Integrate with Identity Service

Use Active Directory, IdM, or generic LDAP as an external authentication back end
Use Active Directory, IdM, or generic LDAP as an external authentication back end

OpenStack Team
rhos-docs@redhat.com
Abstract

Use Active Directory, IdM, or generic LDAP as an external authentication back end
# Table of Contents

PREFACE ............................................................................................................. 4

## CHAPTER 1. ACTIVE DIRECTORY INTEGRATION ............................................. 5
1.1. KEY TERMS ................................................................................................. 5
1.2. ASSUMPTIONS ............................................................................................. 5
1.3. IMPACT STATEMENT .................................................................................. 5
1.3.1. High Availability options ........................................................................ 5
1.4. OUTAGE REQUIREMENTS ........................................................................ 6
1.5. FIREWALL CONFIGURATION ......................................................................... 6
1.6. CONFIGURE ACTIVE DIRECTORY DOMAIN SERVICES ................................ 6
1.7. CONFIGURE THE LDAPS CERTIFICATE .................................................. 8
1.8. CONFIGURE IDENTITY SERVICE ................................................................. 8
1.8.1. Enable command line access to keystone v3 ........................................ 8
1.8.2. Configure the controller ......................................................................... 8
1.8.3. Configure Compute to use keystone v3 ............................................... 14
1.8.4. Configure Block Storage to use keystone v3 ....................................... 15
1.8.5. Allow Active Directory group members to access Projects ................... 15
1.8.6. Allow Active Directory users to access Projects .................................... 16
1.9. GRANT ACCESS TO THE DOMAIN TAB .................................................. 17
1.10. CREATING A NEW PROJECT ................................................................. 17
1.11. CHANGES TO THE COMMAND LINE ................................................... 18
1.12. TEST AD DS INTEGRATION ...................................................................... 18
1.13. CONFIGURE FOR HIGH AVAILABILITY .................................................. 18
1.14. CREATE A RC FILE FOR A NON-ADMIN USER ....................................... 19
1.15. TROUBLESHOOTING .............................................................................. 19
1.15.1. Test LDAP connections .................................................................... 19
1.15.2. Test the Certificate Trust Configuration ............................................. 20
1.15.3. Test port access ................................................................................... 20

## CHAPTER 2. IDENTITY MANAGEMENT INTEGRATION ................................. 21
2.1. KEY TERMS ................................................................................................. 21
2.2. ASSUMPTIONS ............................................................................................. 21
2.3. IMPACT STATEMENT .................................................................................. 21
2.3.1. High Availability options ........................................................................ 21
2.4. OUTAGE REQUIREMENTS ........................................................................ 22
2.5. FIREWALL CONFIGURATION ......................................................................... 22
2.6. CONFIGURE THE IDM SERVER .................................................................. 22
2.7. CONFIGURE THE LDAPS CERTIFICATE .................................................. 22
2.8. CONFIGURE IDENTITY SERVICE ................................................................. 23
2.8.1. Enable command line access to keystone v3 ........................................ 23
2.8.2. Configure the controller ......................................................................... 24
2.8.3. Configure Compute to use keystone v3 ............................................... 27
2.8.4. Configure Block Storage to use keystone v3 ....................................... 28
2.8.5. Allow IdM users to access Projects ....................................................... 28
2.9. GRANT ACCESS TO THE DOMAIN TAB .................................................. 29
2.10. CREATING A NEW PROJECT ................................................................. 29
2.10.1. Changes to the command line ............................................................. 30
2.10.2. Test IdM integration ............................................................................ 30
2.11. CONFIGURE FOR HIGH AVAILABILITY .................................................. 30
2.12. CREATE A RC FILE FOR A NON-ADMIN USER ....................................... 31
2.13. TROUBLESHOOTING .............................................................................. 31
2.13.1. Test LDAP connections .................................................................... 31

---

Table: Table of Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>PREFACE</td>
<td>4</td>
</tr>
<tr>
<td>CHAPTER 1. ACTIVE DIRECTORY INTEGRATION</td>
<td>5</td>
</tr>
<tr>
<td>1.1. KEY TERMS</td>
<td>5</td>
</tr>
<tr>
<td>1.2. ASSUMPTIONS</td>
<td>5</td>
</tr>
<tr>
<td>1.3. IMPACT STATEMENT</td>
<td>5</td>
</tr>
<tr>
<td>1.3.1. High Availability options</td>
<td>5</td>
</tr>
<tr>
<td>1.4. OUTAGE REQUIREMENTS</td>
<td>6</td>
</tr>
<tr>
<td>1.5. FIREWALL CONFIGURATION</td>
<td>6</td>
</tr>
<tr>
<td>1.6. CONFIGURE ACTIVE DIRECTORY DOMAIN SERVICES</td>
<td>6</td>
</tr>
<tr>
<td>1.7. CONFIGURE THE LDAPS CERTIFICATE</td>
<td>8</td>
</tr>
<tr>
<td>1.8. CONFIGURE IDENTITY SERVICE</td>
<td>8</td>
</tr>
<tr>
<td>1.8.1. Enable command line access to keystone v3</td>
<td>8</td>
</tr>
<tr>
<td>1.8.2. Configure the controller</td>
<td>8</td>
</tr>
<tr>
<td>1.8.3. Configure Compute to use keystone v3</td>
<td>14</td>
</tr>
<tr>
<td>1.8.4. Configure Block Storage to use keystone v3</td>
<td>15</td>
</tr>
<tr>
<td>1.8.5. Allow Active Directory group members to access Projects</td>
<td>15</td>
</tr>
<tr>
<td>1.8.6. Allow Active Directory users to access Projects</td>
<td>16</td>
</tr>
<tr>
<td>1.9. GRANT ACCESS TO THE DOMAIN TAB</td>
<td>17</td>
</tr>
<tr>
<td>1.10. CREATING A NEW PROJECT</td>
<td>17</td>
</tr>
<tr>
<td>1.11. CHANGES TO THE COMMAND LINE</td>
<td>18</td>
</tr>
<tr>
<td>1.12. TEST AD DS INTEGRATION</td>
<td>18</td>
</tr>
<tr>
<td>1.13. CONFIGURE FOR HIGH AVAILABILITY</td>
<td>18</td>
</tr>
<tr>
<td>1.14. CREATE A RC FILE FOR A NON-ADMIN USER</td>
<td>19</td>
</tr>
<tr>
<td>1.15. TROUBLESHOOTING</td>
<td>19</td>
</tr>
<tr>
<td>1.15.1. Test LDAP connections</td>
<td>19</td>
</tr>
<tr>
<td>1.15.2. Test the Certificate Trust Configuration</td>
<td>20</td>
</tr>
<tr>
<td>1.15.3. Test port access</td>
<td>20</td>
</tr>
<tr>
<td>CHAPTER 2. IDENTITY MANAGEMENT INTEGRATION</td>
<td>21</td>
</tr>
<tr>
<td>2.1. KEY TERMS</td>
<td>21</td>
</tr>
<tr>
<td>2.2. ASSUMPTIONS</td>
<td>21</td>
</tr>
<tr>
<td>2.3. IMPACT STATEMENT</td>
<td>21</td>
</tr>
<tr>
<td>2.3.1. High Availability options</td>
<td>21</td>
</tr>
<tr>
<td>2.4. OUTAGE REQUIREMENTS</td>
<td>22</td>
</tr>
<tr>
<td>2.5. FIREWALL CONFIGURATION</td>
<td>22</td>
</tr>
<tr>
<td>2.6. CONFIGURE THE IDM SERVER</td>
<td>22</td>
</tr>
<tr>
<td>2.7. CONFIGURE THE LDAPS CERTIFICATE</td>
<td>22</td>
</tr>
<tr>
<td>2.8. CONFIGURE IDENTITY SERVICE</td>
<td>23</td>
</tr>
<tr>
<td>2.8.1. Enable command line access to keystone v3</td>
<td>23</td>
</tr>
<tr>
<td>2.8.2. Configure the controller</td>
<td>24</td>
</tr>
<tr>
<td>2.8.3. Configure Compute to use keystone v3</td>
<td>27</td>
</tr>
<tr>
<td>2.8.4. Configure Block Storage to use keystone v3</td>
<td>28</td>
</tr>
<tr>
<td>2.8.5. Allow IdM users to access Projects</td>
<td>28</td>
</tr>
<tr>
<td>2.9. GRANT ACCESS TO THE DOMAIN TAB</td>
<td>29</td>
</tr>
<tr>
<td>2.10. CREATING A NEW PROJECT</td>
<td>29</td>
</tr>
<tr>
<td>2.10.1. Changes to the command line</td>
<td>30</td>
</tr>
<tr>
<td>2.10.2. Test IdM integration</td>
<td>30</td>
</tr>
<tr>
<td>2.11. CONFIGURE FOR HIGH AVAILABILITY</td>
<td>30</td>
</tr>
<tr>
<td>2.12. CREATE A RC FILE FOR A NON-ADMIN USER</td>
<td>31</td>
</tr>
<tr>
<td>2.13. TROUBLESHOOTING</td>
<td>31</td>
</tr>
<tr>
<td>2.13.1. Test LDAP connections</td>
<td>31</td>
</tr>
</tbody>
</table>
2.13.2. Test port access

CHAPTER 3. GENERIC LDAP INTEGRATION .................................................. 33

3.1. KEY TERMS 33
3.2. ASSUMPTIONS 33
3.3. IMPACT STATEMENT 33
  3.3.1. High Availability options 33
3.4. OUTAGE REQUIREMENTS 34
3.5. FIREWALL CONFIGURATION 34
3.6. CONFIGURE THE LDAP SERVER 34
3.7. CONFIGURE THE LDAPS CERTIFICATE 34
3.8. CONFIGURE IDENTITY SERVICE 35
  3.8.1. Enable command line access to keystone v3 35
  3.8.2. Configure the controller 35
  3.8.3. Configure Compute to use keystone v3 39
  3.8.4. Configure Block Storage to use keystone v3 40
  3.8.5. Allow LDAP users to access Projects 40
3.9. GRANT ACCESS TO THE DOMAIN TAB 41
3.10. CREATING A NEW PROJECT 41
  3.10.1. Changes to the command line 42
  3.10.2. Test LDAP integration 42
3.11. CONFIGURE FOR HIGH AVAILABILITY 42
3.12. CREATE A RC FILE FOR A NON-ADMIN USER 43
3.13. TROUBLESHOOTING 43
  3.13.1. Test LDAP connections 43
  3.13.2. Test port access 44
PREFACE

Identity Service (codename keystone) provides authentication and authorization for Red Hat OpenStack Platform 8.

This guide describes how to integrate Identity Service with Microsoft Active Directory Domain Service (AD DS) and Red Hat Identity Management (IdM).
CHAPTER 1. ACTIVE DIRECTORY INTEGRATION

This chapter describes how to integrate Identity Service (keystone) with Active Directory Domain Services. In this use case, Identity Service authenticates certain Active Directory Domain Services (AD DS) users, while retaining authorization settings and critical service accounts in the Identity Service database. As a result, Identity Service has read-only access to AD DS for user account authentication, while retaining management over the privileges assigned to authenticated accounts.

1.1. KEY TERMS

- **Authentication** - The process of using a password to verify that the user is who they claim to be.
- **Authorization** - Validating that authenticated users have proper permissions to the resources they are attempting to access.
- **Domain** - This term is not the same as an AD DS domain, and instead refers to the additional namespaces that are configured in Identity Service for partitioning users, groups, and projects. These separate domains can be configured to authenticate users in different LDAP or AD DS environments.

1.2. ASSUMPTIONS

This example deployment makes the following assumptions:

- Active Directory Domain Services is configured and operational.
- Red Hat OpenStack Platform is configured and operational.
- DNS name resolution is fully functional and all hosts are registered appropriately.
- AD DS authentication traffic is encrypted with LDAPS, using port 636.

**IMPORTANT**

Multidomain Dashboard configuration is not supported in this version of Red Hat OpenStack Platform. As a result, this guide only describes Dashboard configuration for a single domain.

1.3. IMPACT STATEMENT

These steps allow AD DS users to authenticate to OpenStack and access resources. OpenStack service accounts (such as keystone and glance), and authorization management (permissions, roles, projects) will remain in the Identity Service database. Permissions and roles are assigned to the AD DS accounts using Identity Service management tools.

1.3.1. High Availability options

This configuration creates a dependency on the availability of a single Active Directory Domain Controller; Project users will be affected if Identity Service is unable to authenticate to the AD Domain Controller. A number of options are available to manage this risk; for example, you might configure Identity Service to query a DNS alias or a load balancing appliance, rather than an individual AD Domain Controller. You can also configure keystone to query a different Domain Controller, should one become unavailable. See Section 1.13, “Configure for high availability” for more information.
1.4. OUTAGE REQUIREMENTS

- The Identity Service will need to be restarted to add the AD DS back end.
- The Compute services on all nodes will need to be restarted in order to switch over to keystone v3.
- Users will be unable to access the dashboard until their accounts have been created in AD DS. To reduce downtime, consider pre-staging the AD DS accounts well in advance of this change.

1.5. FIREWALL CONFIGURATION

If firewalls are filtering traffic between AD DS and OpenStack, you will need to allow access through the following port:

<table>
<thead>
<tr>
<th>Source</th>
<th>Destination</th>
<th>Type</th>
<th>Port</th>
</tr>
</thead>
<tbody>
<tr>
<td>OpenStack Controller</td>
<td>Active Directory</td>
<td>LDAPS</td>
<td>TCP 636</td>
</tr>
<tr>
<td>Node</td>
<td>Domain Controller</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1.6. CONFIGURE ACTIVE DIRECTORY DOMAIN SERVICES

This section describes the tasks that Active Directory administrators will need to complete:

Table 1.1. Configuration steps

<table>
<thead>
<tr>
<th>Task</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create a service account.</td>
<td>This can be named according to your naming convention for service accounts, for example: svc-ldap. This can be a regular domain user account. Administrator privileges are not required.</td>
</tr>
<tr>
<td>Create a user group.</td>
<td>If a user needs access to OpenStack, they must be a member of this group. This can be named according to your naming convention for user groups, for example: grp-openstack. Members of this group can be granted access to Projects in the dashboard, if they are also members of the Project groups.</td>
</tr>
<tr>
<td>Create the Project groups.</td>
<td>Each OpenStack Project will require a corresponding AD group. For example, grp-openstack-demo and grp-openstack-admin.</td>
</tr>
<tr>
<td>Configure the service account.</td>
<td>The service account svc-ldap must be a member of the grp-openstack group.</td>
</tr>
<tr>
<td>Export the LDAPS public key.</td>
<td>Export the public key (not the private key) in the following format: DER-encoded x509 .cer file.</td>
</tr>
</tbody>
</table>
Send the key to the OpenStack administrators. The OpenStack administrators will use this key to encrypt LDAPS communications between OpenStack and Active Directory.

Retrieve the NetBIOS name of your AD DS domain. The OpenStack administrators will use this name for the Keystone domain, allowing consistent domain naming between the environments.

For example, the procedure below shows the PowerShell commands that would be run on the Active Directory Domain Controller:

1. Create the LDAP lookup account. This account is used by Identity Service to query the AD DS LDAP service:

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

2. Set a password for this account, and then enable it. You will be prompted to specify a password that complies with your AD domain's complexity requirements:

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

3. Create a group for OpenStack users, called `grp-openstack`.

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

4. Create the Project groups:

   ```powershell
   PS C:\> NEW-ADGroup -name "grp-openstack-demo" -groupscope Global -path "OU=labUsers,DC=lab,DC=local"
   PS C:\> NEW-ADGroup -name "grp-openstack-admin" -groupscope Global -path "OU=labUsers,DC=lab,DC=local"
   ```

5. Add the `svc-ldap` user to the `grp-openstack` group:

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

6. From an AD Domain Controller, use a Certificates MMC to export your LDAPS certificate's public key (not the private key) as a DER-encoded x509 .cer file. Send this file to the OpenStack administrators.

7. Retrieve the NetBIOS name of your AD DS domain.

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

   Send this value to the OpenStack administrators.
1.7. CONFIGURE THE LDAPS CERTIFICATE

1. Copy the LDAPS public key to the node running OpenStack Identity (keystone), and convert the .cer to .pem. This example uses a certificate file named `addc.lab.local.cer`:

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

2. Install the .pem on your OpenStack controller. For example, in Red Hat Enterprise Linux:

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

3. Convert the .pem to .crt and copy to the certificate directory:

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

1.8. CONFIGURE IDENTITY SERVICE

These steps prepare Identity Service for integration with AD DS.

1.8.1. Enable command line access to keystone v3

To manage Identity Service domains from the command line, you need to enable access to keystone v3.

Perform this procedure from the controller running the keystone service.

1. Create a copy of the existing environment variable file. In a director-based deployment, it will be called `overcloudrc`:

   ```bash
   $ cp overcloudrc overcloudrc-v3
   ```

2. Edit the new `overcloudrc-v3` file:

   - Change `OS_AUTH_URL` from v2.0 to v3. For example:

     ```bash
     ```

   - Add the following entries to the bottom of `overcloudrc-v3`:

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

3. Enable these options for your current command line session by sourcing the file:

   ```bash
   $ source overcloudrc-v3
   ```

1.8.2. Configure the controller
Perform this procedure from the controller running the keystone service. If running a HA environment with multiple controllers, then these steps must be performed on each controller:

1. Configure SELinux:
   ```bash
   # setsebool -P authlogin_nsswitch_use_ldap=on
   
   The output might include messages similar to this. They can be ignored:
   ```
   ```bash
   Full path required for exclude: net:[4026532245].
   ```

2. Create the `domains` directory:
   ```bash
   # mkdir /etc/keystone/domains/
   # chown keystone /etc/keystone/domains/
   ```

3. Configure Identity Service to use multiple back ends:
   ```bash
   # crudini --set /etc/keystone/keystone.conf identity
domain_specific_drivers_enabled true
   # crudini --set /etc/keystone/keystone.conf identity domain_config_dir
   /etc/keystone/domains
   # crudini --set /etc/keystone/keystone.conf assignment driver sql
   
   **NOTE**
   
   You might need to install `crudini` using `yum install crudini`.

   ```bash
   # crudini --set /etc/keystone/keystone.conf identity
domain_specific_drivers_enabled true
   ```

   **NOTE**
   
   If you are using Red Hat OpenStack Platform director, then you will need to be aware that `/etc/keystone/keystone.conf` is managed by Puppet. Consequently, any custom configuration you add might be overwritten whenever you run the `openstack overcloud deploy` process. As a result, you might need to re-add this configuration manually each time. It is expected that a future release of director will include the Puppet parameters that will allow you to re-apply these settings automatically using a post-deployment script.

4. Configure an additional back end:

   In this example, LAB is the NetBIOS name to use as the Identity Service domain.

   a. Create the keystone domain for AD DS integration.

   Use the NetBIOS name value retrieved previously as the domain name. This approach allows you to present a consistent domain name to users during the login process. For example, if the NetBIOS name is LAB:

   ```bash
   # openstack domain create LAB
   ```
NOTE

If this command is not available, check that you have enabled keystone v3 for your command line session by running `source overcloudrc-v3`.

b. Create the configuration file:

To add the AD DS back end, enter the LDAP settings in a new file called `/etc/keystone/domains/keystone.LAB.conf` (where LAB is the NetBIOS name retrieved previously). You will need to edit the sample settings below to suit your AD DS deployment:

```python
[ldap]
url                  = ldaps://addc.lab.local:636
user                  = CN=srv-ldap,OU=labUsers,DC=lab,DC=local
password                 = RedactedComplexPassword
suffix                   = DC=lab,DC=local
user_tree_dn             = OU=labUsers,DC=lab,DC=local
user_objectclass         = person
user_filter                  = ((memberOf=cn=grp-openstack,OU=labUsers,DC=lab,DC=local)(memberOf=cn=grp-openstack-admin,OU=labUsers,DC=lab,DC=local)(memberOf=memberOf=cn=grp-openstack-demo,OU=labUsers,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
group_objectclass        = group
group_tree_dn             = OU=labUsers,DC=lab,DC=local
group_filter              = (CN=grp-openstack*)
group_id_attribute       = cn
group_name_attribute     = name
group_allow_create       = False
group_allow_update       = False
group_allow_delete       = False
use_tls                   = False
tls_cacertfile                  = /etc/ssl/certs/addc.lab.local.crt
query_scope                  = sub
chase_referrals                  = false

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

Explanation of each setting:

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>url</strong></td>
<td>ldaps://addc.lab.local:636</td>
</tr>
<tr>
<td><strong>user</strong></td>
<td>CN=srv-ldap,OU=labUsers,DC=lab,DC=local</td>
</tr>
<tr>
<td><strong>password</strong></td>
<td>RedactedComplexPassword</td>
</tr>
<tr>
<td><strong>suffix</strong></td>
<td>DC=lab,DC=local</td>
</tr>
<tr>
<td><strong>user_tree_dn</strong></td>
<td>OU=labUsers,DC=lab,DC=local</td>
</tr>
<tr>
<td><strong>user_objectclass</strong></td>
<td>person</td>
</tr>
<tr>
<td><strong>user_filter</strong></td>
<td>((memberOf=cn=grp-openstack,OU=labUsers,DC=lab,DC=local)(memberOf=cn=grp-openstack-admin,OU=labUsers,DC=lab,DC=local)(memberOf=memberOf=cn=grp-openstack-demo,OU=labUsers,DC=lab,DC=local))</td>
</tr>
<tr>
<td><strong>user_id_attribute</strong></td>
<td>sAMAccountName</td>
</tr>
<tr>
<td><strong>user_name_attribute</strong></td>
<td>sAMAccountName</td>
</tr>
<tr>
<td><strong>user_mail_attribute</strong></td>
<td>mail</td>
</tr>
<tr>
<td><strong>user_pass_attribute</strong></td>
<td></td>
</tr>
<tr>
<td><strong>user_enabled_attribute</strong></td>
<td>userAccountControl</td>
</tr>
<tr>
<td><strong>user_enabled_mask</strong></td>
<td>2</td>
</tr>
<tr>
<td><strong>user_enabled_default</strong></td>
<td>512</td>
</tr>
<tr>
<td><strong>user_attribute_ignore</strong></td>
<td>password,tenant_id,tenants</td>
</tr>
<tr>
<td><strong>user_allow_create</strong></td>
<td>False</td>
</tr>
<tr>
<td><strong>user_allow_update</strong></td>
<td>False</td>
</tr>
<tr>
<td><strong>user_allow_delete</strong></td>
<td>False</td>
</tr>
<tr>
<td><strong>group_objectclass</strong></td>
<td>group</td>
</tr>
<tr>
<td><strong>group_tree_dn</strong></td>
<td>OU=labUsers,DC=lab,DC=local</td>
</tr>
<tr>
<td><strong>group_filter</strong></td>
<td>(CN=grp-openstack*)</td>
</tr>
<tr>
<td><strong>group_id_attribute</strong></td>
<td>cn</td>
</tr>
<tr>
<td><strong>group_name_attribute</strong></td>
<td>name</td>
</tr>
<tr>
<td><strong>group_allow_create</strong></td>
<td>False</td>
</tr>
<tr>
<td><strong>group_allow_update</strong></td>
<td>False</td>
</tr>
<tr>
<td><strong>group_allow_delete</strong></td>
<td>False</td>
</tr>
<tr>
<td><strong>use_tls</strong></td>
<td>False</td>
</tr>
<tr>
<td><strong>tls_cacertfile</strong></td>
<td>/etc/ssl/certs/addc.lab.local.crt</td>
</tr>
<tr>
<td><strong>query_scope</strong></td>
<td>sub</td>
</tr>
<tr>
<td><strong>chase_referrals</strong></td>
<td>false</td>
</tr>
<tr>
<td><strong>driver</strong></td>
<td>keystone.identity.backends.ldap.Identity</td>
</tr>
<tr>
<td>Setting</td>
<td>Description</td>
</tr>
<tr>
<td>----------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>url</td>
<td>The AD Domain Controller to use for authentication. Uses LDAPS port 636.</td>
</tr>
<tr>
<td>user</td>
<td>The <em>Distinguished Name</em> of an AD account to use for LDAP queries. For example, you can locate the <em>Distinguished Name</em> value of the svc-ldap account in AD using `Get-ADUser svc-ldap</td>
</tr>
<tr>
<td>password</td>
<td>The plaintext password of the AD account used above.</td>
</tr>
<tr>
<td>suffix</td>
<td>The <em>Distinguished Name</em> of your AD domain. You can locate this value using `Get-ADDomain</td>
</tr>
<tr>
<td>user_tree_dn</td>
<td>The <em>Organizational Unit</em> (OU) that contains the OpenStack accounts.</td>
</tr>
<tr>
<td>user_objectclass</td>
<td>Defines the type of LDAP user. For AD, use the <em>person</em> type.</td>
</tr>
<tr>
<td>user_filter</td>
<td>Filters the users presented to Identity Service. As a result, only members of the <code>grp-openstack</code> group can have permissions defined in Identity Service. This value requires the full <em>Distinguished Name</em> of the group: `Get-ADGroup grp-openstack</td>
</tr>
<tr>
<td>user_id_attribute</td>
<td>Maps the AD value to use for user IDs.</td>
</tr>
<tr>
<td>user_name_attribute</td>
<td>Maps the AD value to use for <em>names</em>.</td>
</tr>
<tr>
<td>user_mail_attribute</td>
<td>Maps the AD value to use for user email addresses.</td>
</tr>
<tr>
<td>user_pass_attribute</td>
<td>Leave this value blank.</td>
</tr>
<tr>
<td>user_enabled_attribute</td>
<td>The AD setting that validates whether the account is enabled.</td>
</tr>
<tr>
<td>user_enabled_mask</td>
<td>Defines the value to check to determine whether an account is enabled. Used when booleans are not returned.</td>
</tr>
<tr>
<td>user_enabled_default</td>
<td>The AD value that indicates that an account is enabled.</td>
</tr>
<tr>
<td>Setting</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>user_attribute_ignore</strong></td>
<td>Defines user attributes that Identity Service should disregard.</td>
</tr>
<tr>
<td><strong>user_allow_create</strong></td>
<td>Set this value to False, as keystone only requires read-only access.</td>
</tr>
<tr>
<td><strong>user_allow_update</strong></td>
<td>Set this value to False, as keystone only requires read-only access.</td>
</tr>
<tr>
<td><strong>user_allow_delete</strong></td>
<td>Set this value to False, as keystone only requires read-only access.</td>
</tr>
<tr>
<td><strong>group_objectclass</strong></td>
<td>Maps the AD value to use for groups.</td>
</tr>
<tr>
<td><strong>group_tree_dn</strong></td>
<td>The Organizational Unit (OU) that contains the user groups.</td>
</tr>
<tr>
<td><strong>group_filter</strong></td>
<td>Filters the groups presented to Identity Service.</td>
</tr>
<tr>
<td><strong>group_id_attribute</strong></td>
<td>Maps the AD value to use for group IDs.</td>
</tr>
<tr>
<td><strong>group_name_attribute</strong></td>
<td>Maps the AD value to use for group names.</td>
</tr>
<tr>
<td><strong>group_allow_create</strong></td>
<td>Set this value to False, as keystone only requires read-only access.</td>
</tr>
<tr>
<td><strong>group_allow_update</strong></td>
<td>Set this value to False, as keystone only requires read-only access.</td>
</tr>
<tr>
<td><strong>group_allow_delete</strong></td>
<td>Set this value to False, as keystone only requires read-only access.</td>
</tr>
<tr>
<td><strong>use_tls</strong></td>
<td>Defines whether TLS is to be used. This needs to be disabled if you are encrypting with LDAPS rather than STARTTLS.</td>
</tr>
<tr>
<td><strong>tls_cacertfile</strong></td>
<td>Specifies the path to the .crt certificate file.</td>
</tr>
<tr>
<td><strong>query_scope</strong></td>
<td>Configures Identity Service to also search within nested child OUs, when locating users that are members of the grp-openstack group.</td>
</tr>
<tr>
<td><strong>chase_referrals</strong></td>
<td>Set to false, this setting prevents python-ldap from chasing all referrals with anonymous access.</td>
</tr>
</tbody>
</table>

6. Change ownership of the configuration file to the keystone user:
# chown keystone /etc/keystone/domains/keystone.LAB.conf

7. Configure the dashboard to use the LAB keystone domain at the login page. Add these lines to /etc/openstack-dashboard/local_settings:

```
IMPORTANT
Multidomain Dashboard configuration is not supported in this version of Red Hat OpenStack Platform. As a result, this guide only describes Dashboard configuration for a single domain.

OPENSTACK_API_VERSIONS = {
    "identity": 3
}
OPENSTACK_KEYSTONE_DEFAULT_DOMAIN = 'LAB'
```

```
NOTE
If you are using Red Hat OpenStack Platform director, then you will need to be aware that /etc/openstack-dashboard/local_settings is managed by Puppet. Consequently, any custom configuration you add might be overwritten whenever you run the openstack overcloud deploy process. As a result, you might need to re-add this configuration manually each time. It is expected that a future release of director will include the Puppet parameters that will allow you to re-apply these settings automatically using a post-deployment script.
```

8. Restart the httpd service to apply the changes:

```
# systemctl restart httpd.service
```

9. Grant the admin user access to the domain:

```
NOTE
This does not grant the OpenStack admin account any permissions on the actual AD DS domain. In this case, the term domain refers to OpenStack’s usage of the keystone domain.
```

a. Get the ID of the LAB domain:

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

b. Get the ID value of the admin user:
c. Get the ID value of the admin role:

```
# openstack role list
+----------------------------------+---------------+
| ID                               | Name          |
+----------------------------------+---------------+
| 544d48aaffde48f1b3c31a52c35f01f9 | SwiftOperator |
| 6d065d783bf0436e882c55c62457d33d | ResellerAdmin |
| 785c70b150ee4c778fe4de088070b4cf | admin         |
| 9fe2ff9ee4384b1894a90878d3e92bab | _member_      |
```

d. Use the returned domain and admin IDs to construct the command that adds the admin user to the admin role of the keystone LAB domain:

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

10. View the list of users in the AD DS domain by adding the NetBIOS name to the command:

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

11. View the service accounts in the local Identity Service database:

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

1.8.3. Configure Compute to use keystone v3

Compute uses keystone v2.0 by default, and so needs to be configured to use keystone v3 in order to use multi-domain capabilities.

1. On each Compute node, and the controller, adjust the keystone_authtoken value:

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

2. Restart these services on the controller to apply the changes:

```
# 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. Restart this service on each Compute node to apply the changes:
1.8.4. Configure Block Storage to use keystone v3

You must also configure Block Storage (cinder) to authenticate to keystone v3.

1. In `/etc/cinder/cinder.conf`:

   ```
   [keystone_authtoken]
   auth_uri = https://controllerIP:5000/v3
   auth_version = v3
   
   - auth_uri - replace controllerIP with the IP address of the controller. If your deployment has more than one controller, you should use the keystone endpoint VIP instead of the controller IP.
   ```

1.8.5. Allow Active Directory group members to access Projects

To allow authenticated users access to OpenStack resources, the recommended method is to authorize certain Active Directory groups to grant access to Projects. This saves the OpenStack administrators from having to allocate each user to a role in a Project. Instead, the Active Directory groups are granted roles in Projects. As a result, Active Directory users that are members of these Active Directory groups will be able to access pre-determined Projects.

**NOTE**

If you would prefer to manually manage the authorization of individual Active Directory users, see the following section: *Allow individual Active Directory users to access Projects*

This section presumes that the Active Directory administrator has already completed these steps:

- Create a group named `grp-openstack-admin` in Active Directory.
- Create a group named `grp-openstack-demo` in Active Directory.
- Add your Active Directory users to one of the above groups, as needed.
- Add your Active Directory users to the `grp-openstack` group.

These steps assign a role to an AD group. Group members will then have permission to access OpenStack resources.

1. Retrieve a list of AD groups:

   ```
   # openstack group list --domain LAB
   +-------------------------------+-----------------------------+-----------------------------+
   | ID                           | Name                       | grp-openstack-admin         |
   +-------------------------------+-----------------------------+-----------------------------+
   | 185277be62ae17e498a69f98a59b66934fb1d6b7f745f14f5f68953a665b8851 | grp-openstack-demo         | grp-openstack-admin         |
   | a8d17f19f464c4548c18b97e4aa331820f9d3be52654aa8094e698a9182cbb88 | grp-openstack-admin         | grp-openstack-admin         |
   ```
2. Retrieve a list of roles:

```
# openstack role list
+----------------------------------+---------------+
| ID                               | Name          |
+----------------------------------+---------------+
| 0969957bce5e4f678ca6ceef00e1abf8a | ResellerAdmin |
| 1fcb3c9b50aa46ee8196aaaec2b76b7  | admin         |
| 9fe2ff9e4384b1894a90878d3e92bab  | _member_      |
| d3570730eb4b4780a7fed97e9a197e1b | SwiftOperator |
+----------------------------------+---------------+
```

3. Grant the Active Directory groups access to Projects by adding them to one or more of these roles. For example, if you want users in the `grp-openstack-demo` group to be general users of the `demo` project, you must add the group to the `member` role:

```
# openstack role add --project demo --group d971bb3bd5e64a454cb60cc7af4077378d61b5f81321809f8323216938cae8  _member_
```

As a result, members of `grp-openstack-demo` are able to log in to the dashboard by entering their AD DS username and password.

**NOTE**

If users receive the error "Error: Unable to retrieve container list.", and expect to be able to manage containers, then they must be added to the `SwiftOperator` role.

1.8.6. Allow Active Directory users to access Projects

AD DS users that are members of the `grp-openstack` AD group can be granted permission to log in to a Project in the dashboard:

1. Retrieve a list of AD users:

```
# openstack user list --domain LAB
+------------------------------------------------------------------+
| ID                                                               |
| Name                                                             |
+------------------------------------------------------------------+
| 1f24ec1f11ae9b0520079c29f70afa060d22e2ce92b2e97a784c841ac48091e | user1     |
| 12c062faddc5f8b065434d0f9f6f1c0e3eb9259537c93b41224588686e9a38bf1 | user2     |
| afa48031eb54c3e44e4cb0353f5b61208403ff70f63c22873d181fda2e73c | user3     |
| e47fc21dcf0d9716d2663766023e2d8dc15a69b01453854a898cabb2396826e | user4     |
```
2. Retrieve a list of roles:

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

3. Grant users access to Projects by adding them to one or more of these roles. For example, if you want `user1` to be a general user of the `demo` project, you add them to the `member` role:

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

Or, if you want `user1` to be an administrative user of the `demo` project, you add them to the `admin` role:

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

As a result, `user1` is able to log in to the dashboard by entering their AD DS username and password.

**NOTE**

If users receive the error "Error: Unable to retrieve container list.", and expect to be able to manage containers, then they must be added to the `SwiftOperator` role.

### 1.9. GRANT ACCESS TO THE DOMAIN TAB

To allow the `admin` user to see the `Domain` tab, you will need to assign it the `admin` role in the `default` domain:

1. Find the `admin` user's UUID:

   ```
   $ openstack user list | grep admin
   | a6a8adb6356f4a879f079485dad1321b | admin |
   ```

2. Add the `admin` role in the `default` domain to the `admin` user:

   ```
   $ openstack role add --domain default --user a6a8adb6356f4a879f079485dad1321b admin
   ```

   As a result, the `admin` user can now see the `Domain` tab.

### 1.10. CREATING A NEW PROJECT
After you have completed these integration steps, when you create a new project you will need to decide whether to create it in the Default domain, or in the keystone domain you’ve just created. This decision can be reached by considering your workflow, and how you administer user accounts. The Default domain can be be thought of as an internal domain, used to manage service accounts and the admin project. For separation purposes, you might want to keep your AD-backed users in a separate keystone domain.

1.11. CHANGES TO THE COMMAND LINE

For certain commands, you might need to specify the applicable domain. For example, appending --domain LAB in this command returns users in the LAB domain (that are members of the grp-openstack group):

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

Appending --domain Default returns the built-in keystone accounts:

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

1.12. TEST AD DS INTEGRATION

This procedure validates AD DS integration by testing user access to dashboard features:

1. Create a test user in AD, and add the user to the grp-openstack AD DS group.
2. Add the user to the _member_ role of the demo tenant.
3. Log in to the dashboard using the credentials of the AD test user.
4. Click on each of the tabs to confirm that they are presented successfully without error messages.
5. Use the dashboard to build a test instance.

**NOTE**

If you experience issues with these steps, perform steps 3-5 with the built-in admin account. If successful, this demonstrates that OpenStack is still working as expected, and that an issue exists somewhere within the AD Identity integration settings. See Section 1.15, "Troubleshooting".

1.13. CONFIGURE FOR HIGH AVAILABILITY

With keystone v3 enabled, you can make this configuration highly available by listing multiple AD Domain Controllers in the configuration file for that domain.

1. Add a second server to the url entry. For example, updating the url setting in the keystone.LAB.conf file will have Identity Service send all query traffic to the first Domain Controller in the list, addc.lab.local:

```
url = ldaps://addc.lab.local,ldaps://addc2.lab.local
```
If a query to `adcc.lab.local` fails due to it being unavailable, Identity Service will attempt to query the next server in the list: `adcc2.lab.local`. Note that this configuration does not perform queries in a round-robin fashion, so cannot be considered a load-balancing solution.

2. Set the network timeout in `/etc/openldap/ldap.conf`:

```ini
NETWORK_TIMEOUT 2
```

In addition, if you have firewalls configured between the controller and the domain controllers, then you should not configure the domain controllers to silently drop packets from the controller. This will allow `python-keystoneclient` to properly detect outages and redirect the request to the next domain controller in the list.

**NOTE**

There might be connection delays while queries are being redirected to the second LDAP server in the list. This is because the connection to the first server must first time out before the second is attempted.

### 1.14. CREATE A RC FILE FOR A NON-ADMIN USER

You might need to create a RC file for a non-admin user. For example:

```bash
$ cat overcloudrc-v3-user1
# Clear any old environment that may conflict.
for key in $( set | awk '{FS="="} /^OS_/ {print $1}' ); do unset $key ; done
export OS_USERNAME=user1
export NOVA_VERSION=1.1
export OS_PROJECT_NAME=demo
export OS_PASSWORD=RedactedComplexPassword
export OS_NO_CACHE=True
export COMPUTE_API_VERSION=1.1
export no_proxy=,10.0.0.5,192.168.2.11
export OS_CLOUDNAME=overcloud
export OS_AUTH_URL=https://10.0.0.5:5000/v3
export OS_AUTH_TYPE=password
export PYTHONWARNINGS="ignore:Certificate has no, ignore:A true SSLContext object is not available"
export OS_IDENTITY_API_VERSION=3
export OS_PROJECT_DOMAIN_NAME=Default
export OS_USER_DOMAIN_NAME=LAB
```

### 1.15. TROUBLESHOOTING

#### 1.15.1. Test LDAP connections

Use `ldapsearch` to remotely perform test queries against the Active Directory Domain Controller. A successful result here indicates that network connectivity is working, and the AD DS services are up. In this example, a test query is performed against the server `adcc.lab.local` on port 636:

```
# ldapsearch -Z -x -H ldaps://adcc.lab.local:636 -D "svc-ldap@lab.local" -W -b "OU=labUsers,DC=lab,DC=local" -s sub ";(cn=*)" cn
```
NOTE

`ldapsearch` is a part of the `openldap-clients` package. You can install this using `# yum install openldap-clients`

1.15.2. Test the Certificate Trust Configuration

If you receive the error `Peer's Certificate issuer is not recognized.` while testing with `ldapsearch`, confirm that your `TLS_CACERTDIR` path is correctly set. For example:

- `/etc/openldap/ldap.conf`

  ```
  TLS_CACERTDIR /etc/openldap/certs
  ```

NOTE

As a temporary workaround, you may want to consider disabling certificate validation.

This setting must not be permanently configured:

- `/etc/openldap/ldap.conf`

  ```
  TLS_REQCERT allow
  ```

  If the `ldapsearch` query works after setting this value, you might need to review whether your certificate trusts are correctly configured.

1.15.3. Test port access

Use `nc` to check that LDAPS port 636 is remotely accessible. In this example, a probe is performed against the server `addc.lab.local`. Press `ctl-c` to exit the prompt.

```
# nc -v addc.lab.local 636
^C
```

Failure to establish a connection could indicate a firewall configuration issue.
CHAPTER 2. IDENTITY MANAGEMENT INTEGRATION

This chapter describes how to integrate Identity Service (keystone) with Red Hat Identity Management.

In this use case, Identity Service authenticates certain Red Hat Identity Management (IdM) users, while retaining authorization settings and critical service accounts in the Identity Service database. As a result, Identity Service has read-only access to IdM for user account authentication, while retaining management over the privileges assigned to authenticated accounts.

2.1. KEY TERMS

- **Authentication** - The process of using a password to verify that the user is who they claim to be.
- **Authorization** - Validating that authenticated users have proper permissions to the systems they're attempting to access.
- **Domain** - Refers to the additional back ends configured in Identity Service. For example, Identity Service can be configured to authenticate users from external IdM environments. The resulting collection of users can be thought of as a domain.

2.2. ASSUMPTIONS

This example deployment makes the following assumptions:

- Red Hat Identity Management is configured and operational.
- Red Hat OpenStack Platform is configured and operational.
- DNS name resolution is fully functional and all hosts are registered appropriately.

**IMPORTANT**

Multidomain Dashboard configuration is not supported in this version of Red Hat OpenStack Platform. As a result, this guide only describes Dashboard configuration for a single domain.

2.3. IMPACT STATEMENT

These steps allow IdM users to authenticate to OpenStack and access resources. OpenStack service accounts (such as keystone and glance), and authorization management (permissions and roles) will remain in the Identity Service database. Permissions and roles are assigned to the IdM accounts using Identity Service management tools.

2.3.1. High Availability options

This configuration creates a dependency on the availability of a single IdM server: Project users will be affected if Identity Service is unable to authenticate to the IdM Server. There are a number of options available to manage this risk, for example: you might configure keystone to query a DNS alias or a load balancing appliance, rather than an individual IdM server. You can also configure keystone to query a different IdM server, should one become unavailable. See Section 2.11, “Configure for high availability” for more information.
2.4. OUTAGE REQUIREMENTS

- The Identity Service will need to be restarted in order to add the IdM back end.
- The Compute services on all nodes will need to be restarted in order to switch over to keystone v3.
- Users will be unable to access the dashboard until their accounts have been created in IdM. To reduce downtime, consider pre-staging the IdM accounts well in advance of this change.

2.5. FIREWALL CONFIGURATION

If firewalls are filtering traffic between IdM and OpenStack, you will need to allow access through the following port:

<table>
<thead>
<tr>
<th>Source</th>
<th>Destination</th>
<th>Type</th>
<th>Port</th>
</tr>
</thead>
<tbody>
<tr>
<td>OpenStack Controller Node</td>
<td>Red Hat Identity Management</td>
<td>LDAPS</td>
<td>TCP 636</td>
</tr>
</tbody>
</table>

2.6. CONFIGURE THE IDM SERVER

Run these commands on the IdM server:

1. Create the LDAP lookup account. This account is used by Identity Service to query the IdM LDAP service:

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

**NOTE**

Review the password expiration settings of this account, once created.

2. Create a group for OpenStack users, called `grp-openstack`. Only members of this group can have permissions assigned in OpenStack Identity.

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

3. Set the `svc-ldap` account password, and add it to the `grp-openstack` group:

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

2.7. CONFIGURE THE LDAPS CERTIFICATE
1. In your IdM environment, locate the LDAPS certificate. This file can be located using
/etc/openldap/ldap.conf:

   TLS_CACERT /etc/ipa/ca.crt

2. Copy the file to the node running OpenStack Identity (keystone). For example, this command uses
   `scp` to copy ca.crt to the controller node named `node.lab.local`:

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

3. On the controller node, convert the `.crt` to `.pem`:

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

4. Install the `.pem` on your OpenStack controller. For example, in Red Hat Enterprise Linux:

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

   `# update-ca-trust`

5. Copy the `.crt` to the certificate directory:

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

## 2.8. CONFIGURE IDENTITY SERVICE

These steps prepare Identity Service for integration with IdM.

### 2.8.1. Enable command line access to keystone v3

To manage Identity Service domains from the command line, you need to enable access to keystone v3.

Perform this procedure from the controller running the keystone service.

1. Create a copy of the existing environment variable file. In a director-based deployment, it will be
called `overcloudrc`:

   `$ cp overcloudrc overcloudrc-v3`

2. Edit the new `overcloudrc-v3` file:

   - Change `OS_AUTH_URL` from v2.0 to v3. For example:


   - Add the following entries to the bottom of `overcloudrc-v3`:

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

3. Enable these options for your current command line session by sourcing the file:
$ source overcloudrc-v3

2.8.2. Configure the controller

Perform this procedure from the controller running the keystone service:

1. Configure SELinux:

   # setsebool -P authlogin_nsswitch_use_ldap=on

2. Create the domains directory:

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

3. Configure Identity Service to use multiple back ends:

   NOTE
   You might need to install crudini using yum install crudini.

   # crudini --set /etc/keystone/keystone.conf identity
   domain_specific_drivers_enabled true
   # crudini --set /etc/keystone/keystone.conf identity domain_config_dir
   /etc/keystone/domains
   # crudini --set /etc/keystone/keystone.conf assignment driver sql

   NOTE
   If you are using Red Hat OpenStack Platform director, then you will need to be aware
   that /etc/keystone/keystone.conf is managed by Puppet. Consequently, any custom
   configuration you add might be overwritten whenever you run the openstack
   overcloud deploy process. As a result, you might need to re-add this configuration
   manually each time. It is expected that a future release of director will include the
   Puppet parameters that will allow you to re-apply these settings automatically using a
   post-deployment script.

4. Configure an additional back end:

   a. Create the keystone domain for IdM integration.

      + Select a name to use for your new keystone domain, and create the domain. For example, this
        command creates a keystone domain named LAB:

        # openstack domain create LAB

       NOTE
       If this command is not available, check that you have enabled keystone v3 for your
       command line session.
b. Create the configuration file:

To add the IdM back end, enter the LDAP settings in a new file called /etc/keystone/domains/keystone.LAB.conf (where LAB is the domain name created previously). You will need to edit the sample settings below to suit your IdM deployment:

```
[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
```

Explanation of each setting:

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>url</td>
<td>The IdM server to use for authentication. Uses LDAPS port 636.</td>
</tr>
<tr>
<td>user</td>
<td>The account in IdM to use for LDAP queries.</td>
</tr>
<tr>
<td>password</td>
<td>The plaintext password of the IdM account used above.</td>
</tr>
<tr>
<td>user_filter</td>
<td>Filters the users presented to Identity Service. As a result, only members of the grp-openstack group can have permissions defined in Identity Service.</td>
</tr>
<tr>
<td>user_tree_dn</td>
<td>The path to the OpenStack accounts in IdM.</td>
</tr>
<tr>
<td>user_objectclass</td>
<td>Defines the type of LDAP user. For IdM, use the inetUser type.</td>
</tr>
<tr>
<td>user_id_attribute</td>
<td>Maps the IdM value to use for user IDs.</td>
</tr>
<tr>
<td>user_name_attribute</td>
<td>Maps the IdM value to use for names.</td>
</tr>
</tbody>
</table>
### Setting | Description
--- | ---
**user_mail_attribute** | Maps the IdM value to use for user email addresses.  
**user_pass_attribute** | Leave this value blank.  
**user_allow_create** | Set this value to `False`, as keystone only requires read-only access.  
**user_allow_update** | Set this value to `False`, as keystone only requires read-only access.  
**user_allow_delete** | Set this value to `False`, as keystone only requires read-only access.  

5. Change ownership of the config file to the keystone user:

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

6. Grant the admin user access to the domain:

**NOTE**

This does not grant the OpenStack admin account any permissions in IdM. In this case, the term domain refers to OpenStack’s usage of the keystone domain.

a. Get the **ID** of the **LAB** domain:

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

b. Get the **ID** value of the **admin** user:

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

c. Get the **ID** value of the **admin role**:

```  
# openstack role list  
+---------------------+-------------------+  
| ID                  | Name              |  
+---------------------+-------------------+  
| 544d48aaffde48f1b3c31a52c35f01f9 | SwiftOperator    |  
```
d. Use the returned domain and admin IDs to construct the command that adds the `admin` user to the `admin` role of the keystone LAB domain:

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

7. Configure the dashboard to use the LAB keystone domain at the login page. Add these lines to `/etc/openstack-dashboard/local_settings`:

```python
IMPORTANT

Multidomain Dashboard configuration is not supported in this version of Red Hat OpenStack Platform. As a result, this guide only describes Dashboard configuration for a single domain.

```python
OPENSTACK_API_VERSIONS = {
    "identity": 3
}
```

```python
OPENSTACK_KEYSTONE_DEFAULT_DOMAIN = 'LAB'
```

```python
# Systemctl restart httpd.service
```

8. Restart the `httpd` service to apply the changes:

```python
9. View the list of users in the IdM domain by adding the keystone domain name to the command:

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

10. View the service accounts in the local keystone database:

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

2.8.3. Configure Compute to use keystone v3

Compute uses `keystone v2.0` by default, and so needs to be configured to use `keystone v3` in order to use multi-domain capabilities.
1. On each Compute node, and the controller, adjust the `keystone_authtoken` value:

   ```
   # crudini --set /etc/nova/nova.conf keystroke_authtoken auth_version v3
   ```

2. Restart these services on the controller to apply the changes:

   ```
   # 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. Restart this service on each Compute node to apply the changes:

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

2.8.4. Configure Block Storage to use keystone v3

You must also configure Block Storage (cinder) to authenticate to keystone v3.

   1. In `/etc/cinder/cinder.conf`:

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

   - `auth_uri` - replace `controllerIP` with the IP address of the controller. If your deployment has more than one controller, you should use the keystone endpoint VIP instead of the controller IP.

2.8.5. Allow IdM users to access Projects

IdM users that are members of the `grp-openstack` IdM group can be granted permission to log in to a project in the dashboard:

1. Retrieve a list of IdM users:

   ```
   # openstack user list --domain LAB
   +------------------------------------------------------------------+-
   | ID     | Name         |
   +------------------------------------------------------------------+-
   | 1f24ec1f11ae90520079c29f70afa060d22e2ce92b2e7a7784c841ac418091e | user1
   | 12c062fd1d5f8b06543d9ff6f6f6ce03eb9259537c93b411224588686e9a38bf1 | user2
   | 4af4831eb543e4e4cb03535f5b612084033f70f63c22873d418091e | user3
   | e47fc21df0d9716d2663766023e2d8dc15a69b01453854a898cabb2396826e | user4
   +------------------------------------------------------------------+-
   ```
2. Retrieve a list of roles:

```
# openstack role list
+----------------------------------+---------------+
| ID                               | Name          |
+----------------------------------+---------------+
| 544d48aaffde48f1b3c31a52c35f01f9 | SwiftOperator |
| 6d005d78b50436e882c55c62457d33d | ResellerAdmin |
| 785c70b150ee4c778fe4de08807b4cf | admin         |
| 9fe2ff9ee4384b1894a90878d3e92bab | _member_      |
+----------------------------------+---------------+
```

3. Grant users access to Projects by adding them to one or more of these roles. For example, if you want *user1* to be a general user of the *demo* project, you add them to the *_member_* role:

```
# openstack role add --project demo --user 1f24ec1f11ae90520079c29f70afa06022e2ce92b2eba7784c84ac418091e _member_
```

Or, if you want *user1* to be an administrative user of the *demo* project, you add them to the *admin* role:

```
# openstack role add --project demo --user 1f24ec1f11ae90520079c29f70afa06022e2ce92b2eba7784c84ac418091e admin
```

As a result, *user1* is able to log in to the dashboard by entering their IdM username and password.

**NOTE**

If users receive the error "Error: Unable to retrieve container list.", and expect to be able to manage containers, then they must be added to the *SwiftOperator* role.

### 2.9. GRANT ACCESS TO THE DOMAIN TAB

To allow the *admin* user to see the *Domain* tab, you will need to assign it the *admin* role in the *default* domain:

1. Find the *admin* user's UUID:

```
$ openstack user list | grep admin
| a6a8adb6356f4a879f079485dad1321b | admin       |
```

2. Add the *admin* role in the *default* domain to the *admin* user:

```
$ openstack role add --domain default --user a6a8adb6356f4a879f079485dad1321b admin
```

As a result, the *admin* user can now see the *Domain* tab.

### 2.10. CREATING A NEW PROJECT

After you have completed these integration steps, when you create a new project you will need to decide whether to create it in the *Default* domain, or in the keystone domain you've just created. This decision can be reached by considering your workflow, and how you administer user accounts. The
Default domain can be thought of as an internal domain, used for service accounts and the admin project, so it might make sense for your AD-backed users to be placed within a different keystone domain; this does not strictly need to be the same keystone domain as the IdM users are in, and for separation purposes, there might be multiple keystone domains.

2.10.1. Changes to the command line

For certain commands, you might need to specify the applicable domain. For example, appending --domain LAB in this command returns users in the LAB domain (that are members of the grp-openstack group):

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

Appending --domain Default returns the built-in keystone accounts:

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

2.10.2. Test IdM integration

This procedure validates IdM integration by testing user access to dashboard features:

1. Create a test user in IdM, and add the user to the _member_ role of the demo tenant.

2. Add the user to the _member_ role of the demo tenant.

3. Log in to the dashboard using the credentials of the IdM test user.

4. Click on each of the tabs to confirm that they are presented successfully without error messages.

5. Use the dashboard to build a test instance.

   **NOTE**

   If you experience issues with these steps, perform steps 3-5 with the built-in admin account. If successful, this demonstrates that OpenStack is still working as expected, and that an issue exists somewhere within the IdM Identity integration settings. See Section 2.13, “Troubleshooting”.

2.11. CONFIGURE FOR HIGH AVAILABILITY

With keystone v3 enabled, you can make this configuration highly available by listing multiple IdM servers in the configuration file for that domain.

1. Add a second server to the url entry. For example, updating the url setting in the keystone.LAB.conf file will have Identity Service send all query traffic to the first IdM server in the list, idm.lab.local:

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

   If a query to idm.lab.local fails due to it being unavailable, Identity Service will attempt to query the next server in the list: idm2.lab.local. Note that this configuration does not perform queries in a round-robin fashion, so cannot be considered a load-balancing solution.
2. Set the network timeout in `/etc/openldap/ldap.conf`:

```
NETWORK_TIMEOUT 2
```

In addition, if you have firewalls configured between the controller and the IdM servers, then you should not configure the IdM servers to silently drop packets from the controller. This will allow `python-keystoneclient` to properly detect outages and redirect the request to the next IdM server in the list.

**NOTE**

There might be connection delays while queries are being redirected to the second IdM server in the list. This is because the connection to the first server must first time out before the second is attempted.

### 2.12. CREATE A RC FILE FOR A NON-ADMIN USER

You might need to create a RC file for a non-admin user. For example:

```
$ cat overcloudrc-v3-user1
# Clear any old environment that may conflict.
for key in $( set | awk '{FS="="}  /^OS_/ {print $1}' ); do unset $key ; done
export OS_USERNAME=user1
export NOVA_VERSION=1.1
export OS_PROJECT_NAME=demo
export OS_PASSWORD=RedactedComplexPassword
export OS_NO_CACHE=True
export COMPUTE_API_VERSION=1.1
export no_proxy=,10.0.0.5,192.168.2.11
export OS_CLOUDNAME=overcloud
export OS_AUTH_URL=https://10.0.0.5:5000/v3
export OS_AUTH_TYPE=password
export PYTHONWARNINGS="ignore:Certificate has no, ignore:A true SSLContext object is not available"
export OS_IDENTITY_API_VERSION=3
export OS_PROJECT_DOMAIN_NAME=Default
export OS_USER_DOMAIN_NAME=LAB
```

### 2.13. TROUBLESHOOTING

#### 2.13.1. Test LDAP connections

Use `ldapsearch` to remotely perform test queries against the IdM server. A successful result here indicates that network connectivity is working, and the IdM services are up. In this example, a test query is performed against the server `idm.lab.local` on port 636:

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

ldapsearch is a part of the openldap-clients package. You can install this using

```bash
# yum install openldap-clients
```

2.13.2. Test port access

Use `nc` to check that the LDAPS port (636) is remotely accessible. In this example, a probe is performed against the server `idm.lab.local`. Press ctrl-c to exit the prompt.

```bash
# nc -v idm.lab.local 636
^C
```

Failure to establish a connection could indicate a firewall configuration issue.
CHAPTER 3. GENERIC LDAP INTEGRATION

This chapter describes how to integrate Identity Service (keystone) with a generic LDAP environment.

In this use case, Identity Service authenticates certain LDAP users, while retaining authorization settings and critical service accounts in the Identity Service database. As a result, Identity Service has read-only access to LDAP for user account authentication, while retaining management over the privileges assigned to authenticated accounts.

3.1. KEY TERMS

- **Authentication** - The process of using a password to verify that the user is who they claim to be.

- **Authorization** - Validating that authenticated users have proper permissions to the systems they’re attempting to access.

- **Domain** - Refers to the additional back ends configured in Identity Service. For example, Identity Service can be configured to authenticate users from external LDAP environments. The resulting collection of users can be thought of as a *domain*.

3.2. ASSUMPTIONS

This example deployment makes the following assumptions:

- Your LDAP server is configured and operational.

- Red Hat OpenStack Platform is configured and operational.

- DNS name resolution is fully functional and all hosts are registered appropriately.

**IMPORTANT**

Multidomain Dashboard configuration is not supported in this version of Red Hat OpenStack Platform. As a result, this guide only describes Dashboard configuration for a single domain.

3.3. IMPACT STATEMENT

These steps allow LDAP users to authenticate to OpenStack and access resources. OpenStack service accounts (such as keystone and glance) and authorization management (permissions and roles) will remain in the Identity Service database. Permissions and roles are assigned to the LDAP accounts using Identity Service management tools.

3.3.1. High Availability options

This configuration creates a dependency on the availability of a single LDAP server: Project users will be affected if Identity Service is unable to authenticate to the LDAP Server. There are a number of options available to manage this risk, for example: you might configure keystone to query a DNS alias or a load balancing appliance, rather than an individual LDAP server. You can also configure keystone to query a different LDAP server, should one become unavailable. See Section 3.11, “Configure for high availability” for more information.
3.4. OUTAGE REQUIREMENTS

- The Identity Service will need to be restarted in order to add the LDAP back end.
- The Compute services on all nodes will need to be restarted in order to switch over to keystone v3.
- Users will be unable to access the dashboard until their accounts have been created in LDAP. To reduce downtime, consider pre-staging the LDAP accounts well in advance of this change.

3.5. FIREWALL CONFIGURATION

If firewalls are filtering traffic between LDAP and OpenStack, you will need to allow access through the following port:

<table>
<thead>
<tr>
<th>Source</th>
<th>Destination</th>
<th>Type</th>
<th>Port</th>
</tr>
</thead>
<tbody>
<tr>
<td>OpenStack Controller Node</td>
<td>LDAP server</td>
<td>LDAPS</td>
<td>TCP 636</td>
</tr>
</tbody>
</table>

3.6. CONFIGURE THE LDAP SERVER

Perform the following steps on your LDAP server to prepare for Identity Service integration:

1. Create the LDAP lookup account.

   This account is used by Identity Service to query the LDAP service. In addition to your standard password policy requirements, it will need the following attributes for this example deployment:
   - First name: OpenStack
   - Last name: LDAP
   - User name: svc-ldap

   **NOTE**
   - Review the password expiration settings of this account, once created.

2. Create a LDAP group for OpenStack users, called `grp-openstack`. Only members of this group can have permissions assigned in OpenStack Identity.

3. Set the `svc-ldap` account password, and add it to the `grp-openstack` group.

3.7. CONFIGURE THE LDAPS CERTIFICATE

1. In your LDAP environment, locate the LDAPS certificate. This file can be located using `/etc/openldap/ldap.conf`:

   ```
   TLS_CACERT /etc/ipa/ca.crt
   ```
2. Copy the file to the node running OpenStack Identity (keystone). For example, this command uses `scp` to copy `ca.crt` to the controller node named `node.lab.local`:

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

3. On the controller node, convert the .crt to .pem:

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

4. Install the .pem on your OpenStack controller. For example, in Red Hat Enterprise Linux:

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

5. Copy the .crt to the certificate directory:

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

### 3.8. CONFIGURE IDENTITY SERVICE

These steps prepare Identity Service for integration with LDAP.

#### 3.8.1. Enable command line access to keystone v3

To manage Identity Service domains from the command line, you need to enable access to keystone v3.

Perform this procedure from the controller running the keystone service.

1. Create a copy of the existing environment variable file. In a director-based deployment, it will be called `overcloudrc`:

```bash
$ cp overcloudrc overcloudrc-v3
```

2. Edit the new `overcloudrc-v3` file:

   - Change `OS_AUTH_URL` from `v2.0` to `v3`. For example:

     ```bash
     ```

   - Add the following entries to the bottom of `overcloudrc-v3`:

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

3. Enable these options for your current command line session by sourcing the file:

```bash
$ source overcloudrc-v3
```

#### 3.8.2. Configure the controller
Perform this procedure from the controller running the keystone service:

1. Configure SELinux:

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

2. Create the `domains` directory:

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

3. Configure Identity Service to use multiple back ends:

   **NOTE**

   You might need to install `crudini` using `yum install crudini`.

   ```bash
   # crudini --set /etc/keystone/keystone.conf identity domain_specific_drivers_enabled true
   # crudini --set /etc/keystone/keystone.conf identity domain_config_dir /etc/keystone/domains
   # crudini --set /etc/keystone/keystone.conf assignment driver sql
   ```

   **NOTE**

   If you are using Red Hat OpenStack Platform director, then you will need to be aware that `/etc/keystone/keystone.conf` is managed by Puppet. Consequently, any custom configuration you add might be overwritten whenever you run the `openstack overcloud deploy` process. As a result, you might need to re-add this configuration manually each time. It is expected that a future release of director will include the Puppet parameters that will allow you to re-apply these settings automatically using a post-deployment script.

4. Configure an additional back end:

   a. Create the keystone domain for LDAP integration.

      Select a name to use for your new keystone domain, and create the domain. For example, this command creates a keystone domain named `LAB`:

      ```bash
      # openstack domain create LAB
      ```

      **NOTE**

      If this command is not available, check that you have enabled `keystone v3` for your command line session.

   b. Create the configuration file:

      To add the LDAP back end, enter the LDAP settings in a new file called `/etc/keystone/domains/keystone.LAB.conf` (where `LAB` is the domain name created previously). You will need to edit the sample settings below to suit your LDAP deployment:
```
[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
```

Explanation of each setting:

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>url</td>
<td>The LDAP server to use for authentication. Uses LDAPS port 636.</td>
</tr>
<tr>
<td>user</td>
<td>The account in LDAP to use for LDAP queries.</td>
</tr>
<tr>
<td>password</td>
<td>The plaintext password of the LDAP account used above.</td>
</tr>
<tr>
<td>user_filter</td>
<td>Filters the users presented to Identity Service. As a result, only members of the grp-openstack group can have permissions defined in Identity Service.</td>
</tr>
<tr>
<td>user_tree_dn</td>
<td>The path to the OpenStack accounts in LDAP.</td>
</tr>
<tr>
<td>user_objectclass</td>
<td>Defines the type of LDAP user. For LDAP, use the inetUser type.</td>
</tr>
<tr>
<td>user_id_attribute</td>
<td>Maps the LDAP value to use for user IDs.</td>
</tr>
<tr>
<td>user_name_attribute</td>
<td>Maps the LDAP value to use for names.</td>
</tr>
<tr>
<td>user_mail_attribute</td>
<td>Maps the LDAP value to use for user email addresses.</td>
</tr>
<tr>
<td>user_pass_attribute</td>
<td>Leave this value blank.</td>
</tr>
<tr>
<td>Setting</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>user_allow_create</td>
<td>Set this value to <strong>False</strong>, as keystone only requires read-only access.</td>
</tr>
<tr>
<td>user_allow_update</td>
<td>Set this value to <strong>False</strong>, as keystone only requires read-only access.</td>
</tr>
<tr>
<td>user_allow_delete</td>
<td>Set this value to <strong>False</strong>, as keystone only requires read-only access.</td>
</tr>
</tbody>
</table>

5. Change ownership of the config file to the keystone user:

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

6. Grant the admin user access to the domain:

**NOTE**

This does not grant the OpenStack admin account any permissions in LDAP. In this case, the term domain refers to OpenStack’s usage of the keystone domain.

a. Get the **ID** of the *LAB* domain:

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

b. Get the **ID** value of the *admin* user:

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

c. Get the **ID** value of the *admin* role:

```
# openstack role list
+--------+-------------------+
| ID     | Name              |
+--------+-------------------+
| 5444d48affde48f1b3c31a52c35f01f9 | SwiftOperator    |
| 6d005d783bf0436e882c55c62457d33d | ResellerAdmin    |
```
d. Use the returned domain and admin IDs to construct the command that adds the admin user to the admin role of the keystone LAB domain:

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

7. Configure the dashboard to use the LAB keystone domain at the login page. Add these lines to `/etc/openstack-dashboard/local_settings`:

```
IMPORTANT
Multidomain Dashboard configuration is not supported in this version of Red Hat OpenStack Platform. As a result, this guide only describes Dashboard configuration for a single domain.

OPENSTACK_API_VERSIONS = {
    "identity": 3
}
OPENSTACK_KEYSTONE_DEFAULT_DOMAIN = 'LAB'
```

```
NOTE
If you are using Red Hat OpenStack Platform director, then you will need to be aware that `/etc/openstack-dashboard/local_settings` is managed by Puppet. Consequently, any custom configuration you add might be overwritten whenever you run the `openstack overcloud deploy` process. As a result, you might need to re-add this configuration manually each time. It is expected that a future release of director will include the Puppet parameters that will allow you to re-apply these settings automatically using a post-deployment script.
```

8. Restart the `httpd` service to apply the changes:

```
# systemctl restart httpd.service
```

9. View the list of users in the LDAP domain by adding the keystone domain name to the command:

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

10. View the service accounts in the local keystone database:

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

3.8.3. Configure Compute to use keystone v3

Compute uses keystone v2.0 by default, and so needs to be configured to use keystone v3 in order to use multi-domain capabilities.

1. On each Compute node, and the controller, adjust the `keystone_authtoken` value:
# crudini --set /etc/nova/nova.conf keystone_authtoken auth_version v3

2. Restart these services on the controller to apply the changes:

```bash
# 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. Restart this service on each Compute node to apply the changes:

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

### 3.8.4. Configure Block Storage to use keystone v3

You must also configure Block Storage (cinder) to authenticate to keystone v3.

1. In `/etc/cinder/cinder.conf`:

   ```
   [keystone_authtoken]
   auth_uri = https://controllerIP:5000/v3
   auth_version = v3
   
   *auth_uri* - replace `controllerIP` with the IP address of the controller. If your deployment has more than one controller, you should use the keystone endpoint VIP instead of the controller IP.
   ```

### 3.8.5. Allow LDAP users to access Projects

LDAP users that are members of the `grp-openstack` LDAP group can be granted permission to log in to a project in the dashboard:

1. Retrieve a list of LDAP users:

   ```bash
   # openstack user list --domain LAB
   +--------------------------+-
   | ID                      | Name       |
   +--------------------------+-
   | 1f24ec1f11aeb9052007c29f70afa060d22e2ce92b2e7a784c841ac418091e | user1     |
   | 12c062fLDAP5f8b065434d9ff6f6e3eb9259537c93b4122458886e9a38bf1 | user2     |
   | afaf48031eb543c3e4e4cb0353f5b61204033f70f63c22873d181fdae2e73c | user3     |
   | e47fc21dcf0d9716c2663766023e22dc15a6d9b01453854a898cab2396826e | user4     |
   +--------------------------+-
   ```

2. Retrieve a list of roles:
3. Grant users access to Projects by adding them to one or more of these roles. For example, if you want user1 to be a general user of the demo project, you add them to the _member_ role:

```bash
# openstack role add --project demo --user 1f24ec1f11ae90520079c29f70afa060d22e2ce92b2eba7784c841ac418091e _member_
```

Or, if you want user1 to be an administrative user of the demo project, you add them to the admin role:

```bash
# openstack role add --project demo --user 1f24ec1f11ae90520079c29f70afa060d22e2ce92b2eba7784c841ac418091e admin
```

As a result, user1 is able to log in to the dashboard by entering their LDAP username and password.

**NOTE**

If users receive the error "Error: Unable to retrieve container list.", and expect to be able to manage containers, then they must be added to the SwiftOperator role.

### 3.9. GRANT ACCESS TO THE DOMAIN TAB

To allow the admin user to see the Domain tab, you will need to assign it the admin role in the default domain:

1. Find the admin user's UUID:

   ```bash
   $ openstack user list | grep admin
   | a6a8adb6356f4a879f079485dad1321b | admin
   ```

2. Add the admin role in the default domain to the admin user:

   ```bash
   $ openstack role add --domain default --user a6a8adb6356f4a879f079485dad1321b admin
   ```

   As a result, the admin user can now see the Domain tab.

### 3.10. CREATING A NEW PROJECT

After you have completed these integration steps, when you create a new project you will need to decide whether to create it in the Default domain, or in the keystone domain you've just created. This decision can be reached by considering your workflow, and how you administer user accounts. The Default domain can be be thought of as an internal domain, used for service accounts and the admin
project, so it might make sense for your AD-backed users to be placed within a different keystone domain; this does not strictly need to be the same keystone domain as the LDAP users are in, and for separation purposes, there might be multiple keystone domains.

3.10.1. Changes to the command line

For certain commands, you might need to specify the applicable domain. For example, appending --domain LAB in this command returns users in the LAB domain (that are members of the grp-openstack group):

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

Appending --domain Default returns the built-in keystone accounts:

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

3.10.2. Test LDAP integration

This procedure validates LDAP integration by testing user access to dashboard features:

1. Create a test user in LDAP, and add the user to the grp-openstack LDAP group.
2. Add the user to the _member_ role of the demo tenant.
3. Log in to the dashboard using the credentials of the LDAP test user.
4. Click on each of the tabs to confirm that they are presented successfully without error messages.
5. Use the dashboard to build a test instance.

**NOTE**

If you experience issues with these steps, perform steps 3-5 with the built-in admin account. If successful, this demonstrates that OpenStack is still working as expected, and that an issue exists somewhere within the LDAP → Identity integration settings. See Section 3.13, “Troubleshooting”.

3.11. CONFIGURE FOR HIGH AVAILABILITY

With keystone v3 enabled, you can make this configuration highly available by listing multiple LDAP servers in the configuration file for that domain.

1. Add a second server to the url entry. For example, updating the url setting in the keystone.LAB.conf file will have Identity Service send all query traffic to the first LDAP server in the list, ldap.lab.local:

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

If a query to ldap.lab.local fails due to it being unavailable, Identity Service will attempt to query the next server in the list: ldap2.lab.local. Note that this configuration does not perform queries in a round-robin fashion, so cannot be considered a load-balancing solution.

2. Set the network timeout in /etc/openldap/ldap.conf:
In addition, if you have firewalls configured between the controller and the LDAP servers, then you should not configure the LDAP servers to silently drop packets from the controller. This will allow python-keystoneclient to properly detect outages and redirect the request to the next LDAP server in the list.

NOTE

There might be connection delays while queries are being redirected to the second LDAP server in the list. This is because the connection to the first server must first time out before the second is attempted.

3.12. CREATE A RC FILE FOR A NON-ADMIN USER

You might need to create a RC file for a non-admin user. For example:

```bash
$ cat overcloudrc-v3-user1
# Clear any old environment that may conflict.
for key in $( set | awk '{FS="="} /^OS_/ {print $1}' ); do unset $key ; done
export OS_USERNAME=user1
export NOVA_VERSION=1.1
export OS_PROJECT_NAME=demo
export OS_PASSWORD=RedactedComplexPassword
export OS_NO_CACHE=True
export COMPUTE_API_VERSION=1.1
export no_proxy=,10.0.0.5,192.168.2.11
export OS_CLOUDNAME=overcloud
export OS_AUTH_URL=https://10.0.0.5:5000/v3
export OS_AUTH_TYPE=password
export PYTHONWARNINGS="ignore:Certificate has no, ignore:A true SSLContext object is not available"
export OS_IDENTITY_API_VERSION=3
export OS_PROJECT_DOMAIN_NAME=Default
export OS_USER_DOMAIN_NAME=LAB
```

3.13. TROUBLESHOOTING

3.13.1. Test LDAP connections

Use ldapsearch to remotely perform test queries against the LDAP server. A successful result here indicates that network connectivity is working, and the LDAP services are up. In this example, a test query is performed against the server ldap.lab.local on port 636:

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

NOTE

ldapsearch is a part of the openldap-clients package. You can install this using # yum install openldap-clients.
3.13.2. Test port access

Use `nc` to check that the LDAPS port (636) is remotely accessible. In this example, a probe is performed against the server `ldap.lab.local`. Press `ctrl-c` to exit the prompt.

```
# nc -v ldap.lab.local 636
^C
```

Failure to establish a connection could indicate a firewall configuration issue.