Red Hat Directory Server 12

ディレクトリーの属性と値の管理

ldapadd、ldapmodify、ldapdelete、dscof ユーティリティーまたは Web コンソールを使用したディレクトリーエントリーの管理

Last Updated: 2024-11-05

Red Hat Directory Server 12 ディレクトリーの属性と値の管理

ldapadd、ldapmodify、ldapdelete、dscof ユーティリティーまたは Web コンソールを使用したディレクトリーエントリーの管理

法律上の通知

概要

openldap-clients パッケージのツールを使用して Directory Server エントリーを管理する方法を説明します。属性の一意性を強制し、サービスクラス (CoS) を割り当ててエントリー管理を簡略化し、ストレージ要件を削減し、レプリケーションの競合を回避します。

目次

RED HAT DIRECTORY SERVER に関するフィードバックの提供

Red Hat のドキュメントおよび製品に関するご意見をお待ちしております。ドキュメントの改善点があればお知らせください。改善点を報告する場合は、以下のように行います。

- Jira を通じて Red Hat Directory Server ドキュメントに関するフィードバックを送信する場合 (アカウントが必要):
  1. Red Hat Issue Tracker にアクセスしてください。
  2. **Summary** フィールドにわかりやすいタイトルを入力します。
  3. **Description** フィールドに、ドキュメントの改善に関するご意見を記入してください。ドキュメントの該当部分へのリンクも追加してください。
  4. ダイアログの下部にある **Create** をクリックします。
- Jira を通じて Red Hat Directory Server 製品に関するフィードバックを送信する場合 (アカウントが必要):
  1. Red Hat Issue Tracker にアクセスしてください。
  2. **Create Issue** ページで、**Next** をクリックします。
  3. **Summary** フィールドに入力します。
  4. **Component** フィールドでコンポーネントを選択します。
  5. **Description** フィールドに以下の内容を入力します。
     a. 選択したコンポーネントのバージョン番号。
     b. 問題を再現するための手順、または改善のための提案。
  6. **Create** をクリックします。

第1章 コマンドラインを使用したディレクトリーエントリーの管理

コマンドラインを使用して、LDAP エントリーを追加、編集、名前変更、および削除できます。

1.1. LDAPADD、LDAPMODIFY、および LDAPDELETE ユーティリティーへの入力の指定

ディレクトリーのエントリーまたは属性を追加、更新、または削除する場合は、ユーティリティーのインタラクティブモードを使用して LDAP Data Interchange Format (LDIF) ステートメントに入るか、LDIF ファイルを渡すことができます。

1.1.1. OpenLDAP クライアントユーティリティーのインタラクティブモード

インタラクティブモードでは、**ldapadd**、**ldapmodify**、および **ldapdelete** ユーティリティーはコマンドラインから入力を読み取ります。インタラクティブモードを終了するには、**Ctrl**+**D** (^D) のキーの組み合わせを押して end-of-file (EOF) エスケープシーケンスを送信します。

インタラクティブモードでは、ユーティリティーは、**Enter** を 2 回押したときに、または EOF シーケンスを送信するときに、ステートメントを LDAP サーバーに送信します。

対話型モードを使用します。

- ファイルを作成せずに LDAP Data Interchange Format (LDIF) ステートメントに入るには、以下を行います。

  例1.1 ldapmodify インタラクティブモードを使用した LDIF ステートメントの開始

  以下の例では、**ldapmodify** を対話モードで実行し、**telephoneNumber** 属性を削除して、**cn=manager_name,ou=people,dc=example,dc=com** の値の **manager** 属性を **uid=user,ou=people,dc=example,dc=com** エントリーに追加します。最後のステートメントの後に **Ctrl+D** を押して、インタラクティブモードを終了します。

  ```
  # ldapmodify -D "cn=Directory Manager" -W -H ldap://server.example.com -x

  dn: uid=user,ou=people,dc=example,dc=com
  changetype: modify
  delete: telephoneNumber
  -
  add: manager
  manager: cn=manager_name,ou=people,dc=example,dc=com

  modifying entry "uid=user,ou=people,dc=example,dc=com"

  ^D
  ```

- 別のコマンドによって出力される LDIF ステートメントをサーバーにリダイレクトするには、次のコマンドを実行します。

  例1.2 リダイレクトされたコンテンツでの ldapmodify インタラクティブモードの使用

  以下の例では、**command_that_outputs_LDIF** コマンドの出力を **ldapmodify** にリダイレクトします。対話モードは、リダイレクトされたコマンドの終了後に自動的に終了します。

```
# command_that_outputs_LDIF | ldapmodify -D "cn=Directory Manager" -W -H
ldap://server.example.com -x
```

関連情報

- **ldif(5)** の man ページ

1.1.2. OpenLDAP クライアントユーティリティーのファイルモード

インタラクティブモードでは、**ldapadd**、**ldapmodify**、および **ldapdelete** ユーティリティーは、ファイルから LDAP Data Interchange Format (LDIF) ステートメントを読み取ります。このモードを使用して、より多くの LDIF ステートメントをサーバーに送信します。

例1.3 LDIF ステートメントを持つファイルを ldapmodify に渡す

1. LDIF ステートメントでファイルを作成します。たとえば、以下のステートメントで **~/example.ldif** ファイルを作成します。

```
dn: uid=user,ou=people,dc=example,dc=com
changetype: modify
delete: telephoneNumber
-
add: manager
manager: cn=manager_name,ou=people,dc=example,dc=com
```

この例では、**telephoneNumber** 属性を削除し、**cn=manager_name,ou=people,dc=example,dc=com** 値を持つ **manager** 属性を **uid=user,ou=people,dc=example,dc=com** エントリーに追加します。

2. **-f** パラメーターを使用して、ファイルを **ldapmodify** コマンドに渡します。

```
# ldapmodify -D "cn=Directory Manager" -W -H ldap://server.example.com -x -f
~/example.ldif
```

関連情報

- **ldif(5)** の man ページ

1.1.3. OpenLDAP クライアントユーティリティーの連続動作モード

デフォルトでは、複数の LDAP Data Interchange Format (LDIF) ステートメントをサーバーに送信し、1 つの操作が失敗すると、プロセスが停止します。ただし、エラーが発生する前に処理されるエントリーは、正常に追加、変更、または削除されています。

エラーを無視してバッチでさらに LDIF ステートメントの処理を続けるには、**-c** パラメーターを **ldapadd** および **ldapmodify** に渡します。

```
# ldpamodify -c -D "cn=Directory Manager" -W -H ldap://server.example.com -x
```

1.2. コマンドラインを使用した LDAP エントリーの追加

新しいエントリーをディレクトリーに追加するには、**ldapadd** ユーティリティーまたは **ldapmodify** ユーティリティーを使用します。**/bin/ldapadd** は **/bin/ldapmodify** へのシンボリックリンクであることに注意してください。そのため、**ldapadd** は **ldapmodify -a** と同じ操作を実行します。

注記

親エントリーがすでに存在する場合のみ、新しいディレクトリーエントリーを追加できます。たとえば、**ou=people,dc=example,dc=com** の親エントリーが存在しない場合は、**cn=user,ou=people,dc=example,dc=com** エントリーを追加できません。

1.2.1. ldapadd を使用したエントリーの追加

ldapadd ユーティリティーを使用して、たとえば **cn=user,ou=people,dc=example,dc=com** ユーザーエントリーを追加するには、以下を実行します。

```
# ldapadd -D "cn=Directory Manager" -W -H ldap://server.example.com -x

dn: uid=user,ou=People,dc=example,dc=com
uid: user
givenName: given_name
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetorgperson
sn: surname
cn: user
```

注記

ldapadd を実行すると、**changetype: add** 操作が自動的に実行されます。そのため、LDIF ステートメントで **changetype: add** を指定する必要はありません。

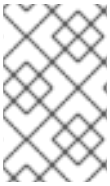
関連情報

- **ldapadd(1)** man ページ

1.2.2. ldapmodify を使用したエントリーの追加

ldapmodify ユーティリティーを使用して、たとえば **cn=user,ou=people,dc=example,dc=com** ユーザーエントリーを追加するには、以下を実行します。

```
# ldapmodify -a -D "cn=Directory Manager" -W -H ldap://server.example.com -x

dn: uid=user,ou=People,dc=example,dc=com
uid: user
givenName: given_name
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetorgperson
sn: surname
cn: user
```



注記

-a パラメーターを **ldapmodify** コマンドに渡すと、ユーティリティーは **changetype: add** 操作を自動的に実行します。そのため、LDIF ステートメントで **changetype: add** を指定する必要はありません。

関連情報

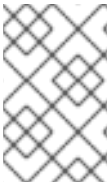
- **ldapmodify(1)** man ページ

1.2.3. データベース接尾辞のルートエントリーの作成

dc=example,dc=com などのデータベース接尾辞のルートエントリーを作成するには、**cn=Directory Manager** ユーザーとしてバインドし、エントリーを追加します。識別名 (DN) は、データベースのルートまたは従属接尾辞の DN に対応します。

たとえば、**dc=example,dc=com** 接尾辞を追加するには、次のコマンドを実行します。

```
# ldapmodify -D "cn=Directory Manager" -W -H ldap://server.example.com -x

dn: dc=example,dc=com
changetype: add
objectClass: top
objectClass: domain
dc: example
```



注記

ルートオブジェクトは、接尾辞に 1 つのデータベースがある場合にのみ追加できます。複数のデータベースに保存される接尾辞を作成する場合は、**dsctl ldif2db** コマンドを使用して、新しいエントリーを保持するデータベースを設定する必要があります。

関連情報

- サーバーのオフライン時にコマンドラインを使用したデータのインポート

1.3. コマンドラインを使用した LDAP エントリーの更新

ディレクトリーエントリーを変更する場合は、**changetype: modify** ステートメントを使用します。change 操作に応じて、エントリーから属性を追加、変更、または削除できます。

1.3.1. LDAP エントリーへの属性の追加

LDAP エントリーに属性を追加するには、**add** 操作を使用します。

たとえば、**555-1234567** の値を持つ **telephoneNumber** 属性を **uid=user,ou=People,dc=example,dc=com** エントリーに追加するには、以下を実行します。

```
# ldapmodify -D "cn=Directory Manager" -W -H ldap://server.example.com -x

dn: uid=user,ou=People,dc=example,dc=com
```

```
changetype: modify
add: telephoneNumber
telephoneNumber: 555-1234567
```

属性が多値である場合、属性名を複数回指定して、1 つの操作ですべての値を追加できます。たとえば、2 つの **telephoneNumber** 属性を一度に **uid=user,ou=People,dc=example,dc=com** に追加するには、次のように入力します。

```
# ldapmodify -D "cn=Directory Manager" -W -H ldap://server.example.com -x

dn: uid=user,ou=People,dc=example,dc=com
changetype: modify
add: telephoneNumber
telephoneNumber: 555-1234567
telephoneNumber: 555-7654321
```

1.3.2. 属性値の更新

属性の値を更新する手順は、属性が単値であるか多値であるかによって異なります。

- 単値属性の更新
  単値属性を更新する場合は、**replace** 操作を使用して既存の値を上書きします。次のコマンドは、**uid=user,ou=People,dc=example,dc=com** エントリーの **manager** 属性を更新します。

  ```
  # ldapmodify -D "cn=Directory Manager" -W -H ldap://server.example.com -x

  dn: uid=user,ou=People,dc=example,dc=com
  changetype: modify
  replace: manager
  manager: uid=manager_name,ou=People,dc=example,dc=com
  ```

- 多値属性の特定値の更新
  多値属性の特定の値を更新するには、最初に置き換えるエントリーを削除してから、新しい値を追加します。次のコマンドは、**uid=user,ou=People,dc=example,dc=com** エントリーで現在 **555-1234567** に設定されている **telephoneNumber** 属性のみを更新します。

  ```
  # ldapmodify -D "cn=Directory Manager" -W -H ldap://server.example.com -x

  dn: uid=user,ou=People,dc=example,dc=com
  changetype: modify
  delete: telephoneNumber
  telephoneNumber: 555-1234567
  -
  add: telephoneNumber
  telephoneNumber: 555-9876543
  ```

1.3.3. エントリーからの属性の削除

エントリーから属性を削除するには、**delete** 操作を実行します。

- 属性の削除
  たとえば、**uid=user,ou=People,dc=example,dc=com** エントリーから **manager** 属性を削除するには、次のように入力します。

```
# ldapmodify -D "cn=Directory Manager" -W -H ldap://server.example.com -x

dn: uid=user,ou=People,dc=example,dc=com
changetype: modify
delete: manager
```

重要

属性に複数の値が含まれる場合、この操作によりすべての値が削除されます。

- 多値属性から特定の値の削除
  複数値属性から特定の値を削除する場合は、LDAP Data Interchange Format (LDIF) ステートメントに属性とその値をリストします。たとえば、**uid=user,ou=People,dc=example,dc=com** エントリーから **555-1234567** に設定されている **telephoneNumber** 属性だけを削除するには、以下のコマンドを実行します。

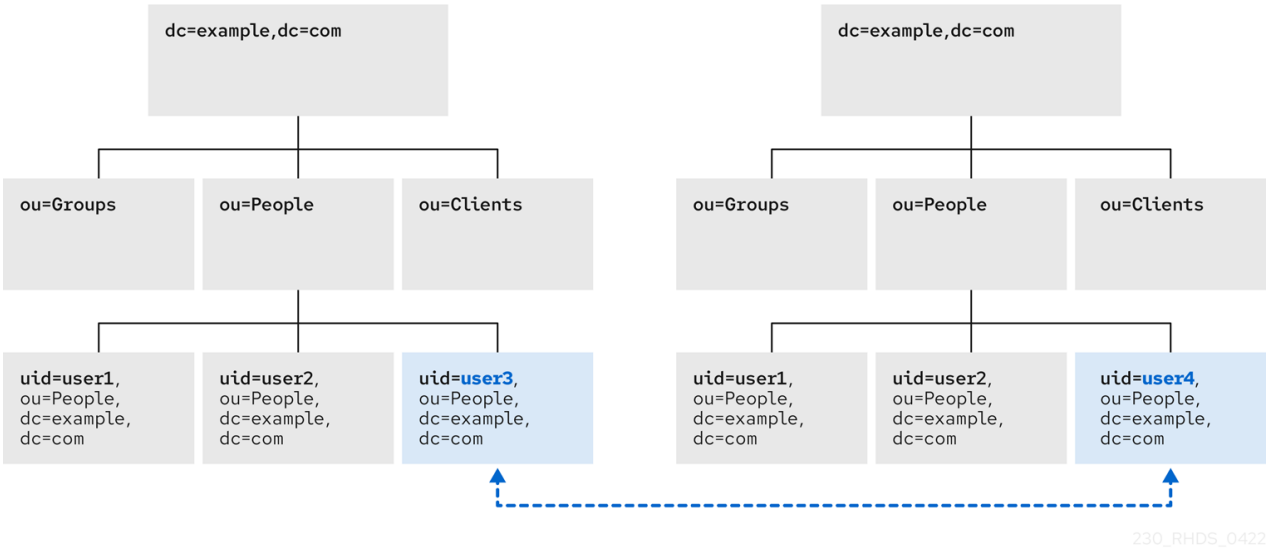
```
# ldapmodify -D "cn=Directory Manager" -W -H ldap://server.example.com -x

dn: uid=user,ou=People,dc=example,dc=com
changetype: modify
delete: telephoneNumber
telephoneNumber: 555-1234567
```

1.4. LDAP エントリーの名前変更と移動

The following rename operations exist:

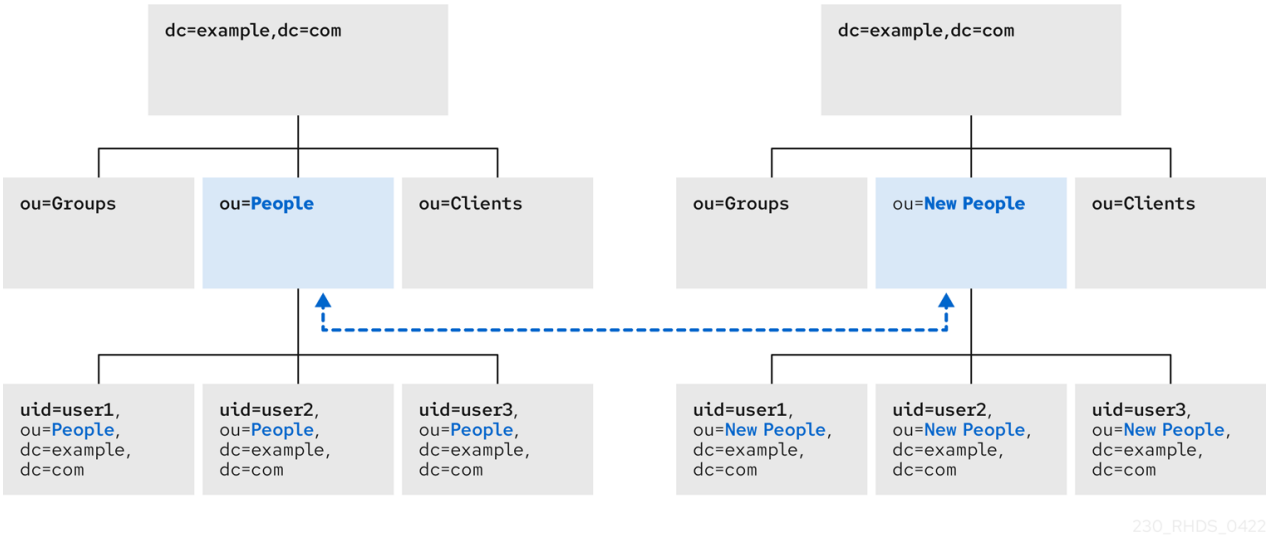
エントリーの名前変更

エントリーの名前を変更すると、**modrdn** 操作はエントリーの RDN (Relative Distinguished Name) を変更します。



サブエントリーの名前変更

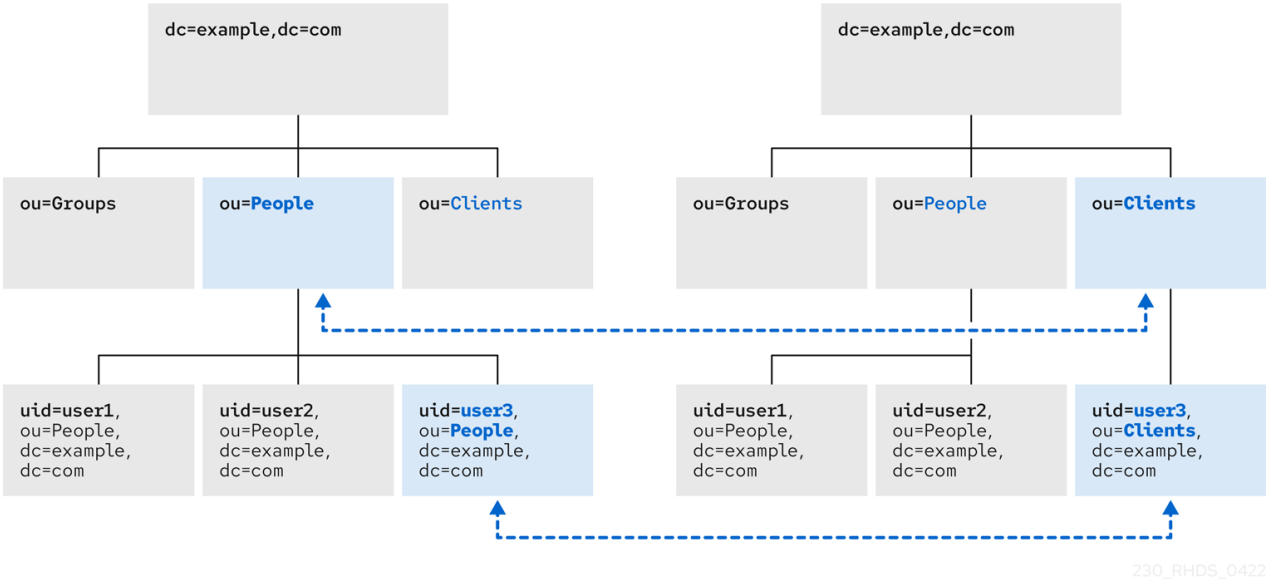
サブツリーエントリーの場合、**modrdn** 操作はサブツリーと子エントリーの DN コンポーネントの名前を変更します。



大規模なサブツリーでは、このプロセスに多くの時間とリソースが必要になる可能性があることに注意してください。

エントリーの新しい親への移動

サブツリーの名前を変更する同様のアクションは、エントリーをあるサブツリーから別のサブツリーに移動することです。これは、**modrdn** 操作の拡張タイプで、エントリーの名前を同時に変更し、**newSuperior** 属性を設定して、エントリーを別の親に移動します。



1.4.1. LDAP エントリーの名前を変更する際の考慮事項

名前変更の操作を実行する場合は、以下の点に留意してください。

- root 接尾辞の名前を変更することはできません。
- サブツリー名前変更操作によるレプリケーションへの影響は最小限に抑えられます。レプリカ合意は、データベースのサブツリーではなく、データベース全体に適用されます。そのため、サブツリーの名前変更操作ではレプリカ合意の再設定は必要ありません。サブツリーの名前変更操作後のすべての名前の変更は、通常どおり複製されます。

- サブツリーの名前を変更し、同期合意を再設定する必要がある場合があります。同期合意は、接尾辞またはサブツリーレベルで設定されます。そのため、サブツリーの名前を変更すると、同期が中断する可能性があります。

- サブツリーの名前を変更するには、サブツリーに設定されたサブツリーレベルのアクセス制御命令 (ACI) を手動で再設定し、サブツリーの子エントリーに設定されたエントリーレベルの ACI を手動で再設定する必要があります。

- **ou** から **dc** への移行など、サブツリーのコンポーネントを変更しようとすると、スキーマ違反で失敗する可能性があります。たとえば、**organizationalUnit** オブジェクトクラスには **ou** 属性が必要です。この属性がサブツリーの名前の一部として削除されると、操作は失敗します。

- グループを移動すると、**MemberOf** プラグインは **memberOf** 属性を自動的に更新します。ただし、グループが含まれるサブツリーを移動する場合は、**cn=memberof** タスクエントリーでタスクを手動で作成するか、**dsconf memberof fixup** コマンドを使用して関連する **memberOf** 属性を更新する必要があります。

1.4.2. エントリーの名前を変更するときの相対的な識別名の動作の制御

エントリーの名前を変更すると、**deleteOldRDN** 属性は、古い相対識別名 (RDN) を削除するか保持するかを制御します。

deleteOldRDN: 0

既存の RDN は、新しいエントリーの値として保持されます。生成されるエントリーには、古い属性と新しい共通名 (CN) を持つ 2 つの **cn** 属性が含まれます。
たとえば、以下の属性は、**deleteOldRDN** 属性を **0** に設定して、**cn=old_group,dc=example,dc=com** から **cn=new_group,dc=example,dc=com** に名前を変更したグループに属しています。

```
dn: cn=new_group,ou=Groups,dc=example,dc=com
objectClass: top
objectClass: groupOfUniqueNames
cn: old_group
cn: new_group
```

deleteOldRDN: 1

Directory Server は古いエントリーを削除し、新しい RDN を使用して新しいエントリーを作成します。新しいエントリーには、新しいエントリーの **cn** 属性のみが含まれます。
たとえば、以下のグループは、**deleteOldRDN** 属性を **1** に設定して、**cn=new_group,dc=example,dc=com** に名前を変更しました。

```
dn: cn=new_group,ou=Groups,dc=example,dc=com
objectClass: top
objectClass: groupofuniquenames
cn: new_group
```

関連情報

- LDAP エントリーまたはサブツリーの名前の変更

1.4.3. LDAP エントリーまたはサブツリーの名前の変更

エントリーまたはサブツリーの名前変更には、**changetype: modrdn** 操作を使用し、**newrdn** 属性に新しい RDN (Relative Distinguished Name) を設定します。

たとえば、**cn=demo1,dc=example,dc=com** エントリーの名前を **cn=demo2,dc=example,dc=com** に変更するには、以下のコマンドを実行します。

```
# ldapmodify -D "cn=Directory Manager" -W -H ldap://server.example.com -x

dn: cn=demo1,dc=example,dc=com
changetype: modrdn
newrdn: cn=demo2
deleteOldRDN: 1
```

関連情報

- エントリーの名前を変更するときの相対的な識別名の動作の制御

1.4.4. LDAP エントリーを新しい親に移動

エントリーを新しい親に移動するには、**changetype: modrdn** 操作を使用して、以下の属性を設定します。

- **newrdn**: 移動したエントリーの相対識別名 (RDN) を設定します。RDN が同じままであっても、このエントリーを設定する必要があります。
- **newSuperior**: 新しい親エントリーの識別名 (DN) を設定します。

たとえば、**cn=demo** エントリーを **ou=Germany,dc=example,dc=com** から **ou=France,dc=example,dc=com** に移動するには、以下のコマンドを実行します。

```
# ldapmodify -D "cn=Directory Manager" -W -H ldap://server.example.com -x

dn: cn=demo,ou=Germany,dc=example,dc=com
changetype: modrdn
newrdn: cn=demo
newSuperior: ou=France,dc=example,dc=com
deleteOldRDN: 1
```

関連情報

- エントリーの名前を変更するときの相対的な識別名の動作の制御

1.5. コマンドラインを使用した LDAP エントリーの削除

LDAP ディレクトリーからエントリーを削除できますが、削除できるのは子エントリーのないエントリーのみです。たとえば、**uid=user,ou=People,dc=example,dc=com** エントリーがまだ存在している場合は、**ou=People,dc=example,dc=com** エントリーを削除できません。

1.5.1. ldapdelete を使用したエントリーの削除

ldapdelete ユーティリティーを使用すると、1 つまたは複数のエントリーを削除できます。たとえば、**uid=user,ou=People,dc=example,dc=com** エントリーを削除するには、次のコマンドを実行します。

-

```
# ldapdelete -D "cn=Directory Manager" -W -H ldap://server.example.com -x
"uid=user,ou=People,dc=example,dc=com"
```

1 つの操作で複数のエントリーを削除するには、それらのエントリーをコマンドに追加します。

```
# ldapdelete -D "cn=Directory Manager" -W -H ldap://server.example.com -x
"uid=user1,ou=People,dc=example,dc=com" "uid=user2,ou=People,dc=example,dc=com"
```

関連情報

- **ldapdelete(1)** man ページ

1.5.2. ldapmodify を使用したエントリーの削除

ldapmodify ユーティリティーを使用してエントリーを削除するには、**changetype: delete** 操作を使用します。たとえば、**uid=user,ou=People,dc=example,dc=com** エントリーを削除するには、次のコマンドを実行します。

```
# ldapmodify -D "cn=Directory Manager" -W -H ldap://server.example.com -x

dn: uid=user,ou=People,dc=example,dc=com
changetype: delete
```

1.6. OPENLDAP クライアントユーティリティーでの特殊文字の使用

コマンドラインを使用する場合は、スペース ( )、アスタリスク (*)、バックスラッシュ (\) などのコマンドラインインタープリターに特別な意味を持つ文字を引用符で囲みます。コマンドラインインタープリターに応じて、一重引用符または二重引用符を使用します。たとえば、**cn=Directory Manager** ユーザーとして認証するには、ユーザーの識別名 (DN) を引用符で囲みます。

```
# ldapmodify -a -D "cn=Directory Manager" -W -H ldap://server.example.com -x
```

また、DN にコンポーネントのコンマが含まれる場合は、バックスラッシュを使用してエスケープします。たとえば、**uid=user,ou=People,dc=example.com Chicago, IL** ユーザーとして認証するには、次のコマンドを実行します。

```
# ldapmodify -a -D "cn=uid=user,ou=People,dc=example.com Chicago\, IL" -W -H
ldap://server.example.com -x
```

1.7. LDIF ステートメントでのバイナリー属性の使用

特定の属性は、**jpegPhoto** 属性などのバイナリー値をサポートします。このような属性を追加または更新すると、ユーティリティーはファイルから属性の値を読み取ります。このような属性を追加または更新するには、**ldapmodify** ユーティリティーを使用できます。

たとえば、**uid=user,ou=People,dc=example,dc=com** エントリーに **jpegPhoto** 属性を追加し、**/home/user_name/photo.jpg** ファイルから属性の値を読み取るには、次のコマンドを実行します。

```
# ldapmodify -D "cn=Directory Manager" -W -H ldap://server.example.com -x
```

```
dn: uid=user,ou=People,dc=example,dc=com
changetype: modify
add: jpegPhoto
jpegPhoto:< file:///home/user_name/photo.jpg
```

重要

: と **<** の間には、スペースがないことに注意してください。

1.8. 国際化されたディレクトリーの LDAP エントリーの更新

属性の値を英語以外の言語で使用するには、属性の値を言語タグに関連付けます。

ldapmodify を使用して言語タグが設定されている属性を更新する場合は、値と言語タグを正確に一致させる必要があります。そうでないと、操作は失敗します。

たとえば、**lang-fr** 言語タグが設定された属性値を変更するには、modify 操作にタグを追加します。

```
# ldapmodify -D "cn=Directory Manager" -W -H ldap://server.example.com -x

dn: uid=user,ou=People,dc=example,dc=com
changetype: modify
replace: homePostalAddress;lang-fr
homePostalAddress;lang-fr: 34 rue de Seine
```

第2章 WEB コンソールを使用したディレクトリーエントリーの管理

Web コンソールを使用して、LDAP エントリーを追加、編集、削除、および名前の変更を実行できます。

2.1. WEB コンソールを使用した LDAP エントリーの追加

Web コンソールを使用して、次のエントリーを作成できます。

- users
- グループ
- roles
- 組織単位 (OU)
- カスタムエントリー

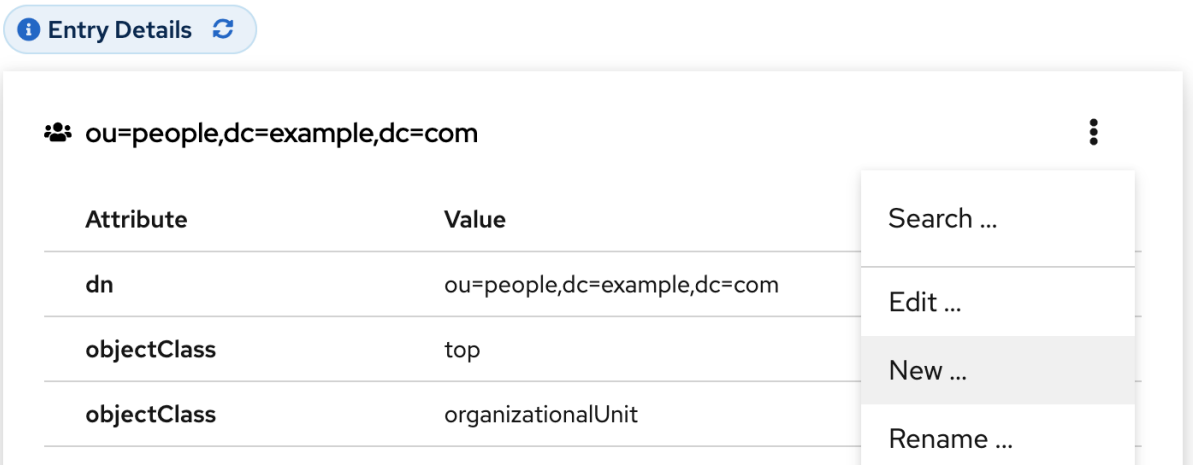
たとえば、POSIX ユーザー **cn=John Smith,ou=people,dc=example,dc=com** をパスワード付きで作成するとします。

前提条件

- ディレクトリーサーバー Web コンソールにログインしている。
- 親エントリーが存在する。たとえば、**ou=people,dc=example,dc=com** になります。

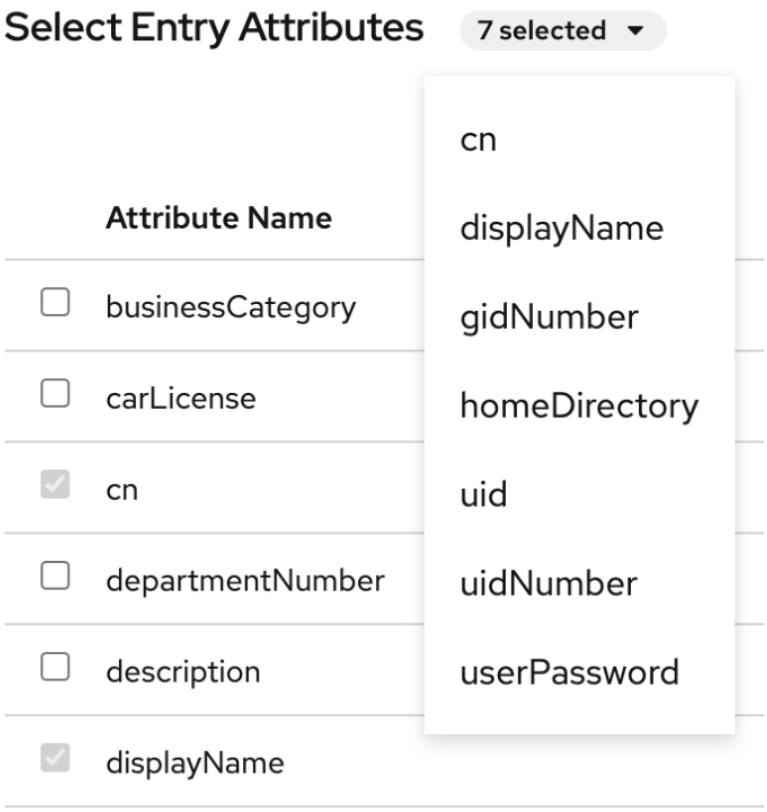
手順

1. **LDAP ブラウザー** メニューを開いて、既存の接尾辞のリストを表示します。
2. **ツリー**ビューまたは **テーブル** ビューを使用して、ユーザーを作成する親エントリー **ou=people,dc=example,dc=com** を展開します。
3. Options menu (⋮) をクリックし、**New** を選択してウィザードウィンドウを開きます。



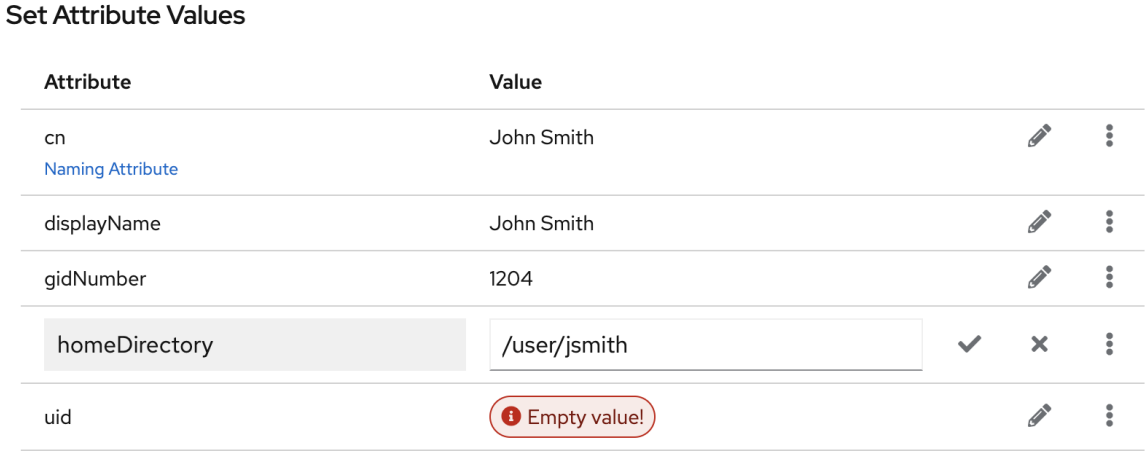
4. **Create a new User** オプションを選択し、**Next** をクリックします。
5. ユーザーエントリーで **Posix Account** タイプを選択し、**Next** をクリックします。

6. オプション: **userPassword** などの追加の属性を選択し、**Next** をクリックします。ステップ名の近くにあるドロップダウンリストを展開すると、選択されたすべての属性を表示できます。



7. 各属性の値を設定します。

   a. 属性の鉛筆ボタンをクリックし、値を追加します。



   userPassword 値を設定すると、別のメニューが開くことに注意してください。プレーンテキストを非表示にするために、値にはアスタリスク (*) が入力されます。

   b. チェックボタンをクリックして変更を保存します。

   c. オプション: **Options menu (⋮) → Add Another Value** をクリックして、追加の属性値を設定します。

   d. すべての値を設定したら、**Next** をクリックします。

8. エントリーの詳細がすべて正しいことを確認し、**Create User** をクリックします。ディレクトリーサーバーは、POSIX ユーザーの必須属性を持つエントリーを作成し、それにパスワードを設定します。**Back** をクリックしてエントリーの設定を変更するか、**Cancel** をクリックしてエ

ントリーの作成をキャンセルできます。

9. Result for Entry Creation表示し、**Finish** をクリックします。

検証

1. LDAP Browser→Search に移動します。
2. **dc=example,cd=com** など、エントリーを含むデータベース接尾辞を選択します。
3. 検索条件 (例: **John**) をフィールドに入力し、**Enter** を押します。
4. エントリーのリストで最近作成したエントリーを見つけます。

2.2. WEB コンソールを使用した LDAP エントリーの編集

Web コンソールを使用してディレクトリーエントリーを変更できます。この例では、ユーザーエントリー **cn=John Smith,ou=people,dc=example,dc=com** を次のように変更します。

- 電話番号 **556778987** および **556897445** を追加します。
- 電子メール **jsmith@example.com** を追加します。
- パスワードを変更します。

前提条件

- ディレクトリーサーバー Web コンソールにログインしている。

手順

1. LDAP Browser メニューを開きます。
2. **ツリー** ビューまたは **テーブル** ビューを使用して、編集するエントリー (**cn=John Smith,ou=people,dc=example,dc=com** など) を展開します。
3. Options menu (▮) をクリックし、**Edit** を選択してウィザードウィンドウを開きます。
4. オプション: Select ObjectClasses の手順で、エントリーのオブジェクトクラスを追加または削除します。**Next** をクリックします。
5. Select Attributes ステップで、**telephoneNumber** および **mail** 属性をエントリーに追加し、**Next** をクリックします。エントリーに追加する属性が表示されない場合は、前の手順で対応するオブジェクトクラスを追加していないことを意味します。

注記

この手順では、選択したオブジェクトクラスの必須属性は **削除できません**。

6. Edit Attribute Values ステップで、**telephoneNumber** を **556778987** と **556897445** に設定し、**mail** を **jsmith@example.com** に設定して、**userPassword** 値を変更します。
   a. 属性の鉛筆ボタンをクリックして、新しい値を追加または変更します。
   b. チェックボタンをクリックして変更を保存します。

c. オプション: **Options menu (▮) → Add Another Value** をクリックして、属性に追加の値を設定します。この例では、**telephoneNumber** 属性に 2 つの値があります。すべての値を設定したら、**Next** をクリックします。

7. 変更内容を確認し、**Next** をクリックします。

8. エントリーを編集するには、**Modify Entry** をクリックします。**Back** をクリックしてエントリーに別の変更を加えるか、**Cancel** をクリックしてエントリーの変更をキャンセルできます。

9. **Result for Entry Modification**表示し、**Finish** をクリックします。

検証

- エントリーの詳細を展開し、エントリー属性に表示される新しく変更された内容を確認します。

関連情報

- Directory Server でのロールの使用
- Directory Server でのグループの使用

2.3. WEB コンソールを使用した LDAP エントリーまたはサブツリーの名前変更と再配置

Web コンソールを使用して、ディレクトリーエントリーまたはサブツリーの名前を変更または再配置できます。この例では、エントリー **cn=John Smith,ou=people,dc=example,dc=com** の名前を変更し **cn=Tom Smith,ou=clients,dc=example,dc=com** に再配置します。

前提条件

- ディレクトリーサーバー Web コンソールにログインしている。

手順

1. **LDAP Browser** メニューを開きます。

2. **ツリー** ビューまたは **テーブル** ビューを使用して、変更するエントリー (**cn=John Smith,ou=people,dc=example,dc=com** など) を展開します。

3. Options menu (▮) をクリックし、**Rename** を選択してウィザードウィンドウを開きます。

4. **Select The Naming Attribute And Value** の手順で以下を実行します。

   a. 命名属性 **cn** に新しい値 **Tom Smith** を設定し、**Next** をクリックします。

   b. オプション: ドロップダウンメニューから別の命名属性を選択します。

   c. オプション: 古いエントリーを削除し、新しい RDN を使用して新規エントリーを作成する場合は、**Delete the old RDN**をオンにします。

5. **Select The Entry Location**の手順で新しい場所の親エントリーを選択し、**Next** をクリックします。

6. エントリーに加えた変更を確認し、**Next** をクリックします。

7. エントリーの詳細が正しい場合は、**Change Entry Name** をクリックします。**Back** をクリックしてエントリーに別の変更を加えるか、**Cancel** をクリックしてエントリーの変更をキャンセルできます。

8. Result for Entry Modification を表示し、**Finish** をクリックします。

検証

- エントリーの詳細を展開し、更新されたエントリーを確認します。

2.4. WEB コンソールを使用した LDAP エントリーの削除

Web コンソールを使用して、ディレクトリーエントリーまたはサブツリーを削除できます。この例では、エントリー **cn=Tom Smith,ou=clients,dc=example,dc=com** を削除します。

前提条件

- ディレクトリーサーバー Web コンソールにログインしている。

手順

1. LDAP Browser メニューを開きます。

2. **ツリー** ビューまたは **テーブル** ビューを使用して、削除するエントリー (**cn=Tom Smith,ou=people,dc=example,dc=com** など) を展開します。

3. Options menu (▮) をクリックし、**Delete** を選択してウィザードウィンドウを開きます。

4. 削除するエントリーに関するデータを確認し、**Next** をクリックします。

5. Deletion の手順でスイッチを **Yes, I'm sure** の位置に切り替え、**Delete** をクリックします。**Cancel** をクリックすると、エントリーの削除をキャンセルできます。

6. Result for Entry Deletion を表示し、**Finish** をクリックします。

検証

1. LDAP Browser → Search に移動します。

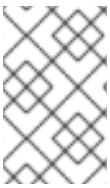
2. **dc=example,cd=com** など、エントリーが以前存在していた接尾辞を選択します。

3. 検索条件 (例: **Tom**) をフィールドに入力し、**Enter** を押します。

4. 削除されたエントリーが存在しないことを確認します。

第3章 一意の数値属性値の割り当ておよび管理

一部のエントリー属性には、**uidNumber** や **gidNumber** などの一意の数値識別子が必要です。Directory Server は、Distributed Numeric Assignment (DNA) プラグインを使用して、指定した属性に対してこれらの一意の番号を自動的に生成し、割り当てることができます。



注記

DNA プラグインは **属性の一意性** を保証するわけではありません。プラグインは重複しない範囲を割り当てます。これにより、一意性を強制したり検証したりすることなく、管理属性に手動で数値を割り当てることができます。

DNA プラグインを使用すると、レプリケーションの競合を事実上回避できます。DNA プラグインは、**単一** のバックエンド全体に一意の番号を割り当てます。マルチサプライヤーのレプリケーションでは、各サプライヤーがローカル DNA プラグインインスタンスを実行している場合、各サーバーに異なる範囲の番号を割り当てる必要があります。これにより、各インスタンスが真に一意の番号セットを使用するようになります。

3.1. 動的番号の割り当ての概要

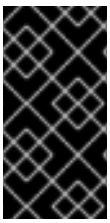
DNA プラグインは、インスタンスが発行できる使用可能な番号の範囲を割り当てます。範囲の定義は 2 つの属性で定義されます。サーバーで次に使用可能な番号 (範囲の下限値) と最大値 (範囲の上限値) です。プラグインを設定するときに、初期の下限値を設定します。追って、プラグインはこの下限値を更新します。

利用可能な数を各レプリカの複数の範囲に分割することで、サーバーは互いに重複することなく、継続的に番号を割り当てることができます。

3.1.1. フィルター、検索、およびターゲットエントリー

サーバーは、内部的にソートされた検索を実行し、次に指定された範囲がすでに別のサーバーによって取得されているかどうかを確認します。管理属性には、適切な順序のマッチングルールで等価インデックスを割り当てる必要があります。

DNA プラグインは、常にディレクトリーツリーの特定領域 (**スコープ**) と、そのサブツリー内の特定のエントリータイプ (**フィルター**) に適用されます。



重要

DNA プラグインは **単一** のバックエンドでのみ機能します。複数のデータベースの番号割り当ては管理できません。DNA プラグインはソートコントロールを使用して、DNA プラグイン以外で手動で値が割り当てられているかどうかを確認します。ただし、ソートコントロールを使用したこの検証は、**単一** のバックエンドでのみ機能します。

3.1.2. 範囲および割り当て番号

Directory Server は、いくつかの異なる方法を使用して属性値を生成できます。

- 基本的なシナリオでは、一意の番号属性を必要とするも属性値がないオブジェクトクラスを持つユーザーエントリーをディレクトリーに追加すると、DNA プラグインがアクティブになり、値を割り当てます。この割り当ては、DNA プラグインが単一の属性に一意の値を割り当てるように設定されている場合に行われます。

- より単純な方法では、管理属性のテンプレート値として **マジック番号** を使用します。このマ

ジック番号は数値または単語で、サーバーの範囲に含まれません。プラグインはこれを信号として認識し、新しく割り当てられた値に置き換えます。マジック値を使用してエントリーが追加され、設定した DNA プラグインのスコープとフィルターに当該エントリーが当てはまる場合、プラグインは新しい値を生成するよう要求されます。たとえば、**ldapmodify** を使用すると、マジック番号として 0 を追加できます。

```
dn: uid=jsmith,ou=people,dc=example,dc=com
 changetype: add
 objectClass: top
 objectClass: person
 objectClass: posixAccount
 uid: jsmith
 cn: John Smith
 uidNumber: 0
 gidNumber: 0
```

DNA プラグインは、新規の一意の値のみを生成します。DNA プラグインが制御する属性に特定の値を使用するようにエントリーを追加または変更した場合、プラグインはその値を上書きしません。

3.1.3. 同じ範囲の複数の属性

DNA プラグインは、一意の番号の単一範囲から、1つまたは複数の属性タイプに一意の番号を割り当てることができます。

これにより、属性に一意の番号を割り当てる方法が複数提供されます。

- 単一の属性タイプに、一意の範囲から単一の番号を割り当てる。
- 1つのエントリー内の 2 つの属性に、同じ一意の番号を割り当てる。
- 2 つの異なる属性に、同じ一意の範囲から異なる番号を割り当てる。

多くの場合、属性タイプごとに一意の番号を割り当てるだけで十分です。たとえば、新しい従業員エントリーに **employeeID** を割り当てる場合、各従業員エントリーに一意の **employeeID** を確実に割り当てることが重要です。

ただし、同じ範囲の番号から一意の番号を複数の属性に割り当てることが役に立つ場合もあります。たとえば、**uidNumber** と **gidNumber** を **posixAccount** エントリーに割り当てる場合、DNA プラグインは両方の属性に同じ番号を割り当てます。これを行うには、マジック値を指定して、両方の管理属性を変更操作に渡します。**ldapmodify** の使用:

```
#  ldapmodify -D "cn=Directory Manager" -W -x

dn: uid=jsmith,ou=people,dc=example,dc=com
changetype: modify
add: uidNumber
uidNumber: 0
-
add:gidNumber
gidNumber: 0
```

オブジェクトクラスで1つの属性しか許可されない場合、DNA プラグインは、複数の属性を処理する際に、一意の値を1つの属性にのみ割り当てることができます。たとえば、**posixGroup** オブジェクトクラスでは **gidNumber** は許可されますが、**uidNumber** は許可されません。DNA プラグインが **uidNumber** と **gidNumber** の両方を管理する場合、**posixGroup** エントリーの作成時に **uidNumber** と

gidNumber の属性範囲から **gidNumber** に一意の番号を割り当てます。すべての管理属性のプールを共有すると、一意の番号の一貫した割り当てが確保され、異なるエントリーの **uidNumber** と **gidNumber** が別々の範囲の同じ番号になるという競合を防止します。

DNA プラグインが複数の属性を管理する場合、1 回の変更操作ですべての属性に同じ値が割り当てられます。同じ範囲から **異なる** 番号を割り当てるには、個別の変更操作を実行する必要があります。たとえば、**ldapmodify** を使用できます。

```
# ldapmodify -D "cn=Directory Manager" -W -x
dn: uid=jsmith,ou=people,dc=example,dc=com
changetype: modify
add: uidNumber
uidNumber: 0
^D

# ldapmodify -D "cn=Directory Manager" -W -x
dn: uid=jsmith,ou=people,dc=example,dc=com
changetype: modify
add: employeeId
employeeId: magic
```

重要

DNA プラグインを使用して複数の属性に一意の番号を割り当てるには、一意の値を必要とする各属性に、一意の値を指定します。このような指定を必要としない単一の属性の場合とは異なり、複数の属性では一意の値を指定する必要があります。場合によっては、エントリーが範囲内のすべての属性を許可しないことや、すべてのタイプを許可するものの、一意の値を必要とするのは一部のみであることもあります。

例3.1例: DNA および一意の銀行口座番号

一例における銀行管理者は、顧客の **primaryAccount** 属性と **customerID** 属性に共通の一意の番号を割り当てるように、DNA プラグインを設定します。

また、銀行は、セカンダリー口座に対して、customer ID および primary account と同じ番号範囲から、プライマリー口座とは異なる一意の番号を割り当てたいと考えています。一例における銀行管理者は、**primaryAccount** と **customerID** に一意の番号を割り当てた後、エントリー作成後に追加した **secondaryAccount** 属性を管理するように DNA プラグインを設定します。これによって確実に、**primaryAccount** と **customerID** に共通の一意の番号が割り当てられ、異なる一意の **secondaryAccount** 番号が同じ範囲から割り当てられます。

第4章 属性の一意性の有効化

属性の値がディレクトリー全体またはサブツリー全体で一意であることを確認するには、デフォルトでは無効になっている属性一意性プラグインを使用できます。

次のいずれかの方法で、属性の一意性を検証するようにプラグインを設定できます。

- **uniqueness-subtrees** パラメーターを使用して、プラグインが属性の一意性をチェックする必要があるサブツリーのリストを設定します。次に例を示します。

  ```
  uniqueness-attribute-name: mail
  uniqueness-subtrees: ou=accounting,dc=example,dc=com
  uniqueness-subtrees: ou=sales,dc=example,dc=com
  uniqueness-across-all-subtrees: on
  uniqueness-exclude-subtrees: ou=private,ou=people,dc=example,dc=com
  ```

  詳細は、サブツリー上の属性一意性プラグインの設定 を参照してください。

- **uniqueness-top-entry-oc** パラメーターを使用して親エントリーオブジェクトクラスを設定します。更新されたエントリーの親エントリーにこのオブジェクトクラスが含まれている場合、プラグインは親エントリーサブツリーの下の属性の一意性をチェックします。たとえば、プラグインを次のように設定できます。

  ```
  uniqueness-attribute-name: mail
  uniqueness-top-entry-oc: nsContainer
  uniqueness-subtree-entries-oc: inetOrgPerson
  uniqueness-exclude-subtrees: ou=private,ou=people,dc=example,dc=com
  ```

  詳細は、オブジェクトクラスに対する属性一意性プラグインの設定 を参照してください。

プラグインの複数の設定エントリーを作成して、異なる条件を適用できます。Directory Server は、プラグインのすべての設定エントリーを **cn=plugins,cn=config** の下に保存します。

4.1. コマンドラインを使用したサブツリー上の ATTRIBUTE UNIQUENESS プラグインの設定

dsconf ユーティリティーを使用して、プラグインで属性の一意性をチェックする必要があるサブツリーのリストを設定できます。サブツリーは、接尾辞など、ディレクトリー内の任意のエントリーになります。

ou=sales,dc=example,dc=com および **ou=accounting,dc=example,dc=com** サブツリーの下のエントリー内の **mail** 属性の一意性を検証するようにプラグインを設定するには、次の例の手順を使用します。

前提条件

- Directory Manager のパーミッションがある。

手順

1. 新しいプラグイン設定エントリーを作成します。

```
# dsconf -D "cn=Directory Manager" ldap://server.example.com plugin attr-uniq add
"Mail Uniqueness" --attr-name mail --subtree ou=sales,dc=example,dc=com
ou=accounting,dc=example,dc=com
```

このコマンドは、**cn=Mail Uniqueness,cn=plugins,cn=config** 設定エントリーを作成します。

注記

プラグインを設定すると、1 つの設定エントリー内の複数の属性の一意性を検証できます。

2. オプション: このプラグイン設定エントリーで設定されたすべてのサブツリーにわたって一意性を設定します。

```
# dsconf -D "cn=Directory Manager" ldap://server.example.com plugin attr-uniq set
"Mail Uniqueness" --across-all-subtrees on
```

このコマンドは **uniqueness-across-all-subtrees** プラグイン設定パラメーターを **on** に設定します。したがって、プラグインは **ou=sales,dc=example,dc=com** および **ou=accounting,dc=example,dc=com** サブツリーの両方で **mail** 属性を一意性をチェックします。デフォルトでは、プラグインはエントリーが作成または更新されるサブツリー全体でのみ一意性をチェックします。つまり、**ou=sales,dc=example,dc=com** の下にエントリーを作成または更新すると、プラグインはこのサブツリー全体でのみ **mail** 属性の一意性をチェックします。

3. オプション: プラグインが属性の一意性の検証から除外する必要があるサブツリーを設定します。
たとえば、プラグインが **ou=internal,ou=sales,dc=example,dc=com** サブツリーを省略するには、**ldapmodify** ユーティリティーを使用して **uniqueness-exclude-subtrees** パラメーターを設定できます。

```
# ldapmodify -D "cn=Directory Manager" -W -H ldap://server.example.com -x

dn: cn=Mail Uniqueness,cn=plugins,cn=config
changetype: modify
add: uniqueness-exclude-subtrees
uniqueness-exclude-subtrees: ou=internal,ou=sales,dc=example,dc=com
```

4. オプション: プラグインで特定のオブジェクトクラスを含むエントリーのみの一意性を検証する場合は、このオブジェクトクラスを **uniqueness-subtree-entries-oc** パラメーターの値として設定します。
たとえば、**inetOrgPerson** オブジェクトクラスが含まれるエントリーでのみ **mail** 属性を一意にするには、次のように入力します。

```
# dsconf -D "cn=Directory Manager" ldap://server.example.com plugin attr-uniq set
"Mail Uniqueness" --subtree-entries-oc=inetOrgPerson
```

5. サーバーでプラグインを有効にします。

```
# dsconf -D "cn=Directory Manager" ldap://server.example.com plugin attr-uniq enable
"Mail Uniqueness"
```

6. インスタンスを再起動します。

```
# dsctl instance_name restart
```

検証

- 設定エントリーの詳細を表示します。

```
# dsconf -D "cn=Directory Manager" ldap://server.example.com plugin attr-uniq show
"Mail Uniqueness"

dn: cn=Mail Uniqueness,cn=plugins,cn=config
cn: Mail Uniqueness
nsslapd-plugin-depends-on-type: database
nsslapd-pluginDescription: Enforce unique attribute values
nsslapd-pluginEnabled: on
...
uniqueness-across-all-subtrees: on
uniqueness-attribute-name: mail
uniqueness-exclude-subtrees: ou=internal,ou=sales,dc=example,dc=com
uniqueness-subtree-entries-oc: inetOrgPerson
uniqueness-subtrees: ou=accounting,dc=example,dc=com
uniqueness-subtrees: ou=sales,dc=example,dc=com
```

関連情報

- Attribute Uniqueness プラグインの属性

4.2. オブジェクトクラスを介した ATTRIBUTE UNIQUENESS プラグインの設定

Attribute Uniqueness プラグインを設定すると、特定のオブジェクトクラスを含むエントリー内で属性の値を一意に保つことができます。プラグインを設定するには、次の設定パラメーターを設定する必要があります。

- **uniqueness-top-entry-oc**.このパラメーターは、プラグインが属性の一意性を検証するサブツリーを一意に識別します。プラグインは、**uniqueness-top-entry-oc** で設定した特定のオブジェクトクラスが親エントリーに含まれるエントリーのみの一意性を検証します。Directory Server が更新されたエントリーの親エントリーでオブジェクトクラスを見つけられなかった場合、検索はディレクトリーツリーのルートまで、次の上位レベルのエントリーで続行されます。

- **uniqueness-subtree-entries-oc**.このパラメーターは、プラグインがチェックする必要があるエントリーを識別します。**uniqueness-subtree-entries-oc** パラメーターでオブジェクトクラスを設定すると、プラグインは、この特定のオブジェクトクラスを含む更新されたエントリー内の属性の一意性のみを検証します。

nsContainer オブジェクトクラスセットが含まれるエントリーの下にあるすべてのエントリーで **mail** 属性が一意になるように設定し、プラグインが **inetOrgPerson** オブジェクトクラスを含むエントリーで **mail** 属性を検索するには、次の例の手順を使用します。

前提条件

- Directory Manager のパーミッションがある。

手順

1. 新しいプラグイン設定エントリーを作成します。

```
# dsconf -D "cn=Directory Manager" ldap://server.example.com plugin attr-uniq add
"Mail Uniqueness with OC" --attr-name mail --subtree-entries-oc=inetOrgPerson --top-
entry-oc=nsContainer
```

このコマンドは、設定された **uniqueness-top-entry-oc** および **uniqueness-subtree-entries-oc** プラグインパラメーターを使用して、 **cn=Mail Uniqueness with OC,cn=plugins,cn=config** エントリーを作成します。

2. オプション: プラグインが属性の一意性の検証から除外する必要があるサブツリーを設定します。
たとえば、プラグインが **ou=internal,ou=sales,dc=example,dc=com** サブツリーを省略するには、**ldapmodify** ユーティリティーを使用して **uniqueness-exclude-subtrees** パラメーターを設定します。

```
# ldapmodify -D "cn=Directory Manager" -W -H ldap://server.example.com -x

dn: cn=Mail Uniqueness with OC,cn=plugins,cn=config
changetype: modify
add: uniqueness-exclude-subtrees
uniqueness-exclude-subtrees: ou=internal,ou=sales,dc=example,dc=com
```

3. サーバーでプラグインを有効にします。

```
# dsconf -D "cn=Directory Manager" ldap://server.example.com plugin attr-uniq enable
"Mail Uniqueness with OC"
```

4. インスタンスを再起動します。

```
# dsctl instance_name restart
```

検証

- 設定エントリーの詳細を表示します。

```
# dsconf -D "cn=Directory Manager" ldap://server.example.com plugin attr-uniq show
"Mail Uniqueness with OC"

dn: cn=Mail Uniqueness with OC,cn=plugins,cn=config
cn: Mail Uniqueness with OC
nsslapd-plugin-depends-on-type: database
nsslapd-pluginDescription: none
nsslapd-pluginEnabled: on
...
uniqueness-attribute-name: mail
uniqueness-exclude-subtrees: ou=internal,ou=sales,dc=example,dc=com
uniqueness-subtree-entries-oc: inetOrgPerson
uniqueness-top-entry-oc: nsContainer
```

4.3. WEB コンソールを使用した ATTRIBUTE UNIQUENESS プラグインの設定

Web コンソールを使用して、Attribute Uniqueness プラグインを設定できます。プラグインの異なる設定エントリーを作成して、異なる条件を適用できることに注意してください。

ou=sales,dc=example,dc=com および **ou=accounting,dc=example,dc=com** サブツリーの下のエントリー内の **mail** 属性の一意性を検証するようにプラグインを設定するには、次の例の手順を使用します。

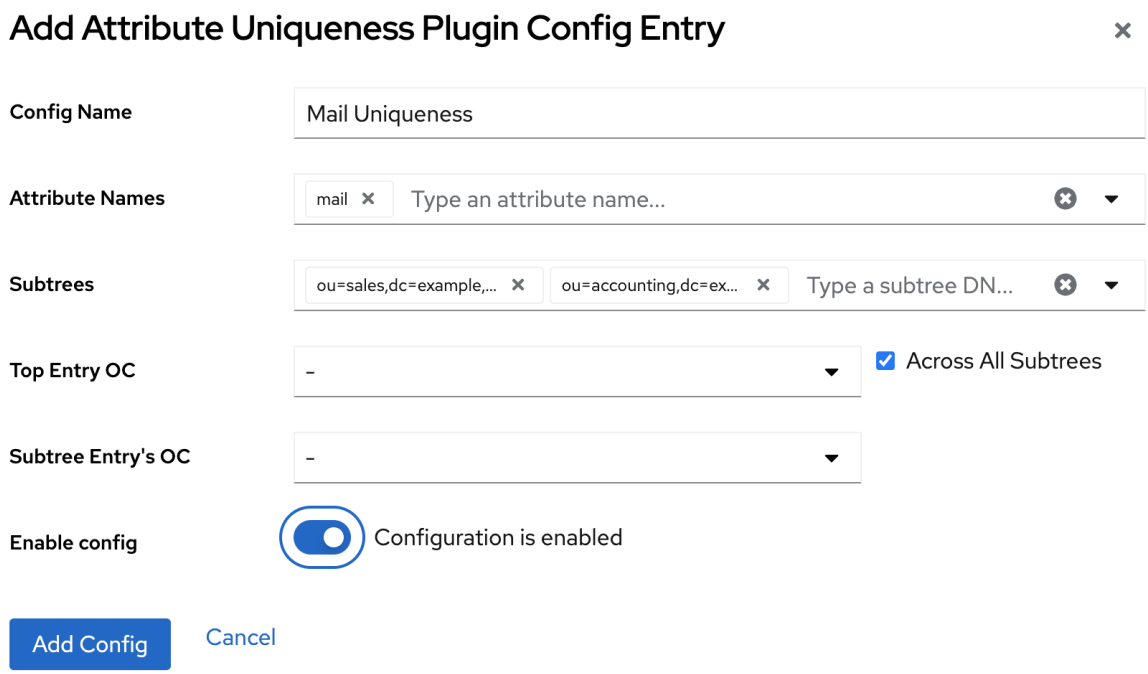
前提条件

- Directory Manager のパーミッションがある。
- Web コンソールにログインしている。詳細は、Web コンソールを使用した Directory Server へのログイン を参照してください。

手順

1. プラグインを設定するインスタンスを選択します。
2. **Plugins** メニューを開き、一覧から **Attribute Uniqueness** プラグインを選択します。
3. 新しい設定エントリーの設定を開始するには、**Add Config** ボタンをクリックします。
4. **Config Name** フィールドに設定エントリーの名前を入力します。
5. **Attribute Names** フィールドで一意である必要がある属性を選択します。このフィールドは **uniqueness-attribute-name** 属性を設定します。
6. プラグインが **Subtrees** フィールドで属性の一意性をチェックするサブツリーを入力します。このフィールドは **uniqueness-subtrees** 属性を設定します。
   デフォルトでは、プラグインはエントリーが作成または更新されるサブツリー全体でのみ一意性をチェックします。リストされているすべてのサブツリーをチェックするには、**uniqueness-across-all-subtrees** 属性を **on** に設定する **Across All Subtrees** チェックボックスをオンにします。
7. **Configuration is enabled** の位置にスイッチを切り替えます。
8. **Add Config** ボタンをクリックしてプラグイン設定エントリーを作成します。

図4.1 Attribute Uniqueness プラグインの設定例。



9. インスタンスを再起動します。詳細は、Web コンソールを使用した Directory Server インスタンスの起動と停止 を参照してください。

検証

- 設定エントリーのリストで新しく作成されたプラグインエントリーを見つけます。

関連情報

- Attribute Uniqueness プラグインの属性