



Red Hat OpenStack Platform

8

Identity サービスとの統合

Active Directory、IdM、汎用 LDAP を外部認証バックエンドとして使用する
方法

OpenStack Team

Active Directory、IdM、汎用 LDAP を外部認証バックエンドとして使用する 方法

OpenStack Team
rhos-docs@redhat.com

法律上の通知

Copyright © 2017 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux ® is the registered trademark of Linus Torvalds in the United States and other countries.

Java ® is a registered trademark of Oracle and/or its affiliates.

XFS ® is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL ® is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js ® is an official trademark of Joyent. Red Hat Software Collections is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack ® Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

概要

Active Directory、IdM、汎用 LDAP を外部認証バックエンドとして使用する方法

目次

前書き	4
第1章 ACTIVE DIRECTORY との統合	5
1.1. 主要な用語	5
1.2. 前提条件	5
1.3. ファイアウォールの設定	5
1.4. 影響評価	5
1.5. 停止の要件	6
1.6. ACTIVE DIRECTORY ドメインサービスの設定	6
1.7. LDAPS 証明書の設定	6
1.8. IDENTITY サービスの設定	7
1.9. 新規プロジェクトの作成	15
1.10. DASHBOARD へのログインプロセスの変更	16
1.11. コマンドラインへの変更	16
1.12. AD DS 統合のテスト	16
1.13. 高可用性の設定	17
1.14. トラブルシューティング	17
第2章 IDENTITY MANAGEMENT の統合	19
2.1. 主要な用語	19
2.2. 前提条件	19
2.3. ファイアウォールの設定	19
2.4. 影響評価	19
2.5. 停止の要件	20
2.6. IDM サーバーの設定	20
2.7. LDAPS 証明書の設定	20
2.8. IDENTITY サービスの設定	21
2.9. コントローラーの設定	22
2.10. COMPUTE が KEYSTONE V3 を使用するようにするための設定	26
2.11. BLOCK STORAGE が KEYSTONE V3 を使用するようにするための設定	27
2.12. IDM ユーザーによるプロジェクトへのアクセスの許可	27
2.13. DASHBOARD へのログインプロセスの変更	28
2.14. コマンドラインへの変更	29
2.15. IDM 統合のテスト	29
2.16. 高可用性の設定	29
2.17. トラブルシューティング	30
第3章 汎用 LDAP の統合	32
3.1. 主要な用語	32
3.2. 前提条件	32
3.3. ファイアウォールの設定	32
3.4. 影響評価	32
3.5. 停止の要件	33
3.6. LDAP サーバーの設定	33
3.7. LDAPS 証明書の設定	33
3.8. IDENTITY サービスの設定	34
3.9. コントローラーの設定	34
3.10. COMPUTE が KEYSTONE V3 を使用するようにするための設定	39
3.11. BLOCK STORAGE が KEYSTONE V3 を使用するようにするための設定	39
3.12. LDAP ユーザーによるプロジェクトへのアクセスの許可	39
3.13. DASHBOARD へのログインプロセスの変更	41
3.14. コマンドラインへの変更	41

3.15. LDAP 統合のテスト	42
3.16. 高可用性の設定	42
3.17. トラブルシューティング	43

前書き

Identity サービス (コード名 **keystone**) は Red Hat OpenStack Platform 8 の認証と承認の機能を提供します。

本ガイドは、Microsoft Active Directory Domain Service (AD DS) および Red Hat Identity Management (IdM) に Identity サービスを統合する方法について説明します。

第1章 ACTIVE DIRECTORY との統合

本章では、Identity サービス (keystone) を Active Directory ドメインサービスに統合する方法について説明します。以下のユースケースでは、Identity サービスが特定の Active Directory ドメインサービス (AD DS) のユーザーを認証しつつ、Identity サービスデータベース内で承認設定および重要なサービスアカウントを保持します。この手順を実行すると、Identity サービスは、AD DS に読み取り専用でアクセスしてユーザーアカウントの認証を行う一方で、認証されたアカウントに割り当てる権限を引き続き管理するようになります。

1.1. 主要な用語

- ※ **認証**: パスワードを使用して、ユーザーが本人であることを検証するプロセス
- ※ **承認**: 認証されたユーザーに対して、アクセスしようとしているシステムの適切なパーミッションが付与されていることを確認するプロセス
- ※ **ドメイン**: この用語は、AD DS のドメインとは異なり、ユーザー、グループ、プロジェクトの領域確保のために Identity サービスで設定される追加の名前空間のことを指します。この機能により、異なる LDAP または AD DS 環境のユーザーを認証する別個のドメインを設定することができます。

1.2. 前提条件

本ガイドのデプロイメント例は、以下を前提としています。

- ※ Active Directory ドメインサービスが設定済みで、稼働していること。
- ※ Red Hat OpenStack Platform が設定済みで、稼働していること。
- ※ DNS 名前解決が完全に機能しており、かつ全ホストが適切に登録されていること。
- ※ AD DS 認証トラフィックが LDAPS で暗号化され、ポート 636 を使用していること。

1.3. ファイアウォールの設定

ファイアウォールで AD DS と OpenStack の間のトラフィックをフィルタリングしている場合には、以下のポートを介したアクセスを許可する必要があります。

送信元	送信先	種別	ポート
OpenStack コントローラーノード	Active Directory ドメインコントローラー	LDAPS	TCP 636

1.4. 影響評価

以下のステップを実行すると、AD DS ユーザーが OpenStack に対して認証を実行して、リソースにアクセスできるようになります。OpenStack のサービスアカウント (keystone、glance など) および承認管理 (パーミッション、ロール、プロジェクト) は Identity サービスのデータベースに残ります。パーミッションとロールは、Identity サービスの管理ツールを使用して AD DS アカウントに割

り当てられます。

1.4.1. 高可用性のオプション

この設定により、単一の Active Directory Domain Controller に依存するようになるため、Identity サービスがその AD Domain Controller に対して認証できない場合には、プロジェクトユーザーが影響を受けることになります。このリスクを管理するオプションは複数あります。たとえば、Identity サービスが個別の AD Domain Controller ではなく DNS エイリアスやロードバランシングアプライアンスにクエリーを実行するように設定することが可能です。また、Domain Controller の 1 つが利用できない場合には、keystone が異なる Domain Controller にクエリーを実行するように設定することもできます。詳しくは、「[高可用性の設定](#)」を参照してください。

1.5. 停止の要件

- ※ AD DS バックエンドを追加するには、Identity サービスを再起動する必要があります。
- ※ **keystone** v3 に切り替えるには、全ノード上の Compute サービスを再起動する必要があります。
- ※ ユーザーは、AD DS でアカウントが作成されるまでは、Dashboard にアクセスできません。ダウンタイムを短縮するには、この変更の前に十分余裕をもって AD DS アカウントのプレステージを行うことを検討してください。

1.6. ACTIVE DIRECTORY ドメインサービスの設定

Active Directory ドメインコントローラーで、次の PowerShell コマンドを実行します。

1. LDAP ルックアップアカウントを作成します。このアカウントは、Identity サービスが AD DS LDAP サービスにクエリーを実行するのに使用されます。

```
PS C:\> New-ADUser -SamAccountName svc-ldap -Name "svc-ldap" -GivenName
LDAP -Surname Lookups -UserPrincipalName svc-ldap@lab.local -Enabled
$false -PasswordNeverExpires $true -Path 'CN=Users,DC=lab,DC=local'
```

2. このアカウントのパスワードを設定し、有効にします。AD ドメインのパスワードの複雑さの要件を満たすパスワードを指定するように要求されます。

```
PS C:\> Set-ADAccountPassword svc-ldap -PassThru | Enable-ADAccount
```

3. **grp-openstack** という名前の OpenStack ユーザーグループを作成します。このグループのメンバーに対してのみ、Dashboard 内の **プロジェクト** へのアクセス権を付与することが可能です。

```
PS C:\> NEW-ADGroup -name "grp-openstack" -groupscope Global -path
"CN=Users,DC=lab,DC=local"
```

4. **grp-openstack** グループに **svc-ldap** ユーザーを追加します。

```
PS C:\> ADD-ADGroupMember "grp-openstack" -members "svc-ldap"
```

1.7. LDAPS 証明書の設定

1. Windows 環境で、LDAPS 証明書の (秘密鍵ではなく) 公開鍵 をDER でエンコードされた **x509** .cer ファイルとしてエクスポートします。
2. OpenStack Identity (keystone) を実行しているノードにエクスポートされたファイルをコピーし、.cer を .pem に変換します。以下の例では、**addc.lab.local.cer** という名前の証明書ファイルを使用しています。

```
# openssl x509 -inform der -in addc.lab.local.cer -out  
addc.lab.local.pem
```

3. Red Hat Enterprise Linux に .pem をインストールします。

```
# cp addc.lab.local.pem /etc/pki/ca-trust/source/anchors/  
# update-ca-trust
```

4. .pem を .crt に変換して、証明書のディレクトリーにコピーします。

```
# openssl x509 -outform der -in addc.lab.local.pem -out  
addc.lab.local.crt  
# cp addc.lab.local.crt /etc/ssl/certs/
```

1.8. IDENTITY サービスの設定

以下のステップでは、Identity サービスを AD DS との統合するための準備をします。

1.8.1. keystone v3 へのコマンドラインアクセスの有効化

コマンドラインから Identity サービスドメインを管理するには、**keystone v3** にアクセス可能である必要があります。Identity サービスを実行しているコントローラーから以下の手順を実行します。

1. 既存の **keystonerc_admin** ファイルのコピーを作成します。

```
# cp keystonerc_admin keystonerc_admin_v3
```

2. 新しい **keystonerc_admin_v3** ファイルを編集して、**OS_AUTH_URL** を **v2.0** から **v3** に変更します。

```
export OS_AUTH_URL=http://controllerIP:5000/v3/
```

keystonerc_admin_v3 の一番下に、以下のエントリーを追加します。

```
export OS_IDENTITY_API_VERSION=3  
export OS_PROJECT_DOMAIN_NAME=Default  
export OS_USER_DOMAIN_NAME=Default
```

3. source コマンドでこの設定ファイルを読み込み、現在のコマンドラインセッションでこれらのオプションを有効にします。

```
# source keystonerc_admin_v3
```

1.8.2. コントローラーの設定

keystone サービスを実行するコントローラーから以下の手順を実行します。

1. SELinux を設定します。

```
# setsebool -P authlogin_nsswitch_use_ldap=on
```

出力には、以下のようなメッセージが含まれている場合がありますが、これは無視できます。

```
Full path required for exclude: net:[4026532245].
```

2. **domains** ディレクトリーを作成します。

```
# mkdir /etc/keystone/domains/
# chown keystone /etc/keystone/domains/
```

3. Identity サービスが複数のバックエンドを使用するように設定します。

```
# openstack-config --set /etc/keystone/keystone.conf identity
domain_specific_drivers_enabled true
# openstack-config --set /etc/keystone/keystone.conf identity
domain_config_dir /etc/keystone/domains
# openstack-config --set /etc/keystone/keystone.conf assignment driver
keystone.assignment.backends.sql.Assignment
```

注記

Red Hat OpenStack Platform director を使用している場合には、**/etc/keystone/keystone.conf** が Puppet で管理されている点を認識する必要があります。このため、カスタムの設定を追加しても、**openstack overcloud deploy** プロセスを実行する度に上書きされる可能性があります。上書きされた場合には、この設定を毎回手動で追加し直す必要がある場合があります。今後の director のリリースでは、デプロイ後のスクリプトを使用して、このような設定を自動的に再適用するための Puppet のパラメーターが追加される予定です。

4. Dashboard で複数のドメインを有効にします。以下の行を **/etc/openstack-dashboard/local_settings** に追加します。

```
OPENSTACK_API_VERSIONS = {
    "identity": 3
}
OPENSTACK_KEYSTONE_MULTIDOMAIN_SUPPORT = True
OPENSTACK_KEYSTONE_DEFAULT_DOMAIN = 'Default'
```

注記

Red Hat OpenStack Platform director を使用している場合には、**/etc/openstack-dashboard/local_settings** が Puppet で管理されている点を認識する必要があります。このため、カスタムを設定を追加しても、**openstack overcloud deploy** プロセスを実行する度に上書きされる可能性があります。上書きされた場合には、この設定を毎回手動で追加し直す必要がある場合があります。今後の director のリリースでは、デプロイ後のスクリプトを使用して、このような設定を自動的に再適用するための Puppet のパラメーターが追加される予定です。

keystone および **dashboard** サービスを再起動して、設定を適用します。

```
# systemctl restart openstack-keystone.service
# systemctl restart httpd
```

5. 追加のバックエンドを設定します。

a. AD DS ドメインから、NetBIOS の名前を取得します。

この値は、次のステップで必要な設定ファイルに適切な名前を付けるのに使用します。Active Directory Domain Controller で以下のコマンドを実行して値を取得します。

```
PS C:\> Get-ADDomain | select NetBIOSName
NetBIOSName
-----
LAB
```

以下の例では、**LAB** は、Identity サービスドメインとして使用する NetBIOS の名前に置き換えます。

b. AD DS を統合するための **keystone** ドメインを作成します。

上記のステップで取得した NetBIOS 名の値をドメイン名として使用します。たとえば、NetBIOS 名が **LAB** の場合には、以下のコマンドを実行します。

```
# openstack domain create LAB
```

注記

このコマンドが使用できない場合には、前述したように、# **source keystone_admin_v3** のコマンドを実行して、コマンドラインセッションでの **keystone v3** へのアクセスが有効化されているかどうかを確認します。

c. 設定ファイルを作成します。

AD DS バックエンドを追加するには、**/etc/keystone/domains/keystone.LAB.conf** (**LAB** は、前のステップで取得した NetBIOS 名に置き換えます) という名前の新規ファイルに LDAP 設定を入力します。

以下の設定例は、実際に使用する AD DS のデプロイメントに適した設定に変更する必要があります。

```
[ldap]
```

```

url = ldaps://addc.lab.local:636
user = CN=svc-ldap,CN=Users,DC=lab,DC=local
password = RedactedComplexPassword
suffix = DC=lab,DC=local
user_tree_dn = CN=Users,DC=lab,DC=local
user_objectclass = person
user_filter = (memberOf=cn=grp-openstack,CN=Users,DC=lab,DC=local)
user_id_attribute = sAMAccountName
user_name_attribute = sAMAccountName
user_mail_attribute = mail
user_pass_attribute =
user_enabled_attribute = userAccountControl
user_enabled_mask = 2
user_enabled_default = 512
user_attribute_ignore = password,tenant_id,tenants
user_allow_create = False
user_allow_update = False
user_allow_delete = False
use_tls = False
tls_cacertfile = /etc/ssl/certs/addc.lab.local.crt

[identity]
driver = keystone.identity.backends.ldap.Identity

```

設定項目についての説明

設定	説明
url	認証に使用する AD Domain Controller。LDAPS ポート 636 を使用します。
user	LDAP クエリーに使用する AD アカウントの 識別名。たとえば、 Get-ADuser svc-ldap select DistinguishedName を使用する AD 内の svc-ldap アカウントの 識別名 の値を特定することができます。
password	上記で使した AD アカウントのパスワード (プレーンテキスト形式)
suffix	AD ドメインの 識別名。この値は、 Get-ADDomain select DistinguishedName を使用して特定することができます。
user_tree_dn	OpenStack アカウントを含む 組織単位 (OU)

設定	説明
user_objectclass	AD タイプ person を使用します。
user_filter	Identity サービスに対して提示するユーザーをフィルタリングすることにより、 grp-openstack グループのメンバーのみに Identity サービスで定義されているパーミッションを付与することができます。この値には、グループの完全な 識別名 が必要です (Get-ADGroup grp-openstack select DistinguishedName)。
user_id_attribute	ユーザー ID に使用する AD 値をマッピングします。
user_name_attribute	names に使用する AD 値をマッピングします。
user_mail_attribute	ユーザーのメールアドレスに使用する AD 値をマッピングします。
user_pass_attribute	この値は、意図的に空白のままにします。
user_enabled_attribute	アカウントが有効にされているかどうかを検証する AD の設定
user_enabled_mask	アカウントが有効化されているかを判断するために確認すべき値を定義します。ブール値が返されない場合に使用します。
user_enabled_default	アカウントが有効化されていることを示す AD 値
user_attribute_ignore	Identity サービスが無視する必要があるユーザー属性を定義します。
user_allow_create	LDAP アカウントに変更を加える機能を予期しないように、Identity サービスに通知します。

設定	説明
user_allow_update	LDAP アカウントに変更を加える機能を予期しないように、Identity サービスに通知します。
user_allow_delete	LDAP アカウントに変更を加える機能を予期しないように、Identity サービスに通知します。
use_tls	TLS を使用するかどうかを定義します。 STARTTLS ではなく LDAPS で暗号化する場合には、無効にする必要があります。
tls_cacertfile	.crt 証明書ファイルへのパスを指定します。

6. 設定ファイルの所有権を **keystone** ユーザーに変更します。

```
# chown keystone /etc/keystone/domains/keystone.LAB.conf
```

7. **admin** ユーザーにドメインへのアクセス権を付与します。



注記

この手順を実行しても、OpenStack admin アカウントには実際の AD DS ドメインに対するパーミッションは付与されない点を念頭に置いてください。この場合には、ドメインという用語は、OpenStack が使用する **keystone** ドメインのことを指しています。

a. **LAB** ドメインの ID を取得します。

```
# openstack domain show LAB
+-----+-----+
| Field  | Value |
+-----+-----+
| enabled | True  |
| id      | 6800b0496429431ab1c4efbb3fe810d4 |
| name    | LAB   |
+-----+-----+
```

b. **admin** ユーザーの ID の値を取得します。

```
# openstack user list --domain default | grep admin
| 3d75388d351846c6a880e53b2508172a | admin |
```

c. **admin** ロールの ID の値を取得します。

```
# openstack role list
```



```
+-----+-----+
| ID                               | Name           |
+-----+-----+
| 544d48aaffde48f1b3c31a52c35f01f9 | SwiftOperator |
| 6d005d783bf0436e882c55c62457d33d | ResellerAdmin |
| 785c70b150ee4c778fe4de088070b4cf | admin         |
| 9fe2ff9ee4384b1894a90878d3e92bab | _member_     |
+-----+-----+
```

d. 返されたドメインおよび admin の ID を使用して、admin ユーザーを keystone **LAB** ドメインの admin ロールに追加するためのコマンドを構築します。

```
# openstack role add --domain 6800b0496429431ab1c4efbb3fe810d4 --user
3d75388d351846c6a880e53b2508172a 785c70b150ee4c778fe4de088070b4cf
```

8. OpenStack サービスを再起動して、変更を適用します。

```
# systemctl restart openstack-keystone.service
```

注記

httpd サービス内で keystone を実行している場合には、「# **systemctl restart httpd**」のコマンドで httpd を再起動して keystone の設定を適用する必要があります。

9. コマンドで NetBIOS 名を指定して、AD DS ドメイン内のユーザー一覧を確認します。

注記

リブートまたはサービスの再起動後には、LDAP に対してクエリーを実行できるようになるまでに多少時間がかかる場合があります。

```
# openstack user list --domain LAB
```

10. ローカルの Identity サービスデータベース内のサービスアカウントを確認します。

```
# openstack user list --domain default
```

1.8.3. Compute が keystone v3 を使用するようにするための設定

Compute はデフォルトで **keystone v2.0** を使用するように設定されているので、マルチドメイン機能を使用するためには、**keystone v3** を使用するように設定する必要があります。

1. 各コンピュートノードとコントローラーで **keystone_authtoken** 値を変更します。

```
# openstack-config --set /etc/nova/nova.conf keystone_authtoken
auth_version v3
```

2. コントローラーでサービスを再起動して、変更を適用します。

```
# systemctl restart openstack-nova-api.service openstack-nova-
cert.service openstack-nova-conductor.service openstack-nova-
consoleauth.service openstack-nova-novncproxy.service openstack-nova-
scheduler.service
```

3. 各コンピューターノードでサービスを再起動して、変更を適用します。

```
# systemctl restart openstack-nova-compute.service
```

1.8.4. Block Storage が keystone v3 を使用するようにするための設定

keystone v3 に対して認証を実行するには、Block Storage (cinder) を設定する必要もあります。

/etc/cinder/cinder.conf を編集します。

```
[keystone_authtoken]
auth_uri = http://controllerIP:5000/v3
auth_version = v3
```

✦ **auth_uri** の **controllerIP** の箇所をコントローラーの IP アドレスに置き換えます。

1.8.5. Active Directory ユーザーによるプロジェクトへのアクセスの許可

grp-openstack AD グループのメンバーである AD DS ユーザーには、Dashboard 内のプロジェクトにログインするパーミッションを付与することができます。

1. AD ユーザーの一覧を取得します。

```
# openstack user list --domain LAB
+-----+
+-----+
| ID                                     |
| Name                                |
+-----+
+-----+
| 1f24ec1f11aeb90520079c29f70afa060d22e2ce92b2eba7784c841ac418091e | user1 |
| 12c062faddc5f8b065434d9ff6fce03eb9259537c93b411224588686e9a38bf1 | user2 |
| afaf48031eb54c3e44e4cb0353f5b612084033ff70f63c22873d181fdae2e73c | user3 |
| e47fc21dcf0d9716d2663766023e2d8dc15a6d9b01453854a898cabb2396826e | user4 |
+-----+
+-----+
```

2. ロールの一覧を取得します。

```
# openstack role list
+-----+-----+
| ID                                     | Name                                |
+-----+-----+
| 544d48aaffde48f1b3c31a52c35f01f9 | SwiftOperator |
```

```
| 6d005d783bf0436e882c55c62457d33d | ResellerAdmin |
| 785c70b150ee4c778fe4de088070b4cf | admin          |
| 9fe2ff9ee4384b1894a90878d3e92bab | _member_      |
+-----+-----+
```

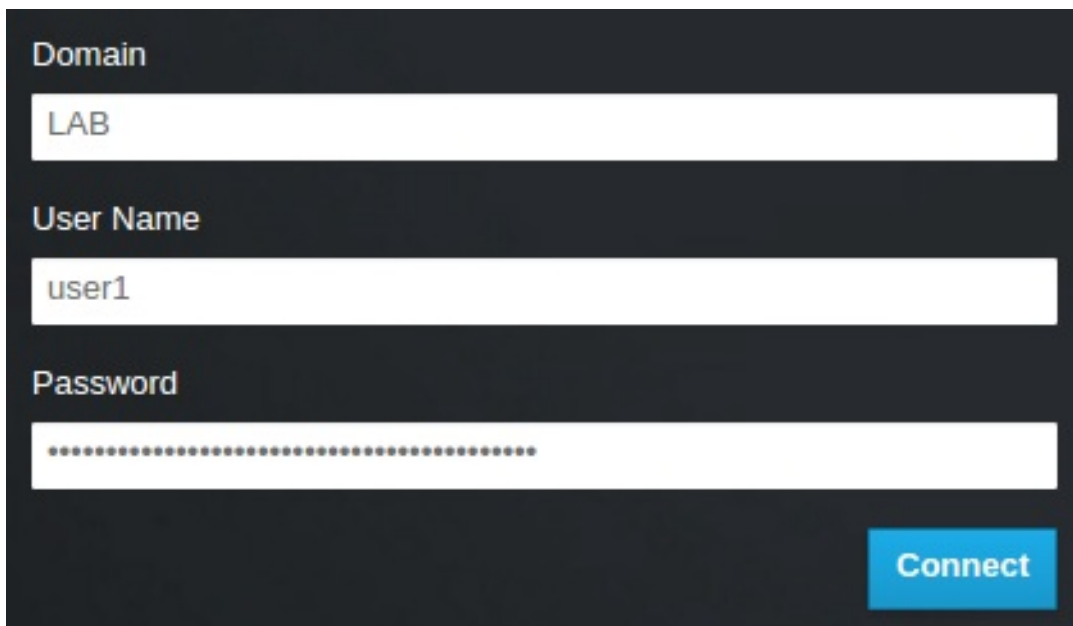
3. 一覧表示されたロールの中から 1 つまたは複数のロールをユーザーに追加して、プロジェクトへのアクセス権を付与します。たとえば、**user1** を demo プロジェクトの一般ユーザーにするには、**member** ロールに追加します。

```
# openstack role add --project demo --user
1f24ec1f11aeb90520079c29f70afa060d22e2ce92b2eba7784c841ac418091e
_member_
```

また、**user1** を demo プロジェクトの管理ユーザーにするには、**admin** ロールに追加します。

```
# openstack role add --project demo --user
1f24ec1f11aeb90520079c29f70afa060d22e2ce92b2eba7784c841ac418091e admin
```

その結果、**user1** は AD DS のユーザー名とパスワードを入力してから、ドメインのフィールドにも **LAB** と入力すると Dashboard にログインすることができます。




注記

ユーザーに「Error: Unable to retrieve container list.」というエラーメッセージが表示され、コンテナの管理が可能であることが想定されている場合には、**SwiftOperator** ロールに追加する必要があります。

1.9. 新規プロジェクトの作成

上記の統合ステップが完了した後に新規プロジェクトを作成する場合には、**Default** ドメインと、自分で作成した **keystone** ドメインのどちらに作成するかを決定する必要があります。これは、ワークフローとユーザーアカウントの管理方法を考慮して決定することができます。**Default** ドメインは、サービスアカウントと **admin** プロジェクトに使用する内部ドメインとして考えるこ

とができるので、AD でバックアップされているユーザーを異なる keystone ドメインに内に配置することは理にかなっています。これは、AD ユーザーが管理されているのと全く同じ keystone ドメインである必要はありません。また、分離の目的で複数の keystone ドメインを使用することも可能です。

1.10. DASHBOARD へのログインプロセスの変更

Identity サービスで複数のドメインを有効にすると、Dashboard のログインページに **ドメイン** という新しいフィールドが表示されます。

このフィールドは、ユーザーがログインするのに使用する認証情報と一致するドメインを入力します。このフィールドには、**keystone** にあるドメインの 1 つを手動で入力する必要があります。**openstack** コマンドで利用可能なエントリーを一覧表示します。

以下の例では、AD DS アカウントには **LAB** ドメインを指定する必要があります。**admin** のような組み込みの **keystone** アカウントには、ドメインに **Default** を指定する必要があります。

```
# openstack domain list
+-----+-----+-----+-----+
| ID                                           | Name      | Enabled | Description |
+-----+-----+-----+-----+
| 6800b0496429431ab1c4efbb3fe810d4 | LAB       | True    |             |
| default                                     | Default   | True    | Owns users  |
| and tenants (i.e. projects) available on Identity API v2. |           |         |             |
+-----+-----+-----+-----+
```

1.11. コマンドラインへの変更

特定のコマンドでは、対象のドメインを指定する必要がある場合があります。たとえば、以下のコマンドに **--domain LAB** を追加すると、LAB ドメイン内のユーザー (**grp-openstack** グループのメンバー) が返されます。

```
# openstack user list --domain LAB
```

--domain Default を追加すると、組み込みの keystone アカウントが返されます。

```
# openstack user list --domain Default
```

1.12. AD DS 統合のテスト

以下の手順では、Dashboard の機能へのユーザーアクセスをテストして、AD DS の統合を検証します。

1. AD にテストユーザーを作成し、そのユーザーを **grp-openstack** AD DSグループに追加します。
2. テストユーザーを **demo** テナントの **_member_** ロールに追加します。
3. AD テストユーザーの認証情報を使用して Dashboard にログインします。

4. 各タブをクリックし、エラーメッセージなしに正常に表示されているかどうかを確認します。
5. Dashboard を使用してテストインスタンスをビルドします。

注記

上記の手順で問題が発生した場合には、ステップ 3 から 5 までを組み込みの **admin** アカウントで実行してください。正常に実行できた場合には、OpenStack が想定通りに機能していることが実証されるので、問題は AD と Identity の統合設定の間のどこかにあることになります。「[トラブルシューティング](#)」を参照してください。

1.13. 高可用性の設定

keystone v3 が有効化されている場合には、ドメインの設定ファイルに複数の AD Domain Controller をリストして、この設定を高可用性にすることが可能です。

1. 2 番目のサーバーを **url** エントリーに追加します。たとえば、**keystone.LAB.conf** ファイル内の **url** 設定を更新すると、Identity サービスは全クエリトラフィックをリスト内で 1 番目のドメインコントローラーである **addc.lab.local** に送信します。

```
url = ldaps://addc.lab.local,ldaps://addc2.lab.local
```

addc.lab.local が利用できないためにクエリーが失敗した場合には、Identity サービスはリストに記載されている次のサーバー **addc2.lab.local** にクエリーの送信を試みます。この設定では、ラウンドロビン式にはクエリーが実行されないので、ロードバランスのソリューションとしては考慮できない点に注意してください。

2. **/etc/openldap/ldap.conf** でネットワークのタイムアウトを設定します。

```
NETWORK_TIMEOUT 2
```

また、コントローラーとドメインコントローラーの間でファイアウォールが設定されている場合には、ドメインコントローラーがダイアログを表示せずにコントローラーからのパケットを破棄してしまわないように設定すべきです。このように設定すると、**python-keystoneclient** が機能停止を適切に検知して、リスト内の次のドメインコントローラーに要求をリダイレクトすることができます。

注記

クエリーがリストの第 2 の LDAP サーバーに転送される間に接続の遅延が発生する可能性があります。これは、第 2 のサーバーへの接続を試みるには、第 1 のサーバーがタイムアウトになる必要があるためです。

1.14. トラブルシューティング

1.14.1. LDAP 接続のテスト

Active Directory Domain Controller に対してテストクエリーをリモートで実行するには、**ldapsearch** を使用します。クエリーが成功した場合には、ネットワーク接続が機能しており、AD DS サービスが稼働中であることを確認できます。以下の例では、テストクエリーはサーバー **addc.lab.local** のポート636 に対して実行されます。

```
# ldapsearch -Z -x -H ldaps://addc.lab.local:636 -D "svc-  
ldap@lab.local" -W -b "CN=Users,DC=lab,DC=local" -s sub "(cn=*)" cn
```



注記

ldapsearch は、**openldap-clients** パッケージに含まれています。このパッケージは、**# yum install openldap-clients** のコマンドを実行するとインストールすることができます。

1.14.2. ポートアクセスのテスト

nc を使用して、LDAPS ポート (636) がリモートでアクセス可能であることを確認します。この例では、サーバー **addc.lab.local** に対してプローブを実行します。ctrl-c を押してプロンプトを終了します。

```
# nc -v addc.lab.local 636  
Ncat: Version 6.40 ( http://nmap.org/ncat )  
Ncat: Connected to 192.168.200.10:636.  
^C
```

接続を確立できなかった場合には、ファイアウォールの設定に問題がある可能性があります。

第2章 IDENTITY MANAGEMENT の統合

本章では、Identity サービス (keystone) を Red Hat Identity Management と統合する方法について説明します。

以下のユースケースでは、Identity サービスが特定の Red Hat Identity Management (IdM) のユーザーを認証しつつ、Identity サービスデータベース内で承認設定および重要なサービスアカウントを保持します。

この手順を実行すると、Identity サービスは、IdM に読み取り専用でアクセスしてユーザーアカウントの認証を行う一方で、認証されたアカウントに割り当てる権限を引き続き管理するようになります。

2.1. 主要な用語

- ※ **認証**: パスワードを使用して、ユーザーが本人であることを検証するプロセス
- ※ **承認**: 認証されたユーザーに対して、アクセスしようとしているシステムの適切なパーミッションが付与されていることを確認するプロセス
- ※ **ドメイン**: Identity サービス内で設定する追加のバックエンドのことを指します。たとえば、Identity サービスは、外部の IdM 環境内のユーザーを認証するように設定することができます。このように設定されたユーザーの集合は、**ドメイン**として考えることができます。

2.2. 前提条件

本ガイドのデプロイメント例は、以下を前提としています。

- ※ Red Hat Identity Management が設定済みで、稼働していること。
- ※ Red Hat OpenStack Platform が設定済みで、稼働していること。
- ※ DNS 名前解決が完全に機能しており、かつ全ホストが適切に登録されていること。

2.3. ファイアウォールの設定

ファイアウォールが IdM と OpenStack の間のトラフィックをフィルタリングしている場合には、以下のポートを介したアクセスを許可する必要があります。

送信元	送信先	種別	ポート
OpenStack コントローラーノード	Red Hat Identity Management	LDAPS	TCP 636

2.4. 影響評価

以下のステップを実行すると、IdM ユーザーが OpenStack に対して認証を実行して、リソースにアクセスできるようになります。OpenStack のサービスアカウント (keystone、glance など) および承認管理 (パーミッションとロール) は Identity サービスのデータベースに残ります。パーミッションとロールは、Identity サービスの管理ツールを使用して IdM アカウントに割り当てられます。

2.4.1. 高可用性のオプション

この設定により、単一の IdM に依存するようになるため、Identity サービスがその IdM サーバー に対して認証できない場合には、プロジェクトユーザーが影響を受けることになります。このリスクを管理するオプションは複数あります。たとえば、**keystone** が 個別の IdM サーバーではなく DNS エイリアスやロードバランシングアプライアンスにクエリーを実行するように設定することが可能です。また、IdM サーバー の 1 つが利用できない場合には、keystone が異なる IdM サーバーにクエリーを実行するように設定することもできます。詳しくは、「[高可用性の設定](#)」を参照してください。

2.5. 停止の要件

- ※ IdM バックエンドを追加するには、Identity サービスを再起動する必要があります。
- ※ **keystone v3** に切り替えるには、全ノード上の Compute サービスを再起動する必要があります。
- ※ ユーザーは、IdM でアカウントが作成されるまでは、Dashboard にアクセスできません。ダウンタイムを短縮するには、この変更の前に十分余裕をもって IdM アカウントのプレステージを行うことを検討してください。

2.6. IDM サーバーの設定

IdM サーバーで、以下のコマンドを実行します。

1. LDAP ルックアップアカウントを作成します。このアカウントは、Identity サービスが IdM の LDAP サービスにクエリーを実行するのに使用します。

```
# kinit admin
# ipa user-add
First name: OpenStack
Last name: LDAP
User [administrator]: svc-ldap
```



注記

作成が完了したら、このアカウントのパスワード期限の設定を確認してください。

2. **grp-openstack** という名前の OpenStack ユーザーグループを作成します。OpenStack Identity でパーミッションを割り当てることができるのは、このグループのメンバーのみです。

```
# ipa group-add --desc="OpenStack Users" grp-openstack
```

3. **svc-ldap** アカウントのパスワードを設定して、**grp-openstack** グループに追加します。

```
# ipa passwd svc-ldap
# ipa group-add-member --users=svc-ldap grp-openstack
```

2.7. LDAPS 証明書の設定

1. IdM の環境で、LDAPS 証明書を見つけます。このファイルの場所は、**/etc/openldap/ldap.conf** で確認することができます。

```
TLS_CACERT /etc/ipa/ca.crt
```

2. OpenStack Identity (keystone) を実行しているノードにファイルをコピーします。たとえば、以下のコマンドは、**scp** を使用して、ca.crt を **node.lab.local** という名前のコントローラーノードにコピーします。

```
scp /etc/ipa/ca.crt root@node.lab.local:/root/
```

3. コントローラーノードで、.crt を .pem に変換します。

```
# openssl x509 -in ca.crt -out ca.pem -outform PEM
```

4. Red Hat Enterprise Linux に .pem をインストールします。

```
# cp ca.pem /etc/pki/ca-trust/source/anchors/
# update-ca-trust
```

5. ca.crt を証明書のディレクトリーにコピーします。

```
# cp ca.crt /etc/ssl/certs/
```

2.8. IDENTITY サービスの設定

以下のステップでは、IdM との統合に備えて、Identity サービスの準備を行います。

2.8.1. keystone v3 へのコマンドラインアクセスの有効化

コマンドラインから Identity サービスドメインを管理するには、**keystone v3** にアクセスする必要があります。**keystone** サービスを実行しているコントローラーから以下の手順を実行します。

1. 既存の **keystonerc_admin** ファイルのコピーを作成します。

```
# cp keystonerc_admin keystonerc_admin_v3
```

2. 新規 **keystonerc_admin_v3** ファイルを編集します。

※ **OS_AUTH_URL** の値を **v2.0** から **v3** に変更します。

```
export OS_AUTH_URL=http://controllerIP:5000/v3/
```

※ **keystonerc_admin_v3** の一番下に、以下のエントリーを追加します。

```
export OS_IDENTITY_API_VERSION=3
export OS_PROJECT_DOMAIN_NAME=Default
export OS_USER_DOMAIN_NAME=Default
```

3. **source** コマンドでこの設定ファイルを読み込み、現在のコマンドラインセッションでこれらのオプションを有効にします。

-

```
# source keystonerc_admin_v3
```

2.9. コントローラーの設定

keystone サービスを実行するコントローラーから以下の手順を実行します。

1. SELinux を設定します。

```
# setsebool -P authlogin_nsswitch_use_ldap=on
```

2. **domains** ディレクトリーを作成します。

```
# mkdir /etc/keystone/domains/
# chown keystone /etc/keystone/domains/
```

3. Identity サービスが複数のバックエンドを使用するように設定します。

```
# openstack-config --set /etc/keystone/keystone.conf identity
domain_specific_drivers_enabled true
# openstack-config --set /etc/keystone/keystone.conf identity
domain_config_dir /etc/keystone/domains
# openstack-config --set /etc/keystone/keystone.conf assignment driver
keystone.assignment.backends.sql.Assignment
```

注記

Red Hat OpenStack Platform director を使用している場合には、**/etc/keystone/keystone.conf** が Puppet で管理されている点を認識する必要があります。このため、カスタムの設定を追加しても、**openstack overcloud deploy** プロセスを実行する度に上書きされる可能性があります。上書きされた場合には、この設定を毎回手動で追加し直す必要がある場合があります。今後の director のリリースでは、デプロイ後のスクリプトを使用して、このような設定を自動的に再適用するための Puppet のパラメーターが追加される予定です。

4. Dashboard の設定ファイル **/etc/openstack-dashboard/local_settings** で複数のドメインを有効にします。

```
OPENSTACK_API_VERSIONS = {
    "identity": 3
}
OPENSTACK_KEYSTONE_MULTIDOMAIN_SUPPORT = True
OPENSTACK_KEYSTONE_DEFAULT_DOMAIN = 'Default'
```

注記

Red Hat OpenStack Platform director を使用している場合には、**/etc/openstack-dashboard/local_settings** が Puppet で管理されている点を認識する必要があります。このため、カスタムを設定を追加しても、**openstack overcloud deploy** プロセスを実行する度に上書きされる可能性があります。上書きされた場合には、この設定を毎回手動で追加し直す必要がある場合があります。今後の director のリリースでは、デプロイ後のスクリプトを使用して、このような設定を自動的に再適用するための Puppet のパラメーターが追加される予定です。

keystone および **dashboard** サービスを再起動して、設定を適用します。

```
# systemctl restart openstack-keystone.service
# systemctl restart httpd
```

5. 追加のバックエンドを設定します。

a. IdM 統合のための keystone ドメインを作成します。
新しい Identity サービスドメインに使用する名前を選択し、そのドメインを作成します。

たとえば、以下のコマンドは **LAB** という名前の **keystone** ドメインを作成します。

```
# openstack domain create LAB
```

注記

このコマンドが使用できない場合には、上記のとおりコマンドラインセッションでの **keystone v3** へのアクセスが有効化されているかどうかを確認してください。

b. 設定ファイルを作成します。

IdM バックエンドを追加するには、**/etc/keystone/domains/keystone.LAB.conf** (**LAB** は、前のステップで作成したドメイン名に置き換えます) という名前の新規ファイルに LDAP 設定を入力します。

以下の設定例は、実際に使用する IdM デプロイメントに合わせて編集する必要があります。

```
[ldap]
url = ldaps://idm.lab.local
user = uid=svc-ldap,cn=users,cn=accounts,dc=lab,dc=local
user_filter = (memberOf=cn=grp-openstack,cn=groups,cn=accounts,dc=lab,dc=local)
password = RedactedComplexPassword
user_tree_dn = cn=users,cn=accounts,dc=lab,dc=local
user_objectclass = inetUser
user_id_attribute = uid
user_name_attribute = uid
user_mail_attribute = mail
user_pass_attribute =
user_allow_create = False
user_allow_update = False
user_allow_delete = False
```

```

tls_cacertfile = /etc/ssl/certs/ca.crt

[identity]
driver = keystone.identity.backends.ldap.Identity

```

設定項目についての説明

設定	説明
url	認証に使用する IdM サーバー。LDAPS ポート 636 を使用します。
user	LDAP クエリーに使用する IdM 内のアカウント
password	上記で使した IdM アカウントのパスワード (プレーンテキスト形式)
user_filter	Identity サービスに対して提示するユーザーをフィルタリングすることにより、 grp-openstack グループのメンバーのみに Identity サービスで定義されているパーミッションを付与することができます。
user_tree_dn	IdM 内の OpenStack アカウントへのパス
user_objectclass	IdM タイプ inetUser を使用します。
user_id_attribute	ユーザー ID に使用する IdM 値をマッピングします。
user_name_attribute	names に使用する IdM 値をマッピングします。
user_mail_attribute	ユーザーのメールアドレスに使用する IdM 値をマッピングします。
user_pass_attribute	この値は、意図的に空白のままにします。

設定	説明
user_allow_create	LDAP アカウントに変更を加える機能を予期しないように、Identity サービスに通知します。
user_allow_update	LDAP アカウントに変更を加える機能を予期しないように、Identity サービスに通知します。
user_allow_delete	LDAP アカウントに変更を加える機能を予期しないように、Identity サービスに通知します。

6. 設定ファイルの所有権を **keystone** ユーザーに変更します。

```
# chown keystone /etc/keystone/domains/keystone.LAB.conf
```

7. admin ユーザーにドメインへのアクセス権を付与します。



注記

この手順を実行しても、OpenStack admin アカウントには IdM のパーミッションは付与されない点を念頭に置いてください。この場合には、ドメインという用語は、OpenStack が使用するバックエンドのことを指しています。

a. **LAB** ドメインの **ID** を取得します。

```
# openstack domain show LAB
+-----+-----+
| Field | Value |
+-----+-----+
| enabled | True |
| id      | 6800b0496429431ab1c4efbb3fe810d4 |
| name    | LAB |
+-----+-----+
```

b. admin ユーザーの **ID** の値を取得します。

```
# openstack user list --domain default | grep admin
| 3d75388d351846c6a880e53b2508172a | admin |
```

c. admin ロールの **ID** の値を取得します。

```
# openstack role list
+-----+-----+-----+
| ID | Name |
+-----+-----+-----+
| 544d48aaffde48f1b3c31a52c35f01f9 | SwiftOperator |
```

```
| 6d005d783bf0436e882c55c62457d33d | ResellerAdmin |
| 785c70b150ee4c778fe4de088070b4cf | admin         |
| 9fe2ff9ee4384b1894a90878d3e92bab | _member_     |
+-----+-----+
```

d. 返されたドメインおよび **admin** の ID を使用して、コマンドを構築し、**admin** ユーザーを **keystone** LAB ドメインの **admin** ロールに追加します。

```
# openstack role add --domain 6800b0496429431ab1c4efbb3fe810d4 --user
3d75388d351846c6a880e53b2508172a 785c70b150ee4c778fe4de088070b4cf
```

8. OpenStack サービスを再起動して、変更を適用します。

```
# systemctl restart openstack-keystone.service
```



注記

httpd サービス内で **keystone** を実行している場合には、「# **systemctl restart httpd**」のコマンドで **httpd** を再起動して **keystone** の設定を適用する必要があります。

9. コマンドで **keystone** ドメイン名を指定して、IdM ドメイン内のユーザー一覧を確認します。

```
# openstack user list --domain LAB
```

10. ローカルの **keystone** データベース内のサービスアカウントを確認します。

```
# openstack user list --domain default
```

2.10. COMPUTE が KEYSTONE V3 を使用するようにするための設定

Compute はデフォルトで **keystone v2.0** を使用するように設定されているので、マルチドメイン機能を使用するためには、**keystone v3** を使用するように設定する必要があります。

1. 各コンピュートノードとコントローラーで **keystone_authtoken** 値を変更します。

```
# openstack-config --set /etc/nova/nova.conf keystone_authtoken
auth_version v3
```

2. コントローラーでサービスを再起動して、変更を適用します。

```
# systemctl restart openstack-nova-api.service openstack-nova-
cert.service openstack-nova-conductor.service openstack-nova-
consoleauth.service openstack-nova-novncproxy.service openstack-nova-
scheduler.service
```

3. 各コンピュートノードでサービスを再起動して、変更を適用します。

```
# systemctl restart openstack-nova-compute.service
```

2.11. BLOCK STORAGE が KEYSTONE V3 を使用するようにするための設定

keystone v3 に対して認証を実行するには、Block Storage (cinder) を設定する必要もあります。

`/etc/cinder/cinder.conf` を編集します。

```
[keystone_authtoken]
auth_uri = http://controllerIP:5000/v3
auth_version = v3
```

※ `auth_uri` の `controllerIP` の箇所をコントローラーの IP アドレスに置き換えます。

2.12. IDM ユーザーによるプロジェクトへのアクセスの許可

`grp-openstack` IdM グループのメンバーである IdM ユーザーには、Dashboard 内のプロジェクトにログインするパーミッションを付与することができます。

1. IdM ユーザーの一覧を取得します。

```
# openstack user list --domain LAB
+-----+
-+-----+
| ID
| Name
+-----+
+-----+
| 1f24ec1f11aeb90520079c29f70afa060d22e2ce92b2eba7784c841ac418091e |
user1
| 12c062fidm5f8b065434d9ff6fce03eb9259537c93b411224588686e9a38bf1 |
user2
| afaf48031eb54c3e44e4cb0353f5b612084033ff70f63c22873d181fdae2e73c |
user3
| e47fc21dcf0d9716d2663766023e2d8dc15a6d9b01453854a898cabb2396826e |
user4
+-----+
+-----+
```

2. ロールの一覧を取得します。

```
# openstack role list
+-----+-----+
| ID
| Name
+-----+-----+
| 544d48aaffde48f1b3c31a52c35f01f9 | SwiftOperator |
| 6d005d783bf0436e882c55c62457d33d | ResellerAdmin |
| 785c70b150ee4c778fe4de088070b4cf | admin
| 9fe2ff9ee4384b1894a90878d3e92bab | _member_
+-----+-----+
```

3. 一覧表示されたロールの中から 1 つまたは複数のロールをユーザーに追加して、プロジェクトへのアクセス権を付与します。たとえば、`user1` を `demo` プロジェクトの一般ユーザーにするには、`_member_` ロールに追加します。

```
# openstack role add --project demo --user
1f24ec1f11aeb90520079c29f70afa060d22e2ce92b2eba7784c841ac418091e
_member_
```

また、**user1** を **demo** プロジェクトの管理ユーザーにするには、**admin** ロールに追加します。

```
# openstack role add --project demo --user
1f24ec1f11aeb90520079c29f70afa060d22e2ce92b2eba7784c841ac418091e admin
```

その結果、**user1** は IdM のユーザー名とパスワードを入力してから、ドメインのフィールドにも **LAB** と入力すると Dashboard にログインすることができます。



注記

ユーザーに「Error: Unable to retrieve container list.」というエラーメッセージが表示され、コンテナの管理が可能であることが想定されている場合には、**SwiftOperator** ロールに追加する必要があります。

2.13. DASHBOARD へのログインプロセスの変更

Identity サービスで複数のドメインを有効にすると、Dashboard のログインページに **ドメイン** という新しいフィールドが表示されます。

このフィールドは、ユーザーがログインするのに使用する認証情報と一致するドメインを入力します。このフィールドには、**keystone** にあるドメインの 1 つを手動で入力する必要があります。**openstack** コマンドで利用可能なエントリーを一覧表示します。

以下の例では、IdM アカUNTには **LAB** ドメインを指定する必要があります。**admin** のような組み込みの **keystone** アカUNTには、ドメインに **Default** を指定する必要があります。

```
# openstack domain list
+-----+-----+-----+-----+
| ID                | Name      | Enabled | Description |
|
```



```

+-----+-----+-----+-----+
| 6800b0496429431ab1c4efbb3fe810d4 | LAB      | True      |
| default                               | Default  | True      | Owns users
and tenants (i.e. projects) available on Identity API v2. |
+-----+-----+-----+-----+

```

2.14. コマンドラインへの変更

特定のコマンドでは、対象のドメインを指定する必要がある場合があります。たとえば、以下のコマンドに **--domain LAB** を追加すると、LAB ドメイン内のユーザー (**grp-openstack** グループのメンバー) が返されます。

```
# openstack user list --domain LAB
```

--domain Default を追加すると、組み込みの keystone アカウントが返されます。

```
# openstack user list --domain Default
```

2.15. IDM 統合のテスト

以下の手順では、Dashboard の機能へのユーザーアクセスをテストして、IdM の統合を検証します。

1. IdM にテストユーザーを作成し、そのユーザーを **grp-openstack** IdM グループに追加します。
2. テストユーザーを **demo** テナントの **_member_** ロールに追加します。
3. IdM テストユーザーの認証情報を使用して Dashboard にログインします。
4. 各タブをクリックし、エラーメッセージなしに正常に表示されているかどうかを確認します。
5. Dashboard を使用してテストインスタンスをビルドします。

注記

上記の手順で問題が発生した場合には、ステップ 3 から 5 までを組み込みの **admin** アカウントで実行してください。正常に実行できた場合には、OpenStack が想定通りに機能していることが実証されるので、問題は IdM と Identity の統合設定の間のどこかにあることになります。[「トラブルシューティング」](#)を参照してください。

2.16. 高可用性の設定

keystone v3 が有効化されている場合には、ドメインの設定ファイルに複数の IdM サーバーをリストして、この設定を高可用性にすることが可能です。

1. 2 番目のサーバーを **url** エントリに追加します。たとえば、**keystone.LAB.conf** ファイル内の **url** 設定を更新すると、Identity サービスは全クエリトラフィックをリスト内で 1 番目の IdM サーバーである「**idm.lab.local_**」に送信します。

■

```
url = ldaps://idm.lab.local,ldaps://idm2.lab.local
```

idm.lab.local が利用できないためにクエリーが失敗した場合には、Identity サービスはリストに記載されている次のサーバー **idm2.lab.local** にクエリーの送信を試みます。この設定では、ラウンドロビン式にはクエリーが実行されないの、ロードバランスのソリューションとしては考慮できない点に注意してください。

2. **/etc/openldap/ldap.conf** でネットワークのタイムアウトを設定します。

```
NETWORK_TIMEOUT 2
```

また、コントローラーと IdM サーバーの間でファイアウォールが設定されている場合には、IdM サーバーがダイアログを表示せずにコントローラーからのパケットを破棄してしまわないように設定すべきです。このように設定すると、**python-keystoneclient** が機能停止を適切に検知して、リスト内の次の IdM サーバーに要求をリダイレクトすることができます。

注記

クエリーがリストの第 2 の IdM サーバーに転送される間に接続の遅延が発生する可能性があります。これは、第 2 のサーバーへの接続を試みるには、第 1 のサーバーがタイムアウトになる必要があるためです。

2.17. トラブルシューティング

2.17.1. LDAP 接続のテスト

IdM サーバーに対してテストクエリーをリモートで実行するには、**ldapsearch** を使用します。クエリーが成功した場合には、ネットワーク接続が機能しており、IdM サービスが稼働中であることを確認できます。以下の例では、テストクエリーはサーバー **idm.lab.local** のポート 636 に対して実行されます。

```
# ldapsearch -D "cn=directory manager" -H ldaps://idm.lab.local:636 -b
"dc=lab,dc=local" -s sub "(objectclass=*)" -w RedactedComplexPassword
```

注記

ldapsearch は、**openldap-clients** パッケージに含まれています。このパッケージは、**yum install openldap-clients** のコマンドを実行するとインストールすることができます。

2.17.2. ポートアクセスのテスト

nc を使用して、LDAPS ポート (636) がリモートでアクセス可能であることを確認します。この例では、サーバー **idm.lab.local** に対してプローブを実行します。ctrl-c を押してプロンプトを終了します。

```
# nc -v idm.lab.local 636
Ncat: Version 6.40 ( http://nmap.org/ncat )
Ncat: Connected to 192.168.200.10:636.
^C
```

接続を確立できなかった場合には、ファイアウォールの設定に問題がある可能性があります。

第3章 汎用 LDAP の統合

本章では、Identity サービス (keystone) を汎用 LDAP 環境と統合する方法について説明します。

以下のユースケースでは、Identity サービスが特定の LDAP のユーザーを認証しつつ、Identity サービスデータベース内で承認設定および重要なサービスアカウントを保持します。この手順を実行すると、Identity サービスは、LDAP に読み取り専用でアクセスしてユーザーアカウントの認証を行う一方で、認証されたアカウントに割り当てる権限を引き続き管理するようになります。

3.1. 主要な用語

- ※ **認証**: パスワードを使用して、ユーザーが本人であることを検証するプロセス
- ※ **承認**: 認証されたユーザーに対して、アクセスしようとしているシステムの適切なパーミッションが付与されていることを確認するプロセス
- ※ **ドメイン**: Identity サービス内で設定する追加のバックエンドのことを指します。たとえば、Identity サービスは、外部の LDAP 環境内のユーザーを認証するように設定することができます。このように設定されたユーザーの集合は、**ドメイン**として考えることができます。

3.2. 前提条件

本ガイドのデプロイメント例は、以下を前提としています。

- ※ LDAP サーバーが設定済みで、稼働していること。
- ※ Red Hat OpenStack Platform が設定済みで、稼働していること。
- ※ DNS 名前解決が完全に機能しており、かつ全ホストが適切に登録されていること。

3.3. ファイアウォールの設定

ファイアウォールが LDAP と OpenStack の間のトラフィックをフィルタリングしている場合には、以下のポートを介したアクセスを許可する必要があります。

送信元	送信先	種別	ポート
OpenStack コントローラーノード	LDAP サーバー	LDAPS	TCP 636

3.4. 影響評価

以下のステップを実行すると、LDAP ユーザーが OpenStack に対して認証を実行して、リソースにアクセスできるようになります。OpenStack のサービスアカウント (keystone、glance など) および承認管理 (パーミッションとロール) は Identity サービスのデータベースに残ります。パーミッションとロールは、Identity サービスの管理ツールを使用して LDAP アカウントに割り当てられます。

3.4.1. 高可用性のオプション

この設定により、単一の LDAP に依存するようになるため、Identity サービスがその LDAP サーバーに対して認証できない場合には、プロジェクトユーザーが影響を受けることになります。このリスクを管理するオプションは複数あります。たとえば、**keystone** が個別の IdM サーバーではなく DNS エイリアスやロードバランシングアプライアンスにクエリーを実行するように設定することが可能です。また、IdM サーバーの 1 つが利用できない場合には、keystone が異なる IdM サーバーにクエリーを実行するように設定することもできます。詳しくは、「[高可用性の設定](#)」を参照してください。

3.5. 停止の要件

- ※ LDAP バックエンドを追加するには、Identity サービスを再起動する必要があります。
- ※ **keystone v3** に切り替えるには、全ノード上の Compute サービスを再起動する必要があります。
- ※ ユーザーは、LDAP でアカウントが作成されるまでは、Dashboard にアクセスできません。ダウンタイムを短縮するには、この変更の前に十分余裕をもって LDAP アカウントのプレステージを行うことを検討してください。

3.6. LDAP サーバーの設定

LDAP サーバーで以下のステップを実行して、Identity サービスの統合の準備をします。

1. LDAP ルックアップアカウントを作成します。

このアカウントは、Identity サービスが LDAP サービスにクエリーを実行するのに使用します。このデプロイメント例では、標準のパスワードポリシーの要件に加えて、以下の属性が必要となります。

- ※ **First name:** OpenStack
- ※ **Last name:** LDAP
- ※ **User name:** svc-ldap



注記

作成が完了したら、このアカウントのパスワード期限の設定を確認してください。

2. grp-openstack という名前の OpenStack ユーザーの LDAP グループを作成します。OpenStack Identity でパーミッションを割り当てることができるのは、このグループのメンバーのみです。

3. svc-ldap アカウントのパスワードを設定して、**grp-openstack** グループに追加します。

3.7. LDAPS 証明書の設定

1. LDAP の環境で、LDAPS 証明書を見つけます。このファイルの場所は、`/etc/openldap/ldap.conf` で確認することができます。

```
TLS_CACERT /etc/ipa/ca.crt
```

2. OpenStack Identity (keystone) を実行しているノードにファイルをコピーします。たとえば、以下のコマンドは、**scp** を使用して、ca.crt を **node.lab.local** という名前のコントローラーノードにコピーします。

```
scp /etc/ipa/ca.crt root@node.lab.local:/root/
```

3. コントローラーノードで、.crt を .pem に変換します。

```
# openssl x509 -in ca.crt -out ca.pem -outform PEM
```

4. Red Hat Enterprise Linux に .pem をインストールします。

```
# cp ca.pem /etc/pki/ca-trust/source/anchors/
# update-ca-trust
```

5. ca.crt を証明書のディレクトリーにコピーします。

```
# cp ca.crt /etc/ssl/certs/
```

3.8. IDENTITY サービスの設定

以下のステップでは、LDAP との統合に備えて、Identity サービスの準備を行います。

3.8.1. keystone v3 へのコマンドラインアクセスの有効化

コマンドラインから Identity サービスドメインを管理するには、**keystone v3** にアクセスする必要があります。**keystone** サービスを実行しているコントローラーから以下の手順を実行します。

1. 既存の **keystonerc_admin** ファイルのコピーを作成します。

```
# cp keystonerc_admin keystonerc_admin_v3
```

2. 新規 **keystonerc_admin_v3** ファイルを編集します。

※ **OS_AUTH_URL** の値を **v2.0** から **v3** に変更します。

```
export OS_AUTH_URL=http://controllerIP:5000/v3/
```

※ **keystonerc_admin_v3** の一番下に、以下のエントリーを追加します。

```
export OS_IDENTITY_API_VERSION=3
export OS_PROJECT_DOMAIN_NAME=Default
export OS_USER_DOMAIN_NAME=Default
```

3. **source** コマンドでこの設定ファイルを読み込み、現在のコマンドラインセッションでこれらのオプションを有効にします。

```
# source keystonerc_admin_v3
```

3.9. コントローラーの設定

keystone サービスを実行するコントローラーから以下の手順を実行します。

1. SELinux を設定します。

```
# setsebool -P authlogin_nsswitch_use_ldap=on
```

2. **domains** ディレクトリーを作成します。

```
# mkdir /etc/keystone/domains/
# chown keystone /etc/keystone/domains/
```

3. Identity サービスが複数のバックエンドを使用するように設定します。

```
# openstack-config --set /etc/keystone/keystone.conf identity
domain_specific_drivers_enabled true
# openstack-config --set /etc/keystone/keystone.conf identity
domain_config_dir /etc/keystone/domains
# openstack-config --set /etc/keystone/keystone.conf assignment driver
keystone.assignment.backends.sql.Assignment
```

注記

Red Hat OpenStack Platform director を使用している場合には、**/etc/keystone/keystone.conf** が Puppet で管理されている点を認識する必要があります。このため、カスタムの設定を追加しても、**openstack overcloud deploy** プロセスを実行する度に上書きされる可能性があります。上書きされた場合には、この設定を毎回手動で追加し直す必要がある場合があります。今後の director のリリースでは、デプロイ後のスクリプトを使用して、このような設定を自動的に再適用するための Puppet のパラメーターが追加される予定です。

4. Dashboard の設定ファイル **/etc/openstack-dashboard/local_settings** で複数のドメインを有効にします。

```
OPENSTACK_API_VERSIONS = {
    "identity": 3
}
OPENSTACK_KEYSTONE_MULTIDOMAIN_SUPPORT = True
OPENSTACK_KEYSTONE_DEFAULT_DOMAIN = 'Default'
```

注記

Red Hat OpenStack Platform director を使用している場合には、**/etc/openstack-dashboard/local_settings** が Puppet で管理されている点を認識する必要があります。このため、カスタムの設定を追加しても、**openstack overcloud deploy** プロセスを実行する度に上書きされる可能性があります。上書きされた場合には、この設定を毎回手動で追加し直す必要がある場合があります。今後の director のリリースでは、デプロイ後のスクリプトを使用して、このような設定を自動的に再適用するための Puppet のパラメーターが追加される予定です。

keystone および **dashboard** サービスを再起動して、設定を適用します。

```
# systemctl restart openstack-keystone.service
# systemctl restart httpd
```

5. 追加のバックエンドを設定します。

a. LDAP 統合のための keystone ドメインを作成します。

新しい Identity サービスドメインに使用する名前を選択し、そのドメインを作成します。

たとえば、以下のコマンドは **LAB** という名前の **keystone** ドメインを作成します。

```
# openstack domain create LAB
```



注記

このコマンドが使用できない場合には、上記のとおりコマンドラインセッションでの **keystone v3** へのアクセスが有効化されているかどうかを確認してください。

b. 設定ファイルを作成します。

LDAP バックエンドを追加するには、**/etc/keystone/domains/keystone.LAB.conf** (**LAB** は、前のステップで作成したドメイン名に置き換えます) という名前の新規ファイルに LDAP 設定を入力します。

以下の設定例は、実際に使用する LDAP デプロイメントに合わせて編集する必要があります。

```
[ldap]
url = ldaps://ldap.lab.local
user = uid=svc-ldap,cn=users,cn=accounts,dc=lab,dc=local
user_filter = (memberOf=cn=grp-
openstack,cn=groups,cn=accounts,dc=lab,dc=local)
password = RedactedComplexPassword
user_tree_dn = cn=users,cn=accounts,dc=lab,dc=local
user_objectclass = inetUser
user_id_attribute = uid
user_name_attribute = uid
user_mail_attribute = mail
user_pass_attribute =
user_allow_create = False
user_allow_update = False
user_allow_delete = False
tls_cacertfile = /etc/ssl/certs/ca.crt

[identity]
driver = keystone.identity.backends.ldap.Identity
```

設定項目についての説明

設定	説明
url	認証に使用する LDAP サーバー。LDAPS ポート 636 を使用します。

設定	説明
user	LDAP クエリーに使用する LDAP 内のアカウント
password	上記で使した LDAP アカウントのパスワード (プレーンテキスト形式)
user_filter	Identity サービスに対して提示するユーザーをフィルタリングすることにより、 grp-openstack グループのメンバーのみに Identity サービスで定義されているパーミッションを付与することができます。
user_tree_dn	LDAP 内の OpenStack アカウントへのパス
user_objectclass	LDAP タイプ inetUser を使用します。
user_id_attribute	ユーザー ID に使用する LDAP 値をマッピングします。
user_name_attribute	names に使用する LDAP 値をマッピングします。
user_mail_attribute	ユーザーのメールアドレスに使用する LDAP 値をマッピングします。
user_pass_attribute	この値は、意図的に空白のままにします。
user_allow_create	LDAP アカウントに変更を加える機能を予期しないように、Identity サービスに通知します。
user_allow_update	LDAP アカウントに変更を加える機能を予期しないように、Identity サービスに通知します。
user_allow_delete	LDAP アカウントに変更を加える機能を予期しないように、Identity サービスに通知します。

6. 設定ファイルの所有権を **keystone** ユーザーに変更します。

```
# chown keystone /etc/keystone/domains/keystone.LAB.conf
```

7. admin ユーザーにドメインへのアクセス権を付与します。



注記

この手順を実行しても、OpenStack admin アカウントには LDAP 内のパーミッションは付与されない点を念頭に置いてください。この場合には、ドメインという用語は、OpenStack が使用するバックエンドのことを指しています。

a. LAB ドメインの ID を取得します。

```
# openstack domain show LAB
+-----+-----+
| Field | Value |
+-----+-----+
| enabled | True |
| id      | 6800b0496429431ab1c4efbb3fe810d4 |
| name    | LAB |
+-----+-----+
```

b. admin ユーザーの ID の値を取得します。

```
# openstack user list --domain default | grep admin
| 3d75388d351846c6a880e53b2508172a | admin |
```

c. admin ロールの ID の値を取得します。

```
# openstack role list
+-----+-----+-----+
| ID | Name |
+-----+-----+-----+
| 544d48aaffde48f1b3c31a52c35f01f9 | SwiftOperator |
| 6d005d783bf0436e882c55c62457d33d | ResellerAdmin |
| 785c70b150ee4c778fe4de088070b4cf | admin |
| 9fe2ff9ee4384b1894a90878d3e92bab | _member_ |
+-----+-----+-----+
```

d. 返されたドメインおよび admin の ID を使用して、コマンドを構築し、admin ユーザーを keystone LAB ドメインの admin ロールに追加します。

```
# openstack role add --domain 6800b0496429431ab1c4efbb3fe810d4 --user
3d75388d351846c6a880e53b2508172a 785c70b150ee4c778fe4de088070b4cf
```

8. OpenStack サービスを再起動して、変更を適用します。

```
# systemctl restart openstack-keystone.service
```

**注記**

httpd サービス内で keystone を実行している場合には、「# **systemctl restart httpd**」のコマンドで httpd を再起動して keystone の設定を適用する必要があります。

9. コマンドで **keystone** ドメイン名を指定して、LDAP ドメイン内のユーザー一覧を確認します。

```
# openstack user list --domain LAB
```

10. ローカルの keystone データベース内のサービスアカウントを確認します。

```
# openstack user list --domain default
```

3.10. COMPUTE が KEYSTONE V3 を使用するようにするための設定

Compute はデフォルトで **keystone v2.0** を使用するように設定されているので、マルチドメイン機能を使用するためには、**keystone v3** を使用するように設定する必要があります。

1. 各コンピュートノードとコントローラーで **keystone_authtoken** 値を変更します。

```
# openstack-config --set /etc/nova/nova.conf keystone_authtoken
auth_version v3
```

2. コントローラーでサービスを再起動して、変更を適用します。

```
# systemctl restart openstack-nova-api.service openstack-nova-
cert.service openstack-nova-conductor.service openstack-nova-
consoleauth.service openstack-nova-novncproxy.service openstack-nova-
scheduler.service
```

3. 各コンピュートノードでサービスを再起動して、変更を適用します。

```
# systemctl restart openstack-nova-compute.service
```

3.11. BLOCK STORAGE が KEYSTONE V3 を使用するようにするための設定

keystone v3 に対して認証を実行するには、Block Storage (cinder) を設定する必要もあります。

/etc/cinder/cinder.conf を編集します。

```
[keystone_authtoken]
auth_uri = http://controllerIP:5000/v3
auth_version = v3
```

※ **auth_uri** の **controllerIP** の箇所をコントローラーの IP アドレスに置き換えます。

3.12. LDAP ユーザーによるプロジェクトへのアクセスの許可

grp-openstack LDAP グループのメンバーである LDAP ユーザーには、Dashboard 内のプロジェクトにログインするパーミッションを付与することができます。

1. LDAP ユーザーの一覧を取得します。

```
# openstack user list --domain LAB
+-----+
+-----+
| ID
| Name
+-----+
+-----+
| 1f24ec1f11aeb90520079c29f70afa060d22e2ce92b2eba7784c841ac418091e |
user1
| 12c062fLDAP5f8b065434d9ff6f6ce03eb9259537c93b411224588686e9a38bf1 |
user2
| afaf48031eb54c3e44e4cb0353f5b612084033ff70f63c22873d181fdae2e73c |
user3
| e47fc21dcf0d9716d2663766023e2d8dc15a6d9b01453854a898cabb2396826e |
user4
+-----+
+-----+
```

2. ロールの一覧を取得します。

```
# openstack role list
+-----+-----+
| ID
| Name
+-----+-----+
| 544d48aaffde48f1b3c31a52c35f01f9 | SwiftOperator |
| 6d005d783bf0436e882c55c62457d33d | ResellerAdmin |
| 785c70b150ee4c778fe4de088070b4cf | admin
| 9fe2ff9ee4384b1894a90878d3e92bab | _member_
+-----+-----+
```

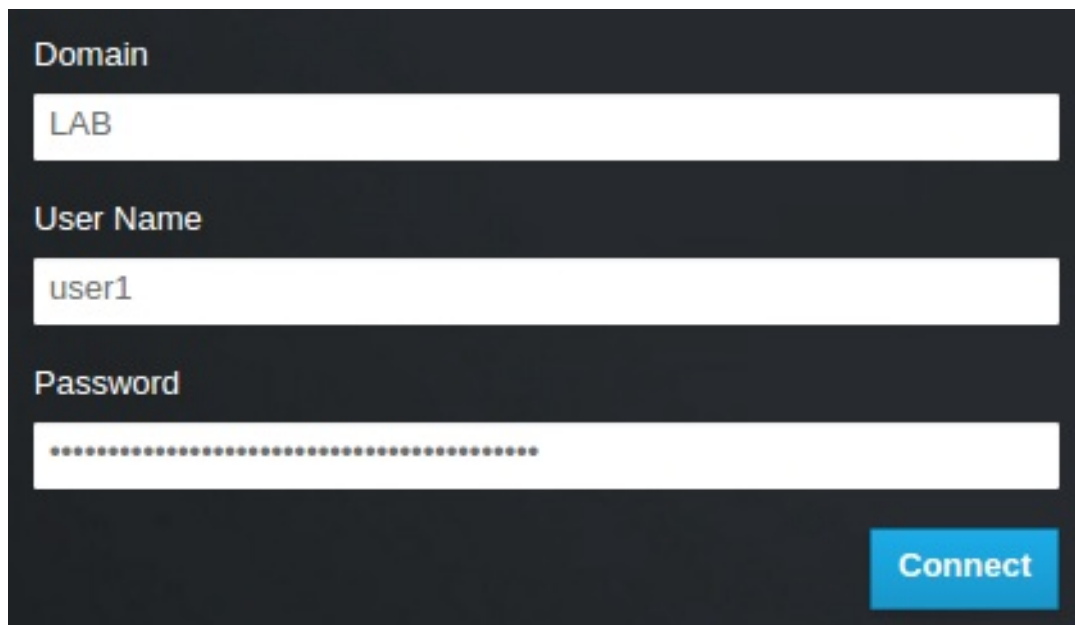
3. 一覧表示されたロールの中から 1 つまたは複数のロールをユーザーに追加して、プロジェクトへのアクセス権を付与します。たとえば、**user1** を **demo** プロジェクトの一般ユーザーにするには、**_member_** ロールに追加します。

```
# openstack role add --project demo --user
1f24ec1f11aeb90520079c29f70afa060d22e2ce92b2eba7784c841ac418091e
_member_
```

また、**user1** を **demo** プロジェクトの管理ユーザーにするには、**admin** ロールに追加します。

```
# openstack role add --project demo --user
1f24ec1f11aeb90520079c29f70afa060d22e2ce92b2eba7784c841ac418091e admin
```

その結果、**user1** は LDAP のユーザー名とパスワードを入力してから、ドメインのフィールドにも **LAB** と入力すると Dashboard にログインすることができます。



Domain

LAB

User Name

user1

Password

.....

Connect



注記

ユーザーに「Error: Unable to retrieve container list.」というエラーメッセージが表示され、コンテナの管理が可能であることが想定されている場合には、**SwiftOperator** ロールに追加する必要があります。

3.13. DASHBOARD へのログインプロセスの変更

Identity サービスで複数のドメインを有効にすると、Dashboard のログインページに **ドメイン** という新しいフィールドが表示されます。

このフィールドは、ユーザーがログインするのに使用する認証情報と一致するドメインを入力します。このフィールドには、**keystone** にあるドメインの 1 つを手動で入力する必要があります。**openstack** コマンドで利用可能なエントリーを一覧表示します。

以下の例では、LDAP アカウントには **LAB** ドメインを指定する必要があります。**admin** のような組み込みの **keystone** アカウントには、ドメインに **Default** を指定する必要があります。

```
# openstack domain list
+-----+-----+-----+-----+
| ID                | Name    | Enabled | Description |
+-----+-----+-----+-----+
| 6800b0496429431ab1c4efbb3fe810d4 | LAB     | True    |             |
| default           | Default | True    | Owns users and tenants (i.e. projects) available on Identity API v2. |
+-----+-----+-----+-----+
```

3.14. コマンドラインへの変更

特定のコマンドでは、対象のドメインを指定する必要がある場合があります。たとえば、以下のコマンドに **--domain LAB** を追加すると、LAB ドメイン内のユーザー (**grp-openstack** グループのメンバー) が返されます。

```
# openstack user list --domain LAB
```

--domain Default を追加すると、組み込みの keystone アカウントが返されます。

```
# openstack user list --domain Default
```

3.15. LDAP 統合のテスト

以下の手順では、Dashboard の機能へのユーザーアクセスをテストして、LDAP の統合を検証します。

1. LDAP にテストユーザーを作成し、そのユーザーを **grp-openstack** LDAP グループに追加します。
2. テストユーザーを **demo** テナントの **_member_** ロールに追加します。
3. LDAP テストユーザーの認証情報を使用して Dashboard にログインします。
4. 各タブをクリックし、エラーメッセージなしに正常に表示されているかどうかを確認します。
5. Dashboard を使用してテストインスタンスをビルドします。

注記

上記の手順で問題が発生した場合には、組み込みの **admin** アカウントでステップ 3 から 5 までを実行してください。正常に実行できた場合には、OpenStack が想定通りに機能していることが実証されるので、問題は LDAP と Identity の統合設定の間のどこかにあることになります。[「トラブルシューティング」](#)を参照してください。

3.16. 高可用性の設定

keystone v3 が有効化されている場合には、ドメインの設定ファイルに複数の LDAP サーバーをリストして、この設定を高可用性にすることが可能です。

1. 2 番目のサーバーを **url** エントリーに追加します。たとえば、**keystone.LAB.conf** ファイル内の **url** 設定を更新すると、Identity サービスは全クエリトラフィックをリスト内で 1 番目の LDAP サーバーである「**ldap.lab.local**」に送信します。

```
url = ldaps://ldap.lab.local,ldaps://ldap2.lab.local
```

ldap.lab.local が利用できないためにクエリーが失敗した場合には、Identity サービスはリストに記載されている次のサーバー **ldap2.lab.local** にクエリーの送信を試みます。この設定では、ラウンドロビン式にはクエリーが実行されないの、ロードバランスのソリューションとしては考慮できない点に注意してください。

2. **/etc/openldap/ldap.conf** でネットワークのタイムアウトを設定します。

```
NETWORK_TIMEOUT 2
```

また、コントローラーと LDAP サーバーの間でファイアウォールが設定されている場合には、LDAP サーバーがダイアログを表示せずにコントローラーからのパケットを破棄してしまわないように設定すべきです。このように設定すると、**python-keystoneclient** が機能停止を適切に検知して、リスト内の次の LDAP サーバーに要求をリダイレクトすることができます。

注記

クエリーがリストの第 2 の LDAP サーバーに転送される間に接続の遅延が発生する可能性があります。これは、第 2 のサーバーへの接続を試みるには、第 1 のサーバーがタイムアウトになる必要があるためです。

3.17. トラブルシューティング

3.17.1. LDAP 接続のテスト

LDAP サーバーに対してテストクエリーをリモートで実行するには、**ldapsearch** を使用します。クエリーが成功した場合には、ネットワーク接続が機能しており、LDAP サービスが稼働中であることを確認できます。以下の例では、テストクエリーはサーバー **ldap.lab.local** のポート 636 に対して実行されます。

```
# ldapsearch -D "cn=directory manager" -H ldaps://ldap.lab.local:636 -b
"dc=lab,dc=local" -s sub "(objectclass=*)" -w RedactedComplexPassword
```

注記

ldapsearch は、**openldap-clients** パッケージに含まれています。このパッケージは、**yum install openldap-clients** のコマンドを実行するとインストールすることができます。

3.17.2. ポートアクセスのテスト

nc を使用して、LDAPS ポート (636) がリモートでアクセス可能であることを確認します。この例では、サーバー **ldap.lab.local** に対してプローブを実行します。ctrl-c を押してプロンプトを終了します。

```
# nc -v ldap.lab.local 636
Ncat: Version 6.40 ( http://nmap.org/ncat )
Ncat: Connected to 192.168.200.10:636.
^C
```

接続を確立できなかった場合には、ファイアウォールの設定に問題がある可能性があります。