes **databases**. La ligne **action: member** indique que lorsque le playbook est exécuté, aucune tentative n'est faite pour s'assurer que le groupe **databases** lui-même est supprimé de l'IdM.

3. Exécutez le manuel de jeu :

```
$ ansible-playbook --vault-password-file=password_file -v -i
path_to_inventory_directory/inventory.file path_to_playbooks_directory/ensure-hosts-
or-hostgroups-are-absent-in-hostgroup.yml
```

Verification steps

1. Connectez-vous à **ipaserver** en tant qu'administrateur :

```
$ ssh admin@server.idm.example.com
Password:
[admin@server /]$
```

2. Demander un ticket Kerberos pour l'administrateur :

```
$ kinit admin
Password for admin@IDM.EXAMPLE.COM:
```

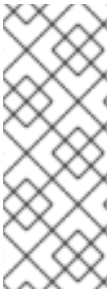
3. Afficher des informations sur le groupe d'hôtes à partir duquel les groupes d'hôtes imbriqués doivent être absents :

```
$ ipa hostgroup-show databases
  Host-group: databases
```

La sortie confirme que les groupes d'hôtes imbriqués **mysql-server** et **oracle-server** sont absents du groupe d'hôtes extérieur **databases**.

43.8. GARANTIR L'ABSENCE DE GROUPES D'HÔTES IDM À L'AIDE DE PLAYBOOKS ANSIBLE

Cette section décrit comment garantir l'absence de groupes d'hôtes dans la gestion des identités (IdM) à l'aide des playbooks Ansible.



NOTE

Sans Ansible, les entrées de groupes d'hôtes sont supprimées de l'IdM à l'aide de la commande **ipa hostgroup-del**. Le résultat de la suppression d'un groupe d'hôtes de l'IdM est l'état du groupe d'hôtes absent de l'IdM. En raison de la dépendance d'Ansible à l'idempotence, pour supprimer un groupe d'hôtes d'IdM à l'aide d'Ansible, vous devez créer un playbook dans lequel vous définissez l'état du groupe d'hôtes comme étant absent : **state: absent**

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

Procédure

1. Créez un fichier d'inventaire, par exemple **inventory.file**, et définissez-y **ipaserver** avec la liste des serveurs IdM à cibler :

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

2. Créez un fichier playbook Ansible avec les informations nécessaires sur le groupe d'hôtes. Pour simplifier cette étape, vous pouvez copier et modifier l'exemple dans le fichier **/usr/share/doc/ansible-freeipa/playbooks/user/ensure-hostgroup-is-absent.yml**.

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

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
```

```
- Ensure host-group databases is absent
  ipahostgroup:
    ipaadmin_password: "{{ ipaadmin_password }}"
    name: databases
    state: absent
```

Ce playbook garantit l'absence du groupe d'hôtes **databases** dans l'IdM. Le **state: absent** signifie une demande de suppression du groupe d'hôtes de l'IdM, à moins qu'il ne soit déjà supprimé.

3. Exécutez le manuel de jeu :

```
$ ansible-playbook --vault-password-file=password_file -v -i
path_to_inventory_directory/inventory.file path_to_playbooks_directory/ensure-
hostgroup-is-absent.yml
```

Verification steps

1. Connectez-vous à **ipaserver** en tant qu'administrateur :

```
$ ssh admin@server.idm.example.com
Password:
[admin@server /]$
```

2. Demander un ticket Kerberos pour l'administrateur :

```
$ kinit admin
Password for admin@IDM.EXAMPLE.COM:
```

3. Afficher des informations sur le groupe d'hôtes dont vous avez assuré l'absence :

```
$ ipa hostgroup-show databases
ipa: ERROR: databases: host group not found
```

Le groupe d'hôtes **databases** n'existe pas dans IdM.

43.9. ASSURER L'ABSENCE DES GESTIONNAIRES DE MEMBRES DES GROUPES HÔTES IDM À L'AIDE DES PLAYBOOKS ANSIBLE

La procédure suivante décrit comment garantir l'absence de gestionnaires membres dans les hôtes IdM et les groupes d'hôtes à l'aide d'un playbook Ansible.

Conditions préalables

- Vous connaissez le mot de passe de l'administrateur IdM.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.

- L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
- L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

- Vous devez disposer du nom de l'utilisateur ou du groupe d'utilisateurs que vous supprimez en tant que membres gestionnaires et du nom du groupe d'hôtes qu'ils gèrent.

Procédure

1. Créez un fichier d'inventaire, par exemple **inventory.file**, et définissez-y **ipaserver**:

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

2. Créer un fichier Ansible playbook avec les informations nécessaires à la gestion des hôtes et des membres du groupe d'hôtes :

```
---

- name: Playbook to handle host group membership management
  hosts: ipaserver

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  - name: Ensure member manager host and host group members are absent for
group_name
    ipahostgroup:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: group_name
      membermanager_user: example_member
      membermanager_group: project_admins
      action: member
      state: absent
```

3. Exécutez le manuel de jeu :

```
$ ansible-playbook --vault-password-file=password_file -v -i
path_to_inventory_directory/inventory.file path_to_playbooks_directory/ensure-
member-managers-host-groups-are-absent.yml
```

Verification steps

Vous pouvez vérifier si le groupe **group_name** ne contient pas **example_member** ou **project_admins** en tant que gestionnaires membres en utilisant la commande **ipa group-show**:

1. Connectez-vous à **ipaserver** en tant qu'administrateur :

```
$ ssh admin@server.idm.example.com
Password:
[admin@server /]$
```

2. Afficher des informations sur *testhostgroup*:

```
ipaserver]$ ipa hostgroup-show group_name
  Host-group: group_name
  Member hosts: server.idm.example.com
  Member host-groups: testhostgroup2
```

Ressources supplémentaires

- Voir **ipa hostgroup-add-member-manager --help**.
- Voir la page de manuel **ipa**.

CHAPITRE 44. ASSURER LA PRÉSENCE DE RÈGLES DE CONTRÔLE D'ACCÈS BASÉES SUR L'HÔTE DANS IDM EN UTILISANT LES PLAYBOOKS ANSIBLE

Ce chapitre décrit les stratégies d'accès basées sur l'hôte de la gestion des identités (IdM) et la manière de les définir à l'aide d'Ansible.

Ansible est un outil d'automatisation utilisé pour configurer des systèmes, déployer des logiciels et effectuer des mises à jour continues. Il inclut la prise en charge de la gestion des identités (IdM).

44.1. RÈGLES DE CONTRÔLE D'ACCÈS BASÉES SUR L'HÔTE DANS L'IDM

Les règles de contrôle d'accès basé sur l'hôte (HBAC) définissent quels utilisateurs ou groupes d'utilisateurs peuvent accéder à quels hôtes ou groupes d'hôtes en utilisant quels services ou services d'un groupe de services. En tant qu'administrateur système, vous pouvez utiliser les règles HBAC pour atteindre les objectifs suivants :

- Limiter l'accès à un système spécifique de votre domaine aux membres d'un groupe d'utilisateurs spécifique.
- Autoriser uniquement l'utilisation d'un service spécifique pour accéder aux systèmes de votre domaine.

Par défaut, IdM est configuré avec une règle HBAC par défaut nommée **allow_all**, ce qui signifie un accès universel à chaque hôte pour chaque utilisateur via chaque service pertinent dans l'ensemble du domaine IdM.

Vous pouvez affiner l'accès à différents hôtes en remplaçant la règle par défaut **allow_all** par votre propre ensemble de règles HBAC. Pour une gestion centralisée et simplifiée du contrôle d'accès, vous pouvez appliquer les règles HBAC à des groupes d'utilisateurs, des groupes d'hôtes ou des groupes de services plutôt qu'à des utilisateurs, des hôtes ou des services individuels.

44.2. ASSURER LA PRÉSENCE D'UNE RÈGLE HBAC DANS IDM À L'AIDE D'UN PLAYBOOK ANSIBLE

Cette section décrit comment assurer la présence d'une règle de contrôle d'accès basé sur l'hôte (HBAC) dans la gestion des identités (IdM) à l'aide d'un playbook Ansible.

Conditions préalables

- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :
  - Vous utilisez la version 2.8 ou ultérieure d'Ansible.
  - Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
  - L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
  - L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

- Les utilisateurs et les groupes d'utilisateurs que vous voulez utiliser pour votre règle HBAC existent dans IdM. Pour plus de détails, voir Gérer les comptes utilisateurs à l'aide des playbooks An sible et Assurer la présence des groupes IdM et des membres des groupes à l'aide des playbooks Ansible.

- Les hôtes et les groupes d'hôtes auxquels vous voulez appliquer votre règle HBAC existent dans IdM. Pour plus de détails, voir Gérer les hôtes à l'aide des playbooks Ansible et Gérer les groupes d'hôtes à l'aide des playbooks Ansible.

Procédure

1. Créez un fichier d'inventaire, par exemple **inventory.file**, et définissez-y **ipaserver**:

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

2. Créez votre fichier playbook Ansible qui définit la politique HBAC dont vous voulez assurer la présence. Pour simplifier cette étape, vous pouvez copier et modifier l'exemple dans le fichier **/usr/share/doc/ansible-freeipa/playbooks/hbacrule/ensure-hbacrule-allhosts-present.yml**:

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

  vars_files:
  - /home/user_name/MyPlaybooks/secret.yml
  tasks:
  # Ensure idm_user can access client.idm.example.com via the sshd service
  - ipahbacrule:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: login
      user: idm_user
      host: client.idm.example.com
      hbacsvc:
      - sshd
      state: present
```

3. Exécutez le manuel de jeu :

```
$ ansible-playbook --vault-password-file=password_file -v -i
path_to_inventory_directory/inventory.file path_to_playbooks_directory/ensure-new-
hbacrule-present.yml
```

Verification steps

1. Connectez-vous à l'interface Web IdM en tant qu'administrateur.
2. Naviguez vers **Policy → Host-Based-Access-Control → HBAC Test**
3. Dans l'onglet **Who**, sélectionnez idm_user.
4. Dans l'onglet **Accessing**, sélectionnez **client.idm.example.com**.
5. Dans l'onglet **Via service**, sélectionnez **sshd**.

6. Dans l'onglet **Rules**, sélectionnez **login**.

7. Dans l'onglet **Run test**, cliquez sur le bouton **Run test**. Si vous voyez ACCÈS ACCORDÉ, la règle HBAC a été mise en œuvre avec succès.

Ressources supplémentaires

- Voir les fichiers **README-hbacsvc.md**, **README-hbacsvcgroup.md**, et **README-hbacrule.md** dans le répertoire **/usr/share/doc/ansible-freeipa**.

- Voir les playbooks dans les sous-répertoires du répertoire **/usr/share/doc/ansible-freeipa/playbooks**.

CHAPITRE 45. GESTION DES CLÉS SSH PUBLIQUES POUR LES UTILISATEURS ET LES HÔTES

SSH (Secure Shell) est un protocole qui fournit des communications sécurisées entre deux systèmes utilisant une architecture client-serveur. SSH permet aux utilisateurs de se connecter à distance aux systèmes hôtes des serveurs et permet également à une machine hôte d'accéder à une autre machine.

45.1. A PROPOS DU FORMAT DES CLÉS SSH

L'IdM accepte les deux formats de clés SSH suivants :

- Clé de type OpenSSH
- Clé brute de type RFC 4253

Notez que IdM convertit automatiquement les clés de type RFC 4253 en clés de type OpenSSH avant de les enregistrer dans le serveur LDAP IdM.

Le serveur IdM peut identifier le type de clé, par exemple une clé RSA ou DSA, à partir du bloc de clés téléchargé. Dans un fichier de clés tel que **~/.ssh/known_hosts**, une entrée de clé est identifiée par le nom d'hôte et l'adresse IP du serveur, son type et la clé. Par exemple :

```
host.example.com, 1.2.3.4 ssh-rsa AAA...ZZZ==
```

Cela diffère de l'entrée d'une clé publique d'utilisateur, dont les éléments sont classés dans l'ordre *type key== comment*:

```
\N- "ssh-rsa ABCD1234...== ipaclient.example.com\N"
```

Un fichier de clé, tel que **id_rsa.pub**, se compose de trois parties : le type de clé, la clé et un commentaire ou un identifiant supplémentaire. Lorsque vous téléchargez une clé vers l'IdM, vous pouvez télécharger les trois parties de la clé ou seulement la clé. Si vous ne téléchargez que la clé, l'IdM identifie automatiquement le type de clé, tel que RSA ou DSA, à partir de la clé téléchargée.

Si vous utilisez l'entrée de la clé publique de l'hôte du fichier **~/.ssh/known_hosts**, vous devez la réorganiser pour qu'elle corresponde au format d'une clé utilisateur, *type key== comment*:

```
ssh-rsa AAA...ZZZ== host.example.com,1.2.3.4
```

L'IdM peut déterminer automatiquement le type de clé à partir du contenu de la clé publique. Le commentaire est facultatif, afin de faciliter l'identification des clés individuelles. Le seul élément requis est le blob de la clé publique.

IdM utilise des clés publiques stockées dans les fichiers OpenSSH suivants :

- Les clés publiques de l'hôte se trouvent dans le fichier **known_hosts**.
- Les clés publiques des utilisateurs se trouvent dans le fichier **authorized_keys**.

Ressources supplémentaires

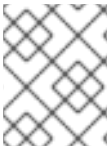
- Voir RFC 4716
- Voir RFC 4253

45.2. À PROPOS D'IDM ET D'OPENSSH

Lors de l'installation d'un serveur ou d'un client IdM, dans le cadre du script d'installation :

- Un serveur et un client OpenSSH sont configurés sur la machine du client IdM.
- SSSD est configuré pour stocker et récupérer les clés SSH de l'utilisateur et de l'hôte dans le cache. Cela permet à l'IdM de servir de dépôt universel et centralisé des clés SSH.

Si vous activez le service SSH lors de l'installation du client, une clé RSA est créée lors du premier démarrage du service SSH.



NOTE

Lorsque vous exécutez le script d'installation **ipa-client-install** pour ajouter la machine en tant que client IdM, le client est créé avec deux clés SSH, RSA et DSA.

Dans le cadre de l'installation, vous pouvez configurer les éléments suivants :

- Configurez OpenSSH pour qu'il fasse automatiquement confiance aux enregistrements DNS IdM où sont stockées les empreintes de clés en utilisant l'option **--ssh-trust-dns**.
- Désactive OpenSSH et empêche le script d'installation de configurer le serveur OpenSSH à l'aide de l'option **--no-sshd**.
- Empêcher l'hôte de créer des enregistrements DNS SSHFP avec ses propres entrées DNS à l'aide de l'option **--no-dns-sshfp**.

Si vous ne configurez pas le serveur ou le client pendant l'installation, vous pouvez configurer manuellement SSSD ultérieurement. Pour plus d'informations sur la configuration manuelle de SSSD, voir Configuration de SSSD pour fournir un cache aux services OpenSSH . Notez que la mise en cache des clés SSH par SSSD nécessite des privilèges administratifs sur les machines locales.

45.3. GÉNÉRER DES CLÉS SSH

Vous pouvez générer une clé SSH en utilisant l'utilitaire OpenSSH **ssh-keygen**.

Procédure

1. Pour générer une clé SSH RSA, exécutez la commande suivante :

```
$ ssh-keygen -t rsa -C user@example.com
Generating public/private rsa key pair.
```

   Remarque : si vous générez une clé d'hôte, remplacez user@example.com par le nom d'hôte requis, par exemple **server.example.com,1.2.3.4**.

2. Spécifiez le fichier dans lequel vous enregistrez la clé ou appuyez sur la touche Entrée pour accepter l'emplacement par défaut affiché.

```
Saisissez le fichier dans lequel vous souhaitez enregistrer la clé (/home/user/.ssh/id_rsa) :
```

Remarque Si vous générez une clé d'hôte, enregistrez la clé dans un emplacement différent du répertoire ~/**.ssh**/ de l'utilisateur afin de ne pas écraser les clés existantes. par exemple, /**home**/**user**/**.ssh**/**host_keys**.

3. Indiquez une phrase d'authentification pour votre clé privée ou appuyez sur la touche Entrée pour laisser la phrase d'authentification vide.

```
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/user/.ssh/id_rsa.
Your public key has been saved in /home/user/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:ONxjcMX7hJ5zly8F8ID9fpbqcuxQK+ylVLKDMsJPxGA user4@example.com
The key's randomart image is:
+---[RSA 3072]----+
|         ..o     |
|         .o +    |
|     E. .  o =   |
|    ..o=  o . +  |
|      +oS. = + o.|
|    . .o .* B =.+|
|     o + . X.+.= |
|      + o o.*+. .|
|       .   o=o . |
+----[SHA256]-----+
```

Pour télécharger cette clé SSH, utilisez la chaîne de clé publique stockée dans le fichier affiché.

45.4. GESTION DES CLÉS SSH PUBLIQUES POUR LES HÔTES

OpenSSH utilise des clés publiques pour authentifier les hôtes. Une machine tente d'accéder à une autre machine et présente sa paire de clés. La première fois que l'hôte s'authentifie, l'administrateur de la machine cible doit approuver la demande manuellement. La machine stocke ensuite la clé publique de l'hôte dans un fichier **known_hosts**. Chaque fois que la machine distante tente à nouveau d'accéder à la machine cible, cette dernière vérifie son fichier **known_hosts** et accorde automatiquement l'accès aux hôtes approuvés.

45.4.1. Téléchargement de clés SSH pour un hôte à l'aide de l'interface Web IdM

La gestion des identités vous permet de télécharger une clé publique SSH dans une entrée d'hôte. OpenSSH utilise des clés publiques pour authentifier les hôtes.

Conditions préalables

- Privilèges d'administrateur pour la gestion de l'interface Web IdM ou rôle d'administrateur des utilisateurs.

Procédure

1. Vous pouvez récupérer la clé de votre hôte à partir d'un fichier ~/**.ssh**/**known_hosts**. Par exemple, vous pouvez récupérer la clé de votre hôte à partir d'un fichier :

```
server.example.com,1.2.3.4 ssh-rsa
AAAAB3NzaC1yc2EAAAABIwAAQEApvjBvSFSkTU0WQW4eOweeo0DZZ08F9Ud21xly6FOhz
wpXFGIyxvXZ52 siHBHbbqGL5
```

```
14N7UvElruyslIHx9LYUR/pPKSMXCGyboLy5aTNl5OQ5EHrhVnFDIKXkvp45945R7SKYCUtR
umm0Iw6wq0XD4o ILeVbV3wmcB1bXs36ZvC/M6riefn9PcJmh6vNCvIsbMY6S
FhkWUTTiOXJUDYRLwM273FfWhzHK
SSQXeBp/zIn1gFvJhSZMRi9HZpDoqxLbBB9QIdIw6U4MIjNmKsSI/ASpkFm2GuQ7ZK9KuMIt
Y2AoCuIRmRAF8iYNHBTXfFurGogXwRDjQ==
```

Vous pouvez également générer une clé d'hôte. Voir Générer des clés SSH.

2. Copiez la clé publique du fichier de clés. L'entrée complète de la clé a la forme **host name,IP type key==**. Seul le **key==** est nécessaire, mais vous pouvez stocker l'ensemble de l'entrée. Pour utiliser tous les éléments de l'entrée, réorganisez l'entrée de manière à ce qu'elle ait l'ordre **type key== [host name,IP]**.

```
cat /home/user/.ssh/host_keys.pub
ssh-rsa AAAAB3NzaC1yc2E...tJG1PK2Mq++wQ== server.example.com,1.2.3.4
```

3. Connectez-vous à l'interface Web IdM.
4. Allez dans l'onglet **Identity>Hosts**.
5. Cliquez sur le nom de l'hôte à modifier.
6. Dans la section **Host Settings**, cliquez sur le bouton Clé publique SSH **Add**.
7. Collez la clé publique de l'hôte dans le champ **SSH public key**.
8. Cliquez sur **Set**.
9. Cliquez sur **Save** en haut de la fenêtre de l'interface Web IdM.

Vérification

- Dans la section **Hosts Settings**, vérifiez que la clé est répertoriée sous **SSH public keys**.

45.4.2. Téléchargement de clés SSH pour un hôte à l'aide de la CLI IdM

La gestion des identités vous permet de télécharger une clé publique SSH dans une entrée d'hôte. OpenSSH utilise des clés publiques pour authentifier les hôtes. Les clés SSH d'hôte sont ajoutées aux entrées d'hôte dans IdM, lorsque l'hôte est créé à l'aide de **host-add** ou en modifiant l'entrée ultérieurement.

Remarque Les clés hôte RSA et DSA sont créées par la commande **ipa-client-install**, sauf si le service SSH est explicitement désactivé dans le script d'installation.

Conditions préalables

- Privilèges d'administrateur pour la gestion de l'IdM ou rôle d'administrateur des utilisateurs.

Procédure

1. Exécutez la commande **host-mod** avec l'option **--sshpubkey** pour télécharger la clé publique codée en base64 dans l'entrée de l'hôte.
   Étant donné que l'ajout d'une clé d'hôte modifie l'enregistrement DNS Secure Shell fingerprint (SSHFP) de l'hôte, utilisez l'option **--updatedns** pour mettre à jour l'entrée DNS de l'hôte. Par exemple, vous pouvez utiliser l'option pour mettre à jour l'entrée DNS de l'hôte :

```
$ ipa host-mod --sshpubkey="ssh-rsa RjlzYQo==" --updatedns host1.example.com
```

Une clé réelle se termine généralement par un signe égal (=), mais elle est plus longue.

2. Pour télécharger plus d'une clé, entrez plusieurs paramètres de ligne de commande --sshpubkey :

```
--sshpubkey="RjlzYQo==" --sshpubkey="ZEt0TAo=="
```

NOTE

Un hôte peut avoir plusieurs clés publiques.

3. Après avoir téléchargé les clés d'hôte, configurez SSSD pour qu'il utilise la gestion des identités comme l'un de ses domaines d'identité et configurez OpenSSH pour qu'il utilise les outils SSSD afin de gérer les clés d'hôte, comme indiqué dans la section Configurer SSSD pour qu'il fournisse un cache pour les services OpenSSH.

Vérification

- Exécutez la commande **ipa host-show** pour vérifier que la clé publique SSH est associée à l'hôte spécifié :

```
$ ipa host-show client.ipa.test
...
SSH public key fingerprint:
SHA256:qGaqTZM60YPFTngFX0PtNPCKbIuudwf1D2LqmDeOcuA
                        client@IPA.TEST (ssh-rsa)
...
```

45.4.3. Suppression des clés SSH pour un hôte à l'aide de l'interface Web IdM

Vous pouvez supprimer les clés d'hôte lorsqu'elles expirent ou ne sont plus valides. Suivez les étapes ci-dessous pour supprimer une clé d'hôte individuelle à l'aide de l'interface Web IdM.

Conditions préalables

- Privilèges d'administrateur pour la gestion de l'interface Web IdM ou rôle d'administrateur d'hôte.

Procédure

1. Connectez-vous à l'interface Web IdM.
2. Allez dans l'onglet **Identity>Hosts**.
3. Cliquez sur le nom de l'hôte à modifier.
4. Dans la section **Host Settings**, cliquez sur **Delete** en regard de la clé publique SSH que vous souhaitez supprimer.
5. Cliquez sur **Save** en haut de la page.

Vérification

- Dans la section **Host Settings**, vérifiez que la clé n'est plus répertoriée sous **SSH public keys**.

45.4.4. Suppression des clés SSH pour un hôte à l'aide de la CLI IdM

Vous pouvez supprimer les clés d'hôte lorsqu'elles expirent ou ne sont plus valides. Suivez les étapes ci-dessous pour supprimer une clé d'hôte individuelle à l'aide de la CLI IdM.

Conditions préalables

- Privilèges d'administrateur pour gérer le CLI IdM ou le rôle d'administrateur de l'hôte.

Procédure

- Pour supprimer toutes les clés SSH attribuées à un compte hôte, ajoutez l'option **--sshpubkey** à la commande **ipa host-mod** sans spécifier de clé :

```
$ kinit admin
$ ipa host-mod --sshpubkey= --updatedns host1.example.com
```

  Il est conseillé d'utiliser l'option **--updatedns** pour mettre à jour l'entrée DNS de l'hôte.

L'IdM détermine automatiquement le type de clé à partir de la clé, si le type n'est pas inclus dans la clé téléchargée.

Vérification

- Exécutez la commande **ipa host-show** pour vérifier que la clé publique SSH n'est plus associée à l'hôte spécifié :

```
ipa host-show client.ipa.test
  Host name: client.ipa.test
  Platform: x86_64
  Operating system: 4.18.0-240.el8.x86_64
  Principal name: host/client.ipa.test@IPA.TEST
  Principal alias: host/client.ipa.test@IPA.TEST
  Password: False
  Member of host-groups: ipaservers
  Roles: helpdesk
  Member of netgroups: test
  Member of Sudo rule: test2
  Member of HBAC rule: test
  Keytab: True
  Managed by: client.ipa.test, server.ipa.test
  Users allowed to retrieve keytab: user1, user2, user3
```

45.5. GESTION DES CLÉS SSH PUBLIQUES POUR LES UTILISATEURS

La gestion des identités vous permet de télécharger une clé SSH publique dans l'entrée d'un utilisateur. L'utilisateur qui a accès à la clé SSH privée correspondante peut utiliser SSH pour se connecter à une machine IdM sans utiliser les informations d'identification Kerberos. Notez que les utilisateurs peuvent toujours s'authentifier en fournissant leurs informations d'identification Kerberos s'ils se connectent à partir d'une machine où leur fichier de clé SSH privée n'est pas disponible.

45.5.1. Téléchargement de clés SSH pour un utilisateur à l'aide de l'interface Web IdM

La gestion des identités vous permet de télécharger une clé SSH publique vers une entrée utilisateur. L'utilisateur qui a accès à la clé SSH privée correspondante peut utiliser SSH pour se connecter à une machine IdM sans utiliser les informations d'identification Kerberos.

Conditions préalables

- Privilèges d'administrateur pour la gestion de l'interface Web IdM ou rôle d'administrateur des utilisateurs.

Procédure

1. Connectez-vous à l'interface Web IdM.
2. Allez dans l'onglet **Identity>Users**.
3. Cliquez sur le nom de l'utilisateur à modifier.
4. Dans la section **Account Settings**, cliquez sur le bouton Clé publique SSH **Add**.
5. Collez la chaîne de la clé publique codée en base 64 dans le champ **SSH public key**.
6. Cliquez sur **Set**.
7. Cliquez sur **Save** en haut de la fenêtre de l'interface Web IdM.

Vérification

- Dans la section **Accounts Settings**, vérifiez que la clé est répertoriée sous **SSH public keys**.

45.5.2. Téléchargement de clés SSH pour un utilisateur à l'aide de la CLI IdM

La gestion des identités vous permet de télécharger une clé SSH publique vers une entrée utilisateur. L'utilisateur qui a accès à la clé SSH privée correspondante peut utiliser SSH pour se connecter à une machine IdM sans utiliser les informations d'identification Kerberos.

Conditions préalables

- Privilèges d'administrateur pour la gestion du CLI IdM ou rôle d'administrateur utilisateur.

Procédure

1. Exécutez la commande **ipa user-mod** avec l'option **--sshpubkey** pour télécharger la clé publique codée en base64 dans l'entrée utilisateur.

```
$ ipa user-mod user --sshpubkey="ssh-rsa AAAAB3Nza...SNc5dv== client.example.com"
```

   Notez que dans cet exemple, vous téléchargez le type de clé, la clé et l'identifiant du nom d'hôte dans l'entrée de l'utilisateur.

2. Pour télécharger plusieurs clés, utilisez plusieurs fois **--sshpubkey**. Par exemple, pour télécharger deux clés SSH :

```
--sshpubkey="AAAAB3Nza...SNc5dv==" --sshpubkey="RjlzYQo...ZEt0TAo="
```

3. Pour utiliser la redirection de commandes et pointer vers un fichier contenant la clé au lieu de coller la chaîne de clés manuellement, utilisez la commande suivante :

```
ipa user-mod user --sshpubkey="$(cat ~/.ssh/id_rsa.pub)" --sshpubkey="$(cat
~/.ssh/id_rsa2.pub)"
```

Vérification

- Exécutez la commande **ipa user-show** pour vérifier que la clé publique SSH est associée à l'utilisateur spécifié :

```
$ ipa user-show user
User login: user
  First name: user
  Last name: user
  Home directory: /home/user
  Login shell: /bin/sh
  Principal name: user@IPA.TEST
  Principal alias: user@IPA.TEST
  Email address: user@ipa.test
  UID: 1118800019
  GID: 1118800019
  SSH public key fingerprint:
SHA256:qGaqTZM60YPFTngFX0PtNPCKbIuudwf1D2LqmDeOcuA
                        user@IPA.TEST (ssh-rsa)
  Account disabled: False
  Password: False
  Member of groups: ipausers
  Subordinate ids: 3167b7cc-8497-4ff2-ab4b-6fcb3cb1b047
  Kerberos keys available: False
```

45.5.3. Suppression des clés SSH pour un utilisateur à l'aide de l'interface Web IdM

Suivez cette procédure pour supprimer une clé SSH d'un profil d'utilisateur dans l'interface Web IdM.

Conditions préalables

- Privilèges d'administrateur pour la gestion de l'interface Web IdM ou rôle d'administrateur des utilisateurs.

Procédure

1. Connectez-vous à l'interface Web IdM.
2. Allez dans l'onglet **Identity>Users**.
3. Cliquez sur le nom de l'utilisateur à modifier.
4. Dans la section **Account Settings**, sous **SSH public key**, cliquez sur **Delete** à côté de la clé que vous souhaitez supprimer.
5. Cliquez sur **Save** en haut de la page.

Vérification

- Dans la section **Account Settings**, vérifiez que la clé n'est plus répertoriée sous **SSH public keys**.

45.5.4. Suppression des clés SSH pour un utilisateur à l'aide de la CLI IdM

Suivez cette procédure pour supprimer une clé SSH d'un profil d'utilisateur à l'aide de la CLI IdM.

Conditions préalables

- Privilèges d'administrateur pour la gestion du CLI IdM ou rôle d'administrateur utilisateur.

Procédure

1. Pour supprimer toutes les clés SSH attribuées à un compte utilisateur, ajoutez l'option **--sshpubkey** à la commande **ipa user-mod** sans spécifier de clé :

```
$ ipa user-mod user --sshpubkey=
```

2. Pour ne supprimer qu'une ou plusieurs clés SSH spécifiques, utilisez l'option **--sshpubkey** pour spécifier les clés que vous souhaitez conserver, en omettant la clé que vous supprimez.

Vérification

- Exécutez la commande **ipa user-show** pour vérifier que la clé publique SSH n'est plus associée à l'utilisateur spécifié :

```
$ ipa user-show user
User login: user
  First name: user
  Last name: user
  Home directory: /home/user
  Login shell: /bin/sh
  Principal name: user@IPA.TEST
  Principal alias: user@IPA.TEST
  Email address: user@ipa.test
  UID: 1118800019
  GID: 1118800019
  Account disabled: False
  Password: False
  Member of groups: ipausers
  Subordinate ids: 3167b7cc-8497-4ff2-ab4b-6fcb3cb1b047
  Kerberos keys available: False
```

CHAPITRE 46. CONFIGURATION DE L'ORDRE DE RÉSOLUTION DU DOMAINE POUR RÉSOUDRE LES NOMS D'UTILISATEUR AD COURTS

Par défaut, vous devez spécifier des noms entièrement qualifiés au format **user_name@domain.com** ou **domain.com\user_name** pour résoudre et authentifier les utilisateurs et les groupes d'un environnement Active Directory (AD). Les sections suivantes décrivent comment configurer les serveurs et les clients IdM pour résoudre les noms d'utilisateur et de groupe AD abrégés.

- Comment fonctionne l'ordre de résolution de domaine
- Définition de l'ordre de résolution du domaine global sur un serveur IdM
- Définition de l'ordre de résolution des domaines pour une vue ID sur un serveur IdM
- Définition de l'ordre de résolution des domaines dans SSSD sur un client IdM

46.1. COMMENT FONCTIONNE L'ORDRE DE RÉSOLUTION DE DOMAINE

Dans les environnements de gestion d'identité (IdM) avec une confiance Active Directory (AD), Red Hat recommande de résoudre et d'authentifier les utilisateurs et les groupes en spécifiant leurs noms entièrement qualifiés. Par exemple, Red Hat recommande de résoudre et d'authentifier les utilisateurs et les groupes en spécifiant leur nom complet :

- ***<idm_username>*@idm.example.com** pour les utilisateurs IdM du domaine **idm.example.com**
- ***<ad_username>*@ad.example.com** pour les utilisateurs AD du domaine **ad.example.com**

Par défaut, si vous effectuez des recherches d'utilisateurs ou de groupes en utilisant le format *short name*, tel que **ad_username**, IdM ne recherche que le domaine IdM et ne parvient pas à trouver les utilisateurs ou les groupes AD. Pour résoudre les utilisateurs ou les groupes AD à l'aide de noms courts, modifiez l'ordre dans lequel l'IdM recherche plusieurs domaines en définissant l'option **domain resolution order**.

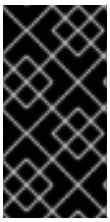
Vous pouvez définir l'ordre de résolution des domaines de manière centralisée dans la base de données IdM ou dans la configuration SSSD des clients individuels. L'IdM évalue l'ordre de résolution des domaines dans l'ordre de priorité suivant :

- La configuration locale de **/etc/sssd/sssd.conf**.
- La configuration de la vue ID.
- La configuration globale de l'IdM.

Notes

- Vous devez utiliser des noms d'utilisateur complets si la configuration SSSD sur l'hôte inclut l'option **default_domain_suffix** et que vous souhaitez adresser une requête à un domaine non spécifié avec cette option.
- Si vous utilisez l'option **domain resolution order** et que vous interrogez l'arbre **compat**, il se peut que vous receviez plusieurs identifiants d'utilisateur (UID). Si vous êtes concerné, consultez le rapport de bogue de Pagure intitulé Inconsistent compat user objects for AD users when

domain resolution order is set (Objets utilisateur compat incohérents pour les utilisateurs AD lorsque l'ordre de résolution du domaine est défini).



IMPORTANT

N'utilisez pas l'option **full_name_format** SSSD sur les clients ou les serveurs IdM. L'utilisation d'une valeur autre que la valeur par défaut pour cette option modifie l'affichage des noms d'utilisateur et peut perturber les recherches dans un environnement IdM.

Ressources supplémentaires

- Active Directory Trust pour les anciens clients Linux .

46.2. DÉFINITION DE L'ORDRE DE RÉSOLUTION DU DOMAINE GLOBAL SUR UN SERVEUR IDM

Cette procédure définit l'ordre de résolution du domaine pour tous les clients du domaine IdM. Cet exemple définit l'ordre de résolution du domaine pour rechercher les utilisateurs et les groupes dans l'ordre suivant :

1. Domaine racine de l'Active Directory (AD) **ad.example.com**
2. Domaine enfant AD **subdomain1.ad.example.com**
3. Domaine IdM **idm.example.com**

Conditions préalables

- Vous avez configuré un trust avec un environnement AD.

Procédure

- Utilisez la commande **ipa config-mod --domain-resolution-order** pour dresser la liste des domaines à rechercher dans l'ordre de votre choix. Séparez les domaines par deux points (**:**).

```
[user@server ~]$ ipa config-mod --domain-resolution-
order='ad.example.com:subdomain1.ad.example.com:idm.example.com'
Maximum username length: 32
Home directory base: /home
...
  Domain Resolution Order:
ad.example.com:subdomain1.ad.example.com:idm.example.com
...
```

Verification steps

- Vérifiez que vous pouvez récupérer les informations d'un utilisateur du domaine **ad.example.com** en utilisant uniquement un nom court.

```
[root@client ~]# id <ad_username>
uid=1916901102(ad_username) gid=1916900513(domain users)
groups=1916900513(domain users)
```

46.3. DÉFINITION DE L'ORDRE DE RÉSOLUTION DES DOMAINES POUR UNE VUE ID SUR UN SERVEUR IDM

Cette procédure définit l'ordre de résolution du domaine pour une vue ID que vous pouvez appliquer à un ensemble spécifique de serveurs et de clients IdM. Cet exemple crée une vue ID nommée **ADsubdomain1_first** pour l'hôte IdM **client1.idm.example.com**, et définit l'ordre de résolution du domaine pour rechercher les utilisateurs et les groupes dans l'ordre suivant :

1. Domaine enfant Active Directory (AD) **subdomain1.ad.example.com**
2. Domaine racine AD **ad.example.com**
3. Domaine IdM **idm.example.com**

NOTE

L'ordre de résolution de domaine défini dans une vue ID remplace l'ordre de résolution de domaine global, mais il ne remplace pas l'ordre de résolution de domaine défini localement dans la configuration SSSD.

Conditions préalables

- Vous avez configuré un trust avec un environnement AD.

Procédure

1. Créez une vue ID avec l'option **--domain-resolution-order**.

```
[user@server ~]$ ipa idview-add ADsubdomain1_first --desc "ID view for resolving AD
subdomain1 first on client1.idm.example.com" --domain-resolution-order
subdomain1.ad.example.com:ad.example.com:idm.example.com
---------------------------------
Added ID View "ADsubdomain1_first"
---------------------------------
ID View Name: ADsubdomain1_first
Description: ID view for resolving AD subdomain1 first on client1.idm.example.com
Domain Resolution Order:
subdomain1.ad.example.com:ad.example.com:idm.example.com
```

2. Appliquer la vue ID aux hôtes IdM.

```
[user@server ~]$ ipa idview-apply ADsubdomain1_first --hosts
client1.idm.example.com
-----------------------------------
Applied ID View "ADsubdomain1_first"
-----------------------------------
  hosts: client1.idm.example.com
-------------------------------------------
Number of hosts the ID View was applied to: 1
-------------------------------------------
```

Verification steps

- Affiche les détails de la vue ID.

```
[user@server ~]$ ipa idview-show ADsubdomain1_first --show-hosts
  ID View Name: ADsubdomain1_first
  Description: ID view for resolving AD subdomain1 first on client1.idm.example.com
  Hosts the view applies to: client1.idm.example.com
  Domain resolution order:
subdomain1.ad.example.com:ad.example.com:idm.example.com
```

- Vérifiez que vous pouvez récupérer les informations d'un utilisateur du domaine **subdomain1.ad.example.com** en utilisant uniquement un nom court.

```
[root@client1 ~]# id <user_from_subdomain1>
uid=1916901106(user_from_subdomain1) gid=1916900513(domain users)
groups=1916900513(domain users)
```

46.4. DÉFINITION DE L'ORDRE DE RÉSOLUTION DES DOMAINES DANS SSSD SUR UN CLIENT IDM

Cette procédure définit l'ordre de résolution des domaines dans la configuration SSSD d'un client IdM. Cet exemple configure l'hôte IdM **client2.idm.example.com** pour qu'il recherche les utilisateurs et les groupes dans l'ordre suivant :

1. Domaine enfant Active Directory (AD) **subdomain1.ad.example.com**
2. Domaine racine AD **ad.example.com**
3. Domaine IdM **idm.example.com**

NOTE

L'ordre de résolution des domaines dans la configuration locale du SSSD est prioritaire sur l'ordre de résolution des domaines de la vue globale et de la vue ID.

Conditions préalables

- Vous avez configuré un trust avec un environnement AD.

Procédure

1. Ouvrez le fichier **/etc/sssd/sssd.conf** dans un éditeur de texte.
2. Définissez l'option **domain_resolution_order** dans la section **[sssd]** du fichier.

```
ordre_de_résolution_du_domaine = sous-domaine1.ad.exemple.com, ad.exemple.com,
idm.exemple.com
```

3. Enregistrez et fermez le fichier.
4. Restart the SSSD service to load the new configuration settings.

```
[root@client2 ~]# systemctl restart sssd
```

Étapes de la vérification

- Vérifiez que vous pouvez récupérer les informations d'un utilisateur du domaine **subdomain1.ad.example.com** en utilisant uniquement un nom court.

```
[root@client2 ~]# id <user_from_subdomain1>
uid=1916901106(user_from_subdomain1) gid=1916900513(domain users)
groups=1916900513(domain users)
```

46.5. RESSOURCES SUPPLÉMENTAIRES

- Utilisation d'une vue ID pour remplacer une valeur d'attribut utilisateur sur un client IdM

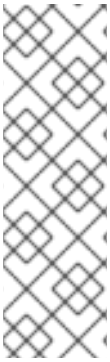
CHAPITRE 47. ACTIVATION DE L'AUTHENTIFICATION À L'AIDE DES NOMS DE PRINCIPAUX D'UTILISATEURS AD DANS L'IDM

47.1. NOMS DES PRINCIPAUX UTILISATEURS DANS UNE FORÊT AD APPROUVÉE PAR IDM

En tant qu'administrateur Identity Management (IdM), vous pouvez autoriser les utilisateurs AD à utiliser des **User Principal Names** (UPN) alternatifs pour accéder aux ressources du domaine IdM. Un UPN est un login alternatif avec lequel les utilisateurs AD s'authentifient au format **user_name@KERBEROS-REALM**. En tant qu'administrateur AD, vous pouvez définir des valeurs alternatives pour **user_name** et **KERBEROS-REALM**, puisque vous pouvez configurer à la fois des alias Kerberos supplémentaires et des suffixes UPN dans une forêt AD.

Par exemple, si une entreprise utilise le domaine Kerberos **AD.EXAMPLE.COM**, l'UPN par défaut d'un utilisateur est **user@ad.example.com**. Pour permettre à vos utilisateurs de se connecter à l'aide de leur adresse électronique, par exemple **user@example.com**vous pouvez configurer **EXAMPLE.COM** comme UPN alternatif dans AD. Les UPN alternatifs (également connus sous le nom de *enterprise UPNs*) sont particulièrement pratiques si votre entreprise a récemment fait l'objet d'une fusion et que vous souhaitez fournir à vos utilisateurs un espace de noms de connexion unifié.

Les suffixes UPN ne sont visibles pour IdM que lorsqu'ils sont définis à la racine de la forêt AD. En tant qu'administrateur AD, vous pouvez définir les UPN à l'aide de l'utilitaire **Active Directory Domain and Trust** ou de l'outil de ligne de commande **PowerShell**.



NOTE

Pour configurer les suffixes UPN pour les utilisateurs, Red Hat recommande d'utiliser des outils qui effectuent une validation d'erreur, tels que l'utilitaire **Active Directory Domain and Trust**.

Red Hat recommande de ne pas configurer les UPN par le biais de modifications de bas niveau, telles que l'utilisation de commandes **ldapmodify** pour définir l'attribut **userPrincipalName** pour les utilisateurs, car Active Directory ne valide pas ces opérations.

Après avoir défini un nouvel UPN du côté AD, exécutez la commande **ipa trust-fetch-domains** sur un serveur IdM pour récupérer les UPN mis à jour. Voir S'assurer que les UPN AD sont à jour dans IdM .

L'IdM stocke les suffixes UPN d'un domaine dans l'attribut multivaleur **ipaNTAdditionalSuffixes** du sous-arbre **cn=trusted_domain_name,cn=ad,cn=trusts,dc=idm,dc=example,dc=com**.

Ressources supplémentaires

- Comment configurer le suffixe UPN à la racine de la forêt AD ?
- Comment modifier manuellement les entrées d'utilisateurs AD et contourner la validation du suffixe UPN ?
- Contrôleurs et agents de confiance

47.2. VEILLER À CE QUE LES UPN AD SOIENT À JOUR DANS IDM

Après avoir ajouté ou supprimé un suffixe de nom de principal utilisateur (UPN) dans une forêt Active Directory (AD) de confiance, actualisez les informations relatives à la forêt de confiance sur un serveur IdM.

Conditions préalables

- Informations d'identification de l'administrateur de l'IdM.

Procédure

- Entrez la commande **ipa trust-fetch-domains**. Notez qu'une sortie apparemment vide est attendue :

```
[root@ipaserver ~]# ipa trust-fetch-domains
Realm-Name: ad.example.com
--------------------------------
No new trust domains were found
--------------------------------
----------------------------
Number of entries returned 0
----------------------------
```

Verification steps

- Entrez la commande **ipa trust-show** pour vérifier que le serveur a récupéré le nouvel UPN. Spécifiez le nom du domaine AD lorsque vous y êtes invité :

```
[root@ipaserver ~]# ipa trust-show
Realm-Name: ad.example.com
  Realm-Name: ad.example.com
  Domain NetBIOS name: AD
  Domain Security Identifier: S-1-5-21-796215754-1239681026-23416912
  Trust direction: One-way trust
  Trust type: Active Directory domain
  UPN suffixes: example.com
```

La sortie montre que le suffixe UPN **example.com** fait maintenant partie de l'entrée de domaine **ad.example.com**.

47.3. COLLECTE DE DONNÉES DE DÉPANNAGE POUR LES PROBLÈMES D'AUTHENTIFICATION AD UPN

Cette procédure décrit comment recueillir des données de dépannage sur la configuration du nom principal de l'utilisateur (UPN) dans votre environnement Active Directory (AD) et votre environnement IdM. Si vos utilisateurs AD ne parviennent pas à se connecter à l'aide d'autres UPN, vous pouvez utiliser ces informations pour limiter vos efforts de dépannage.

Conditions préalables

- Vous devez être connecté à un contrôleur IdM Trust ou à un agent Trust pour récupérer les informations d'un contrôleur de domaine AD.

- Vous devez disposer des autorisations **root** pour modifier les fichiers de configuration suivants et pour redémarrer les services IdM.

Procédure

1. Ouvrez le fichier de configuration **/usr/share/ipa/smb.conf.empty** dans un éditeur de texte.
2. Ajoutez le contenu suivant au fichier.

```
[global]
log level = 10
```

3. Enregistrez et fermez le fichier **/usr/share/ipa/smb.conf.empty**.
4. Ouvrez le fichier de configuration **/etc/ipa/server.conf** dans un éditeur de texte. Si vous n'avez pas ce fichier, créez-en un.
5. Ajoutez le contenu suivant au fichier.

```
[global]
debug = True
```

6. Enregistrez et fermez le fichier **/etc/ipa/server.conf**.
7. Redémarrez le service du serveur web Apache pour appliquer les changements de configuration :

```
[root@server ~]# systemctl restart httpd
```

8. Récupérez les informations de confiance de votre domaine AD :

```
[root@server ~]# ipa trust-fetch-domains <ad.example.com>
```

9. Examinez les données de débogage et les informations de dépannage dans les fichiers journaux suivants :
   - **/var/log/httpd/error_log**
   - **/var/log/samba/log.***

Ressources supplémentaires

- Voir Utilisation de rpcclient pour recueillir des données de dépannage concernant les problèmes d'authentification AD UPN.

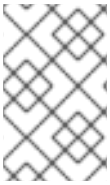
CHAPITRE 48. PERMETTRE AUX UTILISATEURS AD D'ADMINISTRER L'IDM

48.1. REMPLACEMENT DES ID POUR LES UTILISATEURS AD

Vous pouvez gérer de manière centralisée l'accès des utilisateurs et des groupes Active Directory (AD) aux ressources Identity Management (IdM) dans un environnement POSIX en ajoutant une dérogation d'utilisateur ID pour un utilisateur AD en tant que membre d'un groupe IdM.

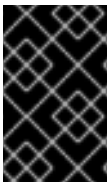
Un remplacement d'ID est un enregistrement décrivant les propriétés d'un utilisateur ou d'un groupe Active Directory spécifique dans une vue d'ID spécifique, en l'occurrence **Default Trust View**. Grâce à cette fonctionnalité, le serveur LDAP IdM est en mesure d'appliquer les règles de contrôle d'accès du groupe IdM à l'utilisateur AD.

Les utilisateurs AD peuvent utiliser les fonctions en libre-service de l'interface IdM, par exemple pour télécharger leurs clés SSH ou modifier leurs données personnelles. Un administrateur AD est en mesure d'administrer entièrement IdM sans avoir deux comptes et mots de passe différents.



NOTE

Actuellement, certaines fonctionnalités d'IdM peuvent encore être indisponibles pour les utilisateurs AD. Par exemple, la définition de mots de passe pour les utilisateurs IdM en tant qu'utilisateur AD du groupe IdM **admins** peut échouer.



IMPORTANT

N'utilisez pas **not** les ID overrides des utilisateurs AD pour les règles **sudo** dans IdM. Les substitutions d'ID des utilisateurs AD ne représentent que les attributs POSIX des utilisateurs AD, et non les utilisateurs AD eux-mêmes.

Ressources supplémentaires

- Utilisation des vues d'identification pour les utilisateurs d'Active Directory

48.2. UTILISATION DES DÉROGATIONS D'ID POUR PERMETTRE AUX UTILISATEURS D'AD D'ADMINISTRER L'IDM

Cette procédure décrit la création et l'utilisation d'un remplacement d'ID pour un utilisateur AD afin de lui donner des droits identiques à ceux d'un utilisateur IdM. Au cours de cette procédure, travaillez sur un serveur IdM configuré en tant que contrôleur de confiance ou agent de confiance.

Conditions préalables

- Un environnement IdM fonctionnel est mis en place. Pour plus de détails, voir Installation de la gestion des identités.
- Une confiance fonctionnelle est établie entre votre environnement IdM et AD.

Procédure

1. En tant qu'administrateur IdM, créez un ID override pour un utilisateur AD dans le site **Default Trust View**. Par exemple, pour créer un ID override pour l'utilisateur **ad_user@ad.example.com**:

```
# kinit admin
# ipa idoverrideuser-add 'default trust view' ad_user@ad.example.com
```

2. Ajoutez l'ID override du site **Default Trust View** en tant que membre d'un groupe IdM. Il doit s'agir d'un groupe non-POSIX, car il interagit avec Active Directory.
Si le groupe en question est membre d'un rôle IdM, l'utilisateur AD représenté par l'ID override obtient toutes les autorisations accordées par le rôle lors de l'utilisation de l'API IdM, y compris l'interface de ligne de commande et l'interface web IdM.

   Par exemple, pour ajouter le remplacement de l'ID de l'utilisateur **ad_user@ad.example.com** au groupe IdM **admins**:

```
# ipa group-add-member admins --idoverrideusers=ad_user@ad.example.com
```

3. Vous pouvez également ajouter l'annulation de l'ID à un rôle, tel que le rôle **User Administrator**:

```
# ipa role-add-member 'User Administrator' --
idoverrideusers=ad_user@ad.example.com
```

Ressources supplémentaires

- Utilisation des vues d'identification pour les utilisateurs d'Active Directory

48.3. UTILISER ANSIBLE POUR PERMETTRE AUX UTILISATEURS AD D'ADMINISTRER IDM

Cette section décrit comment utiliser un playbook Ansible pour s'assurer qu'un remplacement d'ID d'utilisateur est présent dans un groupe de gestion des identités (IdM). Le remplacement de l'ID utilisateur est le remplacement d'un utilisateur Active Directory (AD) que vous avez créé dans la vue de confiance par défaut après avoir établi une confiance avec AD. Suite à l'exécution du playbook, un utilisateur AD, par exemple un administrateur AD, est en mesure d'administrer entièrement l'IdM sans avoir deux comptes et mots de passe différents.

Conditions préalables

- Vous connaissez le mot de passe de l'IdM **admin**.
- Vous avez installé un trust avec AD.
- Le remplacement de l'ID de l'utilisateur AD existe déjà dans l'IdM. Si ce n'est pas le cas, créez-le avec la commande **ipa idoverrideuser-add 'default trust view'** ***ad_user@ad.example.com*** commande.
- Le groupe auquel vous ajoutez le remplacement de l'ID utilisateur existe déjà dans IdM .
- Vous utilisez la version 4.8.7 d'IdM ou une version ultérieure. Pour connaître la version d'IdM installée sur votre serveur, entrez **ipa --version**.
- Vous avez configuré votre nœud de contrôle Ansible pour qu'il réponde aux exigences suivantes :

- Vous utilisez la version 2.8 ou ultérieure d'Ansible.
- Vous avez installé le paquetage **ansible-freeipa** sur le contrôleur Ansible.
- L'exemple suppose que dans le répertoire ***~/MyPlaybooks/*** vous avez créé un fichier d'inventaire Ansible avec le nom de domaine complet (FQDN) du serveur IdM.
- L'exemple suppose que le coffre-fort **secret.yml** Ansible stocke votre **ipaadmin_password**.

Procédure

1. Naviguez jusqu'à votre répertoire ***~/MyPlaybooks/*** répertoire :

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

2. Créez un playbook **add-useridoverride-to-group.yml** avec le contenu suivant :

```
---
- name: Playbook to ensure presence of users in a group
  hosts: ipaserver


  - name: Ensure the ad_user@ad.example.com user ID override is a member of the admins
group:
    ipagroup:
      ipaadmin_password: "{{ ipaadmin_password }}"
      name: admins
      idoverrideuser:
      - ad_user@ad.example.com
```

Dans l'exemple :

- Secret123 est le mot de passe de l'IdM **admin**.
- **admins** est le nom du groupe POSIX IdM auquel vous ajoutez l'annulation de l'ID **ad_user@ad.example.com**. Les membres de ce groupe disposent de tous les privilèges d'administrateur.
- **ad_user@ad.example.com** est le remplacement de l'ID utilisateur d'un administrateur AD. L'utilisateur est stocké dans le domaine AD avec lequel une confiance a été établie.

3. Enregistrer le fichier.
4. Exécutez le playbook Ansible. Spécifiez le fichier du livre de jeu, le fichier contenant le mot de passe protégeant le fichier **secret.yml** et le fichier d'inventaire :

```
$ ansible-playbook --vault-password-file=password_file -v -i inventory add-
useridoverride-to-group.yml
```

Ressources supplémentaires

- Remplacement des ID pour les utilisateurs AD
- /usr/share/doc/ansible-freeipa/README-group.md

- /usr/share/doc/ansible-freeipa/playbooks/user
- Utilisation des vues d'identification dans les environnements Active Directory

48.4. VÉRIFIER QU'UN UTILISATEUR AD PEUT EXÉCUTER DES COMMANDES CORRECTES DANS LE CLI IDM

Cette procédure permet de vérifier qu'un utilisateur d'Active Directory (AD) peut se connecter à l'interface de ligne de commande (CLI) de Identity Management (IdM) et exécuter les commandes correspondant à son rôle.

1. Détruire le ticket Kerberos actuel de l'administrateur IdM :

```
# kdestroy -A
```

NOTE

La destruction du ticket Kerberos est nécessaire parce que l'implémentation GSSAPI dans MIT Kerberos choisit de préférence des informations d'identification dans le domaine du service cible, qui dans ce cas est le domaine IdM. Cela signifie que si une collection de cache d'informations d'identification, à savoir le type de cache d'informations d'identification **KCM:**, **KEYRING:**, ou **DIR:**, est utilisée, les informations d'identification obtenues précédemment par **admin** ou tout autre principal IdM seront utilisées pour accéder à l'API IdM au lieu des informations d'identification de l'utilisateur AD.

2. Obtenir les informations d'identification Kerberos de l'utilisateur AD pour lequel un remplacement d'ID a été créé :

```
# kinit ad_user@AD.EXAMPLE.COM
Password for ad_user@AD.EXAMPLE.COM:
```

3. Vérifier que l'ID override de l'utilisateur AD bénéficie des mêmes privilèges découlant de l'appartenance au groupe IdM que n'importe quel utilisateur IdM de ce groupe. Si l'ID override de l'utilisateur AD a été ajouté au groupe **admins**, l'utilisateur AD peut, par exemple, créer des groupes dans IdM :

```
# ipa group-add some-new-group
---------------------------
Added group "some-new-group"
---------------------------
  Group name: some-new-group
  GID: 1997000011
```

CHAPITRE 49. UTILISATION DE FOURNISSEURS D'IDENTITÉ EXTERNES POUR S'AUTHENTIFIER AUPRÈS DE L'IDM

Cette section aborde les sujets suivants :

- Les avantages de la connexion de l'IdM à un IdP externe
- Création d'une référence à un fournisseur d'identité externe
- Gestion des références à des IdP externes
- Permettre à un utilisateur IdM de s'authentifier via un IdP externe
- Récupération d'un ticket IdM en tant qu'utilisateur IdP
- Connexion à un client IdM via SSH en tant qu'utilisateur IdP
- Liste des modèles pour les fournisseurs d'identité externes

49.1. LES AVANTAGES DE LA CONNEXION D'IDM À UN IDP EXTERNE

En tant qu'administrateur, vous pouvez souhaiter autoriser les utilisateurs stockés dans une source d'identité externe, telle qu'un fournisseur de services en nuage, à accéder aux systèmes RHEL reliés à votre environnement de gestion des identités (IdM). Pour ce faire, vous pouvez déléguer à cette entité externe le processus d'authentification et d'autorisation consistant à émettre des tickets Kerberos pour ces utilisateurs.

Vous pouvez utiliser cette fonction pour étendre les capacités de l'IdM et permettre aux utilisateurs stockés dans des fournisseurs d'identité externes (IdP) d'accéder aux systèmes Linux gérés par l'IdM.

49.1.1. Comment l'IdM intègre-t-il les connexions via des IdP externes ?

SSSD 2.7.0 contient le paquetage **sssd-idp**, qui met en œuvre la méthode de pré-authentification Kerberos **idp**. Cette méthode d'authentification suit le flux OAuth 2.0 Device Authorization Grant pour déléguer les décisions d'autorisation à des IdP externes :

1. Un utilisateur client IdM lance le flux OAuth 2.0 Device Authorization Grant, par exemple, en tentant de récupérer un TGT Kerberos à l'aide de l'utilitaire **kinit** en ligne de commande.
2. Un code spécial et un lien vers le site web sont envoyés par le serveur d'autorisation au backend IdM KDC.
3. Le client IdM affiche le lien et le code à l'utilisateur. Dans cet exemple, le client IdM affiche le lien et le code sur la ligne de commande.
4. L'utilisateur ouvre le lien du site web dans un navigateur, qui peut se trouver sur un autre hôte, un téléphone portable, etc :
    a. L'utilisateur introduit le code spécial.
    b. Si nécessaire, l'utilisateur se connecte à l'IdP basé sur OAuth 2.0.
    c. L'utilisateur est invité à autoriser le client à accéder aux informations.
5. L'utilisateur confirme l'accès à l'invite du dispositif d'origine. Dans cet exemple, l'utilisateur appuie sur la touche **Entrée de** la ligne de commande.

6. Le backend IdM KDC interroge le serveur d'autorisation OAuth 2.0 pour accéder aux informations de l'utilisateur.

Ce qui est pris en charge :

- Connexion à distance via SSH avec la méthode d'authentification **keyboard-interactive** activée, ce qui permet d'appeler les bibliothèques PAM (Pluggable Authentication Module).
- Connexion locale à la console via le service **logind**.
- Récupération d'un ticket Kerberos (TGT) avec l'utilitaire **kinit**.

Ce qui n'est pas pris en charge actuellement :

- Se connecter directement à l'IdM WebUI. Pour se connecter à l'IdM WebUI, il faut d'abord obtenir un ticket Kerberos.
- Se connecter directement au Cockpit WebUI. Pour se connecter au Cockpit WebUI, il faut d'abord acquérir un ticket Kerberos.

Ressources supplémentaires

- Authentification par rapport à des fournisseurs d'identité externes
- RFC 8628 : OAuth 2.0 Device Authorization Grant (Accord d'autorisation de dispositif)

49.2. CRÉATION D'UNE RÉFÉRENCE À UN FOURNISSEUR D'IDENTITÉ EXTERNE

Pour connecter des fournisseurs d'identité externes (IdP) à votre environnement de gestion des identités (IdM), créez des références IdP dans IdM. Ces exemples montrent comment configurer les références aux fournisseurs d'identité externes en fonction des différents modèles IdP. Utilisez les options suivantes pour spécifier vos paramètres :

--provider

le modèle prédéfini pour l'un des fournisseurs d'identité connus

--client-id

l'identifiant du client OAuth 2.0 émis par l'IdP lors de l'enregistrement de l'application. La procédure d'enregistrement de l'application étant spécifique à chaque IdP, reportez-vous à leur documentation pour plus de détails. Si l'IdP externe est Red Hat Single Sign-On (SSO), voir Création d'un client OpenID Connect.

--base-url

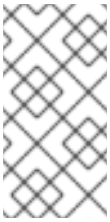
uRL de base pour les modèles IdP, requis par Keycloak et Okta

--organization

ID de domaine ou d'organisation de l'IdP, requis par Microsoft Azure

--secret

(optional) Utilisez cette option si vous avez configuré votre IdP externe pour qu'il demande un secret aux clients OAuth 2.0 confidentiels. Si vous utilisez cette option lors de la création d'une référence IdP, le secret vous est demandé de manière interactive. Protéger le secret du client sous forme de mot de passe.



NOTE

SSSD dans RHEL 9.1 prend uniquement en charge les clients OAuth 2.0 non confidentiels qui n'utilisent pas de secret client. Si vous souhaitez utiliser des IdP externes qui exigent un secret client pour les clients confidentiels, vous devez utiliser SSSD dans RHEL 9.2 et les versions ultérieures.

Conditions préalables

- Vous avez enregistré IdM en tant qu'application OAuth auprès de votre IdP externe et obtenu un identifiant client.
- Vous pouvez vous authentifier en tant que compte administrateur IdM.
- Vos serveurs IdM utilisent RHEL 9.1 ou une version ultérieure.
- Vos serveurs IdM utilisent SSSD 2.7.0 ou une version ultérieure.

Procédure

1. S'authentifier en tant qu'administrateur IdM sur un serveur IdM.

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

2. Créer une référence à l'IdP requis dans IdM comme indiqué dans le tableau suivant.

Identity Provider	Options importantes	Exemple de commande
Microsoft Identity Platform, Azure AD	**--provider microsoft --organization**	# ipa idp-add *my-azure-idp* \ **--provider microsoft** \ --organization *main* \ --client-id *<azure_client_id>*
Google	**--provider google**	# ipa idp-add *my-google-idp* \ **--provider google** \ --client-id *<google_client_id>*
GitHub	**--provider github**	# ipa idp-add my-github-idp \ **--provider github** \ --client-id *<github_client_id>*

Identity Provider	Options importantes	Exemple de commande
Keycloak, Red Hat Single Sign-On	**--provider keycloak** **--organization** **--base-url**	# ipa idp-add my-keycloak-idp \ **--provider keycloak** \ --organization main \ --base-url *keycloak.idm.example.com:8443/auth* \ --client-id *<keycloak_client_id>* **NOTE** La version Quarkus de Keycloak 17 et les versions ultérieures ont supprimé la partie **/auth/** de l'URI. Si vous utilisez la distribution non Quarkus de Keycloak dans votre déploiement, incluez **/auth/** dans l'option **--base-url**.
Okta	**--provider okta**	# ipa idp-add my-okta-idp \ **--provider okta** --base-url *dev-12345.okta.com* \ --client-id *<okta_client_id>*

Par exemple, la commande suivante crée une référence appelée **my-keycloak-idp** à un IdP basé sur le modèle Keycloak, où l'option **--base-url** spécifie l'URL du serveur Keycloak au format **server-name.$DOMAIN:$PORT/prefix**.

```
[root@server ~]# ipa idp-add my-keycloak-idp \
                --provider keycloak --organization main \
                --base-url keycloak.idm.example.com:8443/auth \
                --client-id id13778
----------------------------------------------
Added Identity Provider server "my-keycloak-idp"
----------------------------------------------
  Identity Provider server name: my-keycloak-idp
  Authorization URI:
https://keycloak.idm.example.com:8443/auth/realms/main/protocol/openid-connect/auth
  Device authorization URI:
https://keycloak.idm.example.com:8443/auth/realms/main/protocol/openid-
connect/auth/device
  Token URI: https://keycloak.idm.example.com:8443/auth/realms/main/protocol/openid-
connect/token
  User info URI: https://keycloak.idm.example.com:8443/auth/realms/main/protocol/openid-
connect/userinfo
  Client identifier: ipa_oidc_client
  Scope: openid email
  External IdP user identifier attribute: email
```

Vérification

Vérification

- Vérifiez que la sortie de la commande **ipa idp-show** indique la référence IdP que vous avez créée.

```
[root@server ~]# ipa idp-show my-keycloak-idp
```

Ressources supplémentaires

- Liste des modèles pour les fournisseurs d'identité externes
- **ipa help idp-add** sortie de commande

49.3. GESTION DES RÉFÉRENCES À DES IDP EXTERNES

Après avoir créé une référence à un fournisseur d'identité externe (IdP), vous pouvez rechercher, afficher, modifier et supprimer cette référence. Cet exemple vous montre comment gérer une référence à un fournisseur d'identité externe nommé **keycloak-server1**.

Conditions préalables

- Vous pouvez vous authentifier en tant que compte administrateur IdM.
- Vos serveurs IdM utilisent RHEL 9.1 ou une version ultérieure.
- Vos serveurs IdM utilisent SSSD 2.7.0 ou une version ultérieure.
- Vous avez créé une référence à un IdP dans IdM. Voir Création d'une référence à un fournisseur d'identité externe.

Procédure

1. S'authentifier en tant qu'administrateur IdM sur un serveur IdM.

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

2. Gérer la référence IdP.
   - Pour trouver une référence IdP dont l'entrée comprend la chaîne **keycloak**:

```
[root@server ~]# ipa idp-find keycloak
```

   - Pour afficher une référence IdP nommée **my-keycloak-idp**:

```
[root@server ~]# ipa idp-show my-keycloak-idp
```

   - Pour modifier une référence IdP, utiliser la commande **ipa idp-mod**. Par exemple, pour modifier le secret d'une référence IdP nommée **my-keycloak-idp**, spécifiez l'option **--secret** pour être invité à saisir le secret :

```
[root@server ~]# ipa idp-mod my-keycloak-idp --secret
```

   - Pour supprimer une référence IdP nommée **my-keycloak-idp**:

```
[root@server ~]# ipa idp-del my-keycloak-idp
```

49.4. PERMETTRE À UN UTILISATEUR IDM DE S'AUTHENTIFIER VIA UN IDP EXTERNE

Pour permettre à un utilisateur IdM de s'authentifier via un fournisseur d'identité externe (IdP), associez la référence IdP externe que vous avez précédemment créée au compte de l'utilisateur. Cet exemple associe la référence IdP externe **keycloak-server1** à l'utilisateur **external-idp-user**.

Conditions préalables

- Votre client IdM et vos serveurs IdM utilisent RHEL 9.1 ou une version ultérieure.
- Votre client IdM et vos serveurs IdM utilisent SSSD 2.7.0 ou une version ultérieure.
- Vous avez créé une référence à un IdP dans IdM. Voir Création d'une référence à un fournisseur d'identité externe.

Procédure

- Modifier l'entrée utilisateur IdM pour associer une référence IdP au compte utilisateur :

```
[root@server ~]# ipa user-mod external-idp-user \
                 --idp my-keycloak-idp \
                 --idp-user-id external-idp-user@idm.example.com \
                 --user-auth-type=idp
--------------------------------
Modified user "external-idp-user"
--------------------------------
  User login: external-idp-user
  First name: Test
  Last name: User1
  Home directory: /home/external-idp-user
  Login shell: /bin/sh
  Principal name: external-idp-user@idm.example.com
  Principal alias: external-idp-user@idm.example.com
  Email address: external-idp-user@idm.example.com
  UID: 35000003
  GID: 35000003
  User authentication types: idp
  External IdP configuration: keycloak
  External IdP user identifier: external-idp-user@idm.example.com
  Account disabled: False
  Password: False
  Member of groups: ipausers
  Kerberos keys available: False
```

Vérification

- Vérifiez que la sortie de la commande **ipa user-show** pour cet utilisateur affiche des références à l'IdP :

```
[root@server ~]# ipa user-show external-idp-user
```

```
User login: external-idp-user
First name: Test
Last name: User1
Home directory: /home/external-idp-user
Login shell: /bin/sh
Principal name: external-idp-user@idm.example.com
Principal alias: external-idp-user@idm.example.com
Email address: external-idp-user@idm.example.com
ID: 35000003
GID: 35000003
User authentication types: idp
External IdP configuration: keycloak
External IdP user identifier: external-idp-user@idm.example.com
Account disabled: False
Password: False
Member of groups: ipausers
Kerberos keys available: False
```

49.5. RÉCUPÉRATION D'UN TICKET IDM EN TANT QU'UTILISATEUR IDP

Pour récupérer un ticket Kerberos (TGT) en tant qu'utilisateur auprès d'un fournisseur d'identité externe (IdP), demander un ticket Kerberos anonyme et activer le canal Flexible Authentication via Secure Tunneling (FAST) afin de fournir une connexion sécurisée entre le client Kerberos et le Centre de distribution Kerberos (KDC).

Conditions préalables

- Votre client IdM et vos serveurs IdM utilisent RHEL 9.1 ou une version ultérieure.
- Votre client IdM et vos serveurs IdM utilisent SSSD 2.7.0 ou une version ultérieure.
- Vous avez créé une référence à un IdP dans IdM. Voir Création d'une référence à un fournisseur d'identité externe.
- Vous avez associé une référence IdP externe au compte utilisateur. Voir Permettre à un utilisateur IdM de s'authentifier via un IdP externe.

Procédure

1. Utilisez Anonymous PKINIT pour obtenir un ticket Kerberos et le stocker dans un fichier nommé **./fast.ccache**.

```
[root@client ~]# kinit -n -c ./fast.ccache
```

2. Commencez à vous authentifier en tant qu'utilisateur, en utilisant l'option **-T** pour activer le canal de communication FAST.

```
[root@client ~]# kinit -T ./fast.ccache external-idp-user
Authenticate at https://oauth2.idp.com:8443/auth/realms/master/device?user_code=YHMQ-XKTL and press ENTER.:
```

3. Dans un navigateur, authentifiez-vous en tant qu'utilisateur sur le site web indiqué dans la sortie de la commande.

4. Dans la ligne de commande, appuyez sur la touche **Entrée** pour terminer le processus d'authentification.

Vérification

- Affichez les informations de votre ticket Kerberos et confirmez que la ligne **config: pa_type** indique **152** pour la préauthentification avec un IdP externe.

```
[root@client ~]# klist -C
Ticket cache: KCM:0:58420
Default principal: external-idp-user@IDM.EXAMPLE.COM

Valid starting     Expires            Service principal
05/09/22 07:48:23  05/10/22 07:03:07  krbtgt/IDM.EXAMPLE.COM@IDM.EXAMPLE.COM
config: fast_avail(krbtgt/IDM.EXAMPLE.COM@IDM.EXAMPLE.COM) = yes
08/17/2022 20:22:45  08/18/2022 20:22:43
krbtgt/IDM.EXAMPLE.COM@IDM.EXAMPLE.COM
config: pa_type(krbtgt/IDM.EXAMPLE.COM@IDM.EXAMPLE.COM) = 152
```

49.6. CONNEXION À UN CLIENT IDM VIA SSH EN TANT QU'UTILISATEUR IDP

Pour se connecter à un client IdM via SSH en tant qu'utilisateur d'un fournisseur d'identité externe (IdP), commencez le processus de connexion sur la ligne de commande. Lorsque vous y êtes invité, effectuez le processus d'authentification sur le site Web associé au fournisseur d'identité et terminez le processus sur le client de gestion d'identité (IdM).

Conditions préalables

- Votre client IdM et vos serveurs IdM utilisent RHEL 9.1 ou une version ultérieure.
- Votre client IdM et vos serveurs IdM utilisent SSSD 2.7.0 ou une version ultérieure.
- Vous avez créé une référence à un IdP dans IdM. Voir Création d'une référence à un fournisseur d'identité externe.
- Vous avez associé une référence IdP externe au compte utilisateur. Voir Permettre à un utilisateur IdM de s'authentifier via un IdP externe.

Procédure

1. Tentative de connexion au client IdM via SSH.

```
[user@client ~]$ ssh external-idp-user@client.idm.example.com
(external-idp-user@client.idm.example.com) Authenticate at
https://oauth2.idp.com:8443/auth/realms/main/device?user_code=XYFL-ROYR and press
ENTER.
```

2. Dans un navigateur, authentifiez-vous en tant qu'utilisateur sur le site web indiqué dans la sortie de la commande.
3. Dans la ligne de commande, appuyez sur la touche **Entrée** pour terminer le processus d'authentification.

Vérification

- Affichez les informations de votre ticket Kerberos et confirmez que la ligne **config: pa_type** indique **152** pour la préauthentification avec un IdP externe.

```
[external-idp-user@client ~]$ klist -C
Ticket cache: KCM:0:58420
Default principal: external-idp-user@IDM.EXAMPLE.COM

Valid starting     Expires            Service principal
05/09/22 07:48:23  05/10/22 07:03:07  krbtgt/IDM.EXAMPLE.COM@IDM.EXAMPLE.COM
config: fast_avail(krbtgt/IDM.EXAMPLE.COM@IDM.EXAMPLE.COM) = yes
08/17/2022 20:22:45  08/18/2022 20:22:43
krbtgt/IDM.EXAMPLE.COM@IDM.EXAMPLE.COM
config: pa_type(krbtgt/IDM.EXAMPLE.COM@IDM.EXAMPLE.COM) = 152
```

49.7. LISTE DES MODÈLES POUR LES FOURNISSEURS D'IDENTITÉ EXTERNES

Les fournisseurs d'identité (IdP) suivants prennent en charge le flux d'autorisation des dispositifs OAuth 2.0 :

- Plate-forme d'identité Microsoft, y compris Azure AD
- Google
- GitHub
- Keycloak, y compris Red Hat Single Sign-On (SSO)
- Okta

Lorsque vous utilisez la commande **ipa idp-add** pour créer une référence à l'un de ces IdP externes, vous pouvez spécifier le type d'IdP avec l'option **--provider**, qui se développe en options supplémentaires comme décrit ci-dessous :

--provider=microsoft

Les IdP Microsoft Azure permettent un paramétrage basé sur l'identifiant du locataire Azure, que vous pouvez spécifier avec l'option **--organization** de la commande **ipa idp-add**. Si vous avez besoin de la prise en charge de l'IdP live.com, spécifiez l'option **--organization common**.
Le choix de l'option **--provider=microsoft** permet d'utiliser les options suivantes. La valeur de l'option **--organization** remplace la chaîne **${ipaidporg}** dans le tableau.

Option	Valeur
--auth-uri=URI	**https://login.microsoftonline.com/${ipaidporg}/oauth2/v2.0/authorize**
--dev-auth-uri=URI	**https://login.microsoftonline.com/${ipaidporg}/oauth2/v2.0/devicecode**
--token-uri=URI	**https://login.microsoftonline.com/${ipaidporg}/oauth2/v2.0/token**

Option	Valeur
--userinfo-uri=URI	**https://graph.microsoft.com/oidc/userinfo**
--keys-uri=URI	**https://login.microsoftonline.com/common/discovery/v2.0/keys**
--scope=STR	**openid email**
--idp-user-id=STR	**email**

--provider=google

Le choix de **--provider=google** permet d'utiliser les options suivantes :

Option	Valeur
--auth-uri=URI	**https://accounts.google.com/o/oauth2/auth**
--dev-auth-uri=URI	**https://oauth2.googleapis.com/device/code**
--token-uri=URI	**https://oauth2.googleapis.com/token**
--userinfo-uri=URI	**https://openidconnect.googleapis.com/v1/userinfo**
--keys-uri=URI	**https://www.googleapis.com/oauth2/v3/certs**
--scope=STR	**openid email**
--idp-user-id=STR	**email**

--provider=github

Le choix de **--provider=github** permet d'utiliser les options suivantes :

Option	Valeur
--auth-uri=URI	**https://github.com/login/oauth/authorize**
--dev-auth-uri=URI	**https://github.com/login/device/code**
--token-uri=URI	**https://github.com/login/oauth/access_token**
--userinfo-uri=URI	**https://openidconnect.googleapis.com/v1/userinfo**
--keys-uri=URI	**https://api.github.com/user**
--scope=STR	**user**

Option	Valeur
--idp-user-id=STR	**login**

--provider=keycloak

Avec Keycloak, vous pouvez définir plusieurs domaines ou organisations. Comme il s'agit souvent d'une partie d'un déploiement personnalisé, l'URL de base et l'ID de domaine sont tous deux nécessaires, et vous pouvez les spécifier avec les options **--base-url** et **--organization** de la commande **ipa idp-add**:

```
[root@client ~]# ipa idp-add MySSO --provider keycloak \
  --org main --base-url keycloak.domain.com:8443/auth \
  --client-id <your-client-id>
```

L'option **--provider=keycloak** permet d'utiliser les options suivantes. La valeur que vous indiquez dans l'option **--base-url** remplace la chaîne **${ipaidpbaseurl}** dans le tableau, et la valeur que vous indiquez pour l'option **--organization `option replaces the string `${ipaidporg}**.

Option	Valeur
--auth-uri=URI	**https://${ipaidpbaseurl}/realms/${ipaidporg}/protocol/openid-connect/auth**
--dev-auth-uri=URI	**https://${ipaidpbaseurl}/realms/${ipaidporg}/protocol/openid-connect/auth/device**
--token-uri=URI	**https://${ipaidpbaseurl}/realms/${ipaidporg}/protocol/openid-connect/token**
--userinfo-uri=URI	**https://${ipaidpbaseurl}/realms/${ipaidporg}/protocol/openid-connect/userinfo**
--scope=STR	**openid email**
--idp-user-id=STR	**email**

--provider=okta

Après avoir enregistré une nouvelle organisation dans Okta, une nouvelle URL de base lui est associée. Vous pouvez spécifier cette URL de base avec l'option **--base-url** de la commande **ipa idp-add**:

```
[root@client ~]# ipa idp-add MyOkta --provider okta --base-url dev-12345.okta.com --client-id
<your-client-id>
```

Le choix de l'option **--provider=okta** permet d'utiliser les options suivantes. La valeur que vous spécifiez pour l'option **--base-url** remplace la chaîne **${ipaidpbaseurl}** dans le tableau.

Option	Valeur
--auth-uri=URI	**https://${ipaidpbaseurl}/oauth2/v1/authorize**
--dev-auth-uri=URI	**https://${ipaidpbaseurl}/oauth2/v1/device/authorize**
--token-uri=URI	**https://${ipaidpbaseurl}/oauth2/v1/token**
--userinfo-uri=URI	**https://${ipaidpbaseurl}/oauth2/v1/userinfo**
--scope=STR	**openid email**
--idp-user-id=STR	**email**

Ressources supplémentaires

- Modèles d'IdP pré-remplis