Red Hat CloudForms 5.0

Managing Authentication for CloudForms

Authentication methods for Red Hat CloudForms

Last Updated: 2020-08-05

Red Hat CloudForms 5.0 Managing Authentication for CloudForms

Authentication methods for Red Hat CloudForms

Red Hat CloudForms Documentation Team
cloudforms-docs@redhat.com

Legal Notice

Copyright © 2020 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, the Red Hat 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 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.

Abstract

This guide provides instructions for configuring and managing authentication on Red Hat CloudForms. Authentication with CloudForms can be configured using the local database, or authentication systems such as Red Hat Identity Management (IdM), Red Hat Single Sign-On (SSO), Active Directory (AD), or AWS Identity and Access Management (IAM), through protocols such as LDAP and SAML. If you have a suggestion for improving this guide or have found an error, please submit a Bugzilla report at http://bugzilla.redhat.com against Red Hat CloudForms Management Engine for the Documentation component. Please provide specific details, such as the section number, guide name, and CloudForms version so we can easily locate the content.

Table of Contents

CHAPTER 1. INTRODUCTION TO AUTHENTICATION IN CLOUDFORMS

Red Hat CloudForms provides several methods to authenticate users. Authentication can be configured from CloudForms using the local database, or CloudForms can use protocols such as LDAP and SAML to connect to a pre-configured authentication system such as Red Hat Identity Management (IdM), Red Hat Single Sign-On (SSO), Active Directory (AD), or AWS Identity and Access Management (IAM) to use existing user accounts and groups.

This guide provides instructions to configure authentication management in your CloudForms environment as an administrative user.

After completing the setup of an authentication system, users can log in with their credentials.

NOTE

For further information on managing users, groups, and account roles, see Access Control in *General Configuration*.

1.1. CONFIGURING AUTHENTICATION SETTINGS IN CLOUDFORMS

As the admin user, configure your authentication method from the **Authentication** tab in the **Configuration** menu.

To change authentication settings:

1. Click ⚙ (**Configuration**).
2. Select your server in the **Settings** accordion.
3. Select the **Authentication** tab.
4. Use **Session Timeout** to set the period of inactivity before a user is logged out of the console.
5. Set the authentication method in **Mode** from the following methods:
   - To configure authentication locally using the Virtual Management Database (VMDB), choose **Database**. This is the default method. See Creating a User in *General Configuration* to create users from CloudForms.
   - To configure LDAP-based authentication to use with IdM or Active Directory, choose **LDAP** or **LDAPS**, see Chapter 2, *Configuring LDAP Authentication with IdM and Active Directory* for configuration steps.
   - To configure Amazon AWS Identity and Access Management (IAM) authentication, choose **Amazon**, see Chapter 3, *Configuring AWS Identity and Access Management (IAM) Authentication* for configuration steps.
   - To configure federated authentication to use with IdM or Red Hat Single Sign-On (SSO), choose **External (httpd)** and follow the steps for your authentication method in Chapter 4, *Configuring Identity Management (External Authentication) with CloudForms* .

CHAPTER 2. CONFIGURING LDAP AUTHENTICATION WITH IDM AND ACTIVE DIRECTORY

Select **LDAP** or **LDAPS** to configure CloudForms authentication using Red Hat Identity Management (IdM), Active Directory, or another identity management service that uses LDAP protocol.

If you choose LDAP or LDAPS as your authentication mode, the required parameters are exposed under **LDAP Settings**. Be sure to validate your settings before saving them.

IMPORTANT

This procedure requires a preconfigured authentication system such as Red Hat Identity Management (IdM) or Active Directory (AD) with user groups configured.

LDAP authentication for CloudForms requires group membership to be defined by the LDAP RFC 2307 schema, where group members are listed by name in the member UID attribute.

For more information about Red Hat Identity Management, see the Linux Domain Identity, Authentication, and Policy Guide and related Red Hat Enterprise Linux documentation.

2.1. CONFIGURING LDAP OR LDAPS AUTHENTICATION

To configure CloudForms to use LDAP for authentication:

1. Click ⚙ (**Configuration**).
2. Select your server in the **Settings** accordion.
3. Select the **Authentication** tab.
4. Select a **Session Timeout** to set the period of inactivity before a user is logged out of the console.
5. Select **LDAP** or **LDAPS** from the **Mode** list. This exposes additional required parameters under **LDAP Settings**.
6. Configure your **LDAP Settings** (the following example configures an IdM directory server):
   - Use **LDAP Host Names** to specify the fully qualified domain names of your directory servers. CloudForms will search each host name in order until it finds one that authenticates the user. Note, CloudForms supports using a maximum of three possible **LDAP Host Names**.
   - Use **LDAP Port** to specify the port for your directory server. The default is 389 for LDAP and 636 for LDAPS.
   - From the **User Type** list, select one of the following and configure the values for your directory server:
     - **User Principal Name**: Type the user name in the format of *user@domainname*, for example, *dbright@acme.com*. (In this case, the user would log on as *dbright*.)
     - **Email Address**: Logs in with the user's email address.
     - **Distinguished Name** (CN=<user>): Uses the common name for the user. Be sure to

enter the correct **User Suffix** and **Distinguished Name** option for your directory service implementation: for example, *cn=dan bright,ou=users,dc=acme,dc=com*. (The user logs on as *dan bright*.)

- **Distinguished Name** (UID=<user>): Uses the user ID (UID). Be sure to enter the correct **User Suffix** and **Distinguished Name** option for your directory service implementation: for example, *uid=dan bright,ou=users,dc=acme,dc=com*. (The user logs on as *dan bright*.)

- **SAM Account Name**: User logon for Active Directory clients and servers using legacy Windows versions. You must also configure the **Domain Prefix** in the next field when selecting this option.

- Specify the **Domain Prefix** if you are configuring an Active Directory LDAP host, and selected **SAM Account Name** as the **User Type**. This field represents the prefix name in the Active Directory domain. Together with the SAM account name, this constructs the fully qualified user name in the format **<domain_prefix>\<user>**.

- Specify the **User Suffix**, such as *acme.com* for **User Principal Name** or *cn=users,dc=acme,dc=com* for **Distinguished Name**, in **Base DN**.

NOTE

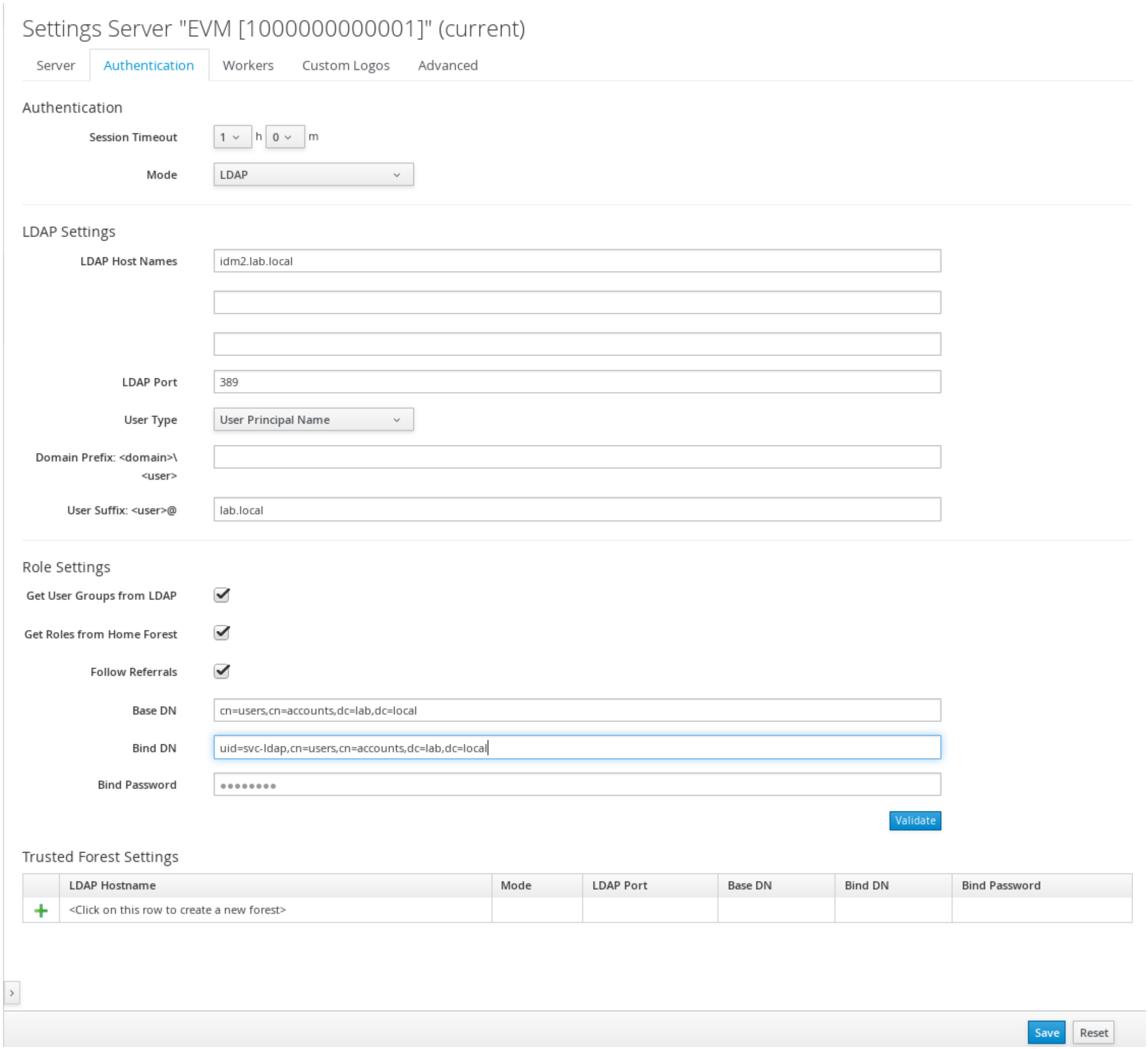
The **ldapsearch(1)** command can be used to get details of your LDAP settings. To get details related to a specific user, run:

```
# ldapsearch -D "cn=directory manager" -H ldap://www.acme.com:389 -b
"dc=acme,dc=com" -s sub "(objectclass=*)" -w password | grep -i dbright
```

To search for your directory server's distinguished name (DN) values, run:

```
# ldapsearch -D "cn=directory manager" -H ldap://www.acme.com:389 -b
"dc=acme,dc=com" -s sub "(objectclass=*)" -w password
```

Example: LDAP Configuration



Settings Server "EVM [1000000000001]" (current)

Authentication

Session Timeout: 1 h 0 m

Mode: LDAP

LDAP Settings

LDAP Host Names: idm2.lab.local

LDAP Port: 389

User Type: User Principal Name

Domain Prefix: <domain>\<user>

User Suffix: <user>@ lab.local

Role Settings

Get User Groups from LDAP ☑

Get Roles from Home Forest ☑

Follow Referrals ☑

Base DN: cn=users,cn=accounts,dc=lab,dc=local

Bind DN: uid=svc-ldap,cn=users,cn=accounts,dc=lab,dc=local

Bind Password: ••••••••

Trusted Forest Settings

	LDAP Hostname	Mode	LDAP Port	Base DN	Bind DN	Bind Password
+	<Click on this row to create a new forest>					

7. Configure your **Role Settings**: In both LDAP and LDAPS, you can use groups from your directory service to set the role for the authenticated LDAP user. The LDAP user must be assigned one of the account role groups. See Section 2.3, “Assigning CloudForms Account Roles Using LDAP Groups” for more information.

   - For LDAP users not belonging to a group:

     - Select a CloudForms group from the **Default Group for Users** list. This default group can be used for all LDAP users who use LDAP for authentication only. Do not select **Get User Groups from LDAP**, which will hide the **Default Group for Users** option.

   - For LDAP users belonging to a group:

     - Select **Get User Groups from LDAP** to retrieve the user’s group membership from LDAP. This looks up all groups in the directory server for the user attempting to log in. This group list is matched against the available groups configured in CloudForms. Once a match is found, the role associated with the matching group identifies the authority the user will have on CloudForms. This requires group names on the directory server to match CloudForms group names. Selecting this box enables user records to be automatically created in CloudForms when a user logs in.

IMPORTANT

If you do not select **Get User Groups from LDAP**, the user must be defined in the VMDB where the User ID is the same as the user's name in your directory service typed in lowercase. See Creating a User in *General Configuration* for steps on creating users.

NOTE

If your LDAP directory uses the **groupMembership** attribute to contain a user's group membership, you can use a multi-value attribute to look up a user's groups in CloudForms. This is not the default CloudForms behaviour, and must be enabled manually in the advanced settings. After configuring your LDAP settings, update the default CloudForms LDAP configuration to support **groupMembership** in **Configuration → Settings → Advanced** by changing **group_attribute: memberof** to **:group_attribute: groupmembership**.

- Select **Get Groups from Home Forest** to use the LDAP groups from the LDAP user's home forest. This will allow you to discover groups on your directory server and create CloudForms groups based on your directory server's group names. Any user logging in will be assigned to that group. This option is only displayed when **Get User Groups from LDAP** is selected.

NOTE

In most environments, it is recommended to select both the **Get User Groups from LDAP** and **Get Groups from Home Forest** options.

- Select **Follow Referrals** to look up and bind a user that exists in a domain other than the one configured in the LDAP authentication settings.

- Specify the user name to bind to the directory server in **Bind DN**. This user must have read access to all users and groups that will be used for CloudForms authentication and role assignment, for example, a service account user with access to all LDAP users (named *svc-ldap* in this example).

- Enter the password for the Bind DN user in **Bind Password**.

8. Click **Validate** to verify your settings.

9. Click **Save**.

LDAP authentication is now configured in your CloudForms environment.

To use a multi-value attribute to look up LDAP group membership, update the **group_attribute** field in the CloudForms advanced settings. In **Configuration → Settings → Advanced**, change **group_attribute: memberof** to **:group_attribute: groupmembership**.

To assign account roles using LDAP groups, see Section 2.3, "Assigning CloudForms Account Roles Using LDAP Groups".

2.2. ADDING TRUSTED FORESTS

Optionally, if a user has group memberships in another LDAP forest, specify the settings to access the memberships in the trusted forest.

When trusted forests are added to the authentication configuration, they are used only for finding groups that a user is a member of. CloudForms will first collect all of the user’s groups from the primary LDAP directory. Then it will collect any additional groups that the user is a member of from all of the configured forests.

The collected LDAP groups are used to match, by name, against the groups defined in CloudForms. The user must be a member of at least one matching LDAP group to be successfully authenticated.

To add another trusted forest:

1. Click ⚙ (**Configuration**).
2. Select your server in the **Settings** accordion.
3. Select the **Authentication** tab.
4. Select **Get User Groups from LDAP**, and enter all items in the **Role Settings** area.
5. In the **Trusted Forest Settings** area, click ➕ (**Click to add a new forest**).
6. Enter the **LDAP Host Name**, select a **Mode**, and enter an **LDAP Port**, **Base DN**, **Bind DN**, and **Bind Password**.
7. Click **Save**.

After adding other trusted LDAP forests, you can then change the order in which CloudForms looks up the forests for authentication. For instructions, see Section 2.4, “Configuring Lookup Priority for LDAP Groups”.

2.3. ASSIGNING CLOUDFORMS ACCOUNT ROLES USING LDAP GROUPS

After configuring LDAP authentication as described in Chapter 2, *Configuring LDAP Authentication with IdM and Active Directory*, you can associate CloudForms account roles with your LDAP users. The LDAP directory server defines the groups and users for CloudForms, while CloudForms defines the account roles, and maps the roles to the privileges the LDAP user has.

There are two ways to associate your LDAP groups with CloudForms account roles:

- Create groups in CloudForms that match your existing LDAP groups by name, and assign the groups account roles; or
- Create groups on your directory server based on the default account roles in CloudForms.

The users in your LDAP groups then inherit the CloudForms account roles for the LDAP group(s) they are in.

The authentication process then happens as such:

1. *LDAPuser1* attempts to log into CloudForms, so CloudForms queries the directory server to verify it knows *LDAPuser1*.

2. The directory server then confirms that it knows *LDAPuser1*, and provides information about the LDAP groups *LDAPuser1* belongs to: *Group1*.

3. CloudForms then looks up *Group1*, and discovers that *Group1* is associated with *Role1*.

4. CloudForms then associates *LDAPuser1* with *Group1* in CloudForms, and then allows the user to perform tasks allowable by that role.

2.3.1. Mapping Existing LDAP Groups to CloudForms User Account Roles

This section provides instructions for mapping your existing LDAP groups to account roles in CloudForms. As a result, the users in the LDAP group will then be assigned to the CloudForms roles associated with that group.

1. Click ⚙ (**Configuration**).

2. Click the **Access Control** accordion, then click **Groups**.

3. Click ⚙ (**Configuration**), and ⊕ (**Add a new Group**) to create a group.

4. There are two ways to specify the group to use:

   - In the **Description** field, enter the common name (*cn*) for your existing LDAP group assigned to users requiring access to CloudForms.

   - Select **Look Up LDAP Groups** to find a list of groups assigned to a specific user in LDAP, then use the **LDAP Group for User** list to choose a group.

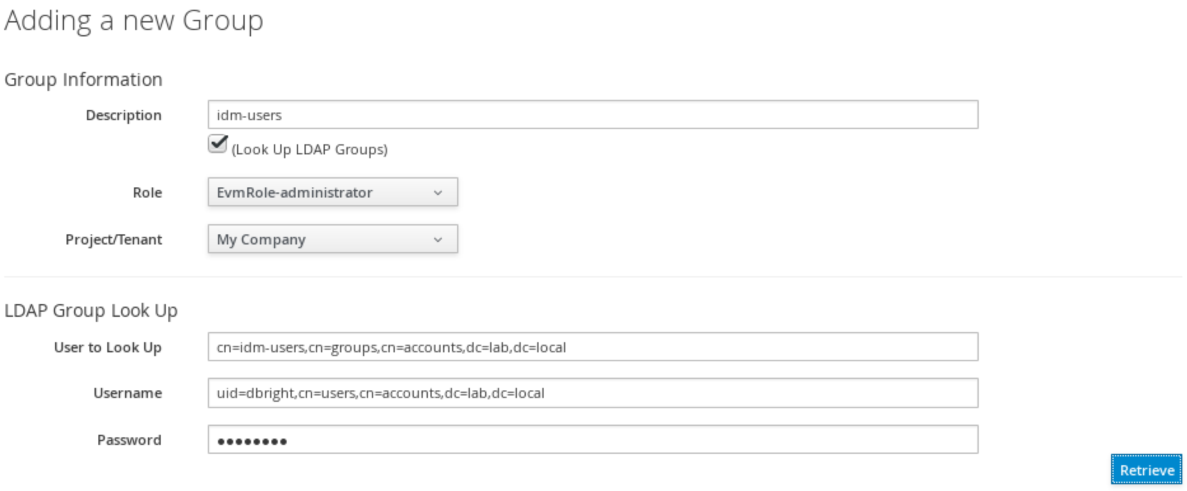
     a. In **User to Look Up**, enter the common name (*cn*) for a user in your LDAP group.

     b. Enter the **Username**.

     c. In **Password**, enter the user's password. Click **Retrieve**.

5. Select a **Role** to map to the group.

6. Select a **Project/Tenant** to map to the group.



Adding a new Group

Group Information

Description: idm-users

☑ (Look Up LDAP Groups)

Role: EvmRole-administrator

Project/Tenant: My Company

LDAP Group Look Up

User to Look Up: cn=idm-users,cn=groups,cn=accounts,dc=lab,dc=local

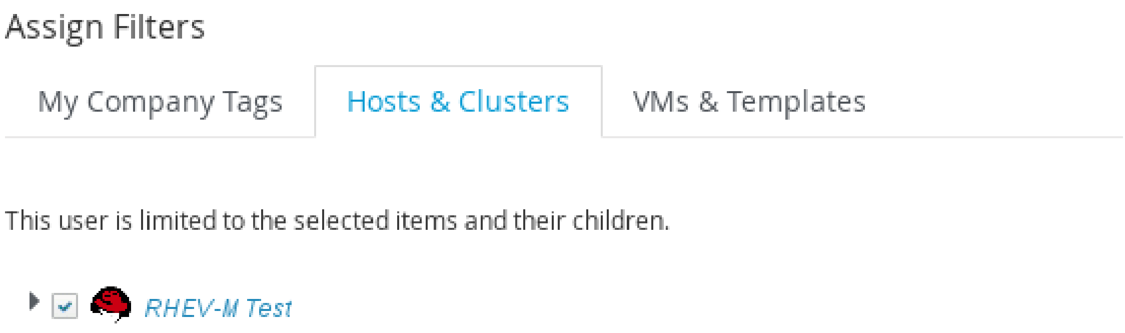
Username: uid=dbright,cn=users,cn=accounts,dc=lab,dc=local

Password: ••••••••

7. Select any filters to apply to what this group can view in the **Assign Filters** area:

   a. In the **My Company Tags** tab, select tags to limit the user to items containing those tags. The items that have changed show in a blue italicized font.

b. In the **Host & Clusters** tab, select the host and clusters to limit the user to. The items that have changed show in a blue italicized font.



Assign Filters

This user is limited to the selected items and their children.

☑ *RHEV-M Test*

c. In the **VMs & Templates** tab, select the folders created in your virtual infrastructure to limit the user to. The items that have changed show in a blue italicized font.

8. Click **Add**.

To configure the LDAP group lookup priority, see Section 2.4, "Configuring Lookup Priority for LDAP Groups".

2.3.2. Creating LDAP Groups Based on CloudForms Account Roles

You can also configure access control for LDAP users by creating groups on your directory server based on CloudForms user account roles.

Your LDAP group names must match the account role names in CloudForms. The LDAP users in that group are then automatically assigned to that specific account role.

In your LDAP directory service:

1. Define a distribution group for one or more of the account roles with the names shown in the table below. This group must be in the LDAP directory source you specified for the server. See *Chapter 2, Configuring LDAP Authentication with IdM and Active Directory* .

Table 2.1. Account Role and Directory Service Group Names

Directory Service Distribution Group Name	Account Role
EvmGroup-administrator	Administrator
EvmGroup-approver	Approver
EvmGroup-auditor	Auditor
EvmGroup-consumption_administrator	Consumption Administrator
EvmGroup-container_administrator	Container Administrator
EvmGroup-container_operator	Container Operator

Directory Service Distribution Group Name	Account Role
EvmGroup-desktop	Desktop
EvmGroup-operator	Operator
EvmGroup-security	Security
EvmGroup-super_administrator	Super Administrator
EvmGroup-support	Support
EvmRole-tenant_administrator	Tenant Administrator
EvmRole-tenant_quota_administrator	Tenant Quota Administrator
EvmGroup-user	User
EvmGroup-user_limited_self_service	User Limited Self Service
EvmGroup-user_self_service	User Self Service
EvmGroup-vm_user	VM User

2. Assign each user of your directory service that you want to have access to CloudForms membership to one of these groups.

On your CloudForms appliance:

1. Click ⚙ (**Configuration**).
2. Click the **Settings** accordion, then select your server under **Zones**.
3. Click the **Authentication** tab and enable **Get User Groups from LDAP** after typing in all of the required LDAP authentication settings. See *Chapter 2, Configuring LDAP Authentication with IdM and Active Directory*.

2.4. CONFIGURING LOOKUP PRIORITY FOR LDAP GROUPS

CloudForms can have multiple LDAP groups configured, which the appliance will attempt to authenticate with one by one until it succeeds. The lookup priority of these groups can be rearranged.

NOTE

On initial login, a user's *current group* assignment is the highest priority group. User group membership, on subsequent logins, is set as the last assigned group from the prior session.

To configure the order in which CloudForms looks up LDAP groups:

1. Click ⚙ (**Configuration**).
2. Click on the **Access Control** accordion, then click **Groups**.
3. Click ⚙ (**Configuration**), and ⊕ (**Edit Sequence of User Groups for LDAP Look Up**) to prioritize which group a user will default to if LDAP returns multiple matching groups.
4. Select one or more consecutive groups and use the arrow buttons to move the user group higher or lower in priority.
5. Click **Save**.

2.5. TESTING LDAP CONFIGURATION

To test that your LDAP or LDAPS group configuration is working correctly with CloudForms:

1. Log out of the CloudForms user interface.
2. Log back in as an LDAP user that is assigned to one or more of the matching groups.
3. Change groups by clicking on the user dropdown menu on the top right of the user interface. The dropdown list will show the groups the user is authorized for.

You can also check the logs in **/var/www/miq/vmdb/log/audit.log** or **/var/www/miq/vmdb/log/evm.log** to verify your LDAP configuration is working correctly with the following steps:

1. Run the following command in a terminal to view the log messages in real time:

   ```
   $ tail -f /var/www/miq/vmdb/log/audit.log
   ```

2. Log into the CloudForms user interface as an LDAP user, while checking **/var/www/miq/vmdb/log/audit.log** for updated status, success, or failure messages. Alternatively, you can test your LDAP configuration by viewing the logs in **/var/www/miq/vmdb/log/evm.log** with **grep**, which are more verbose.

2.6. TROUBLESHOOTING LDAP CONFIGURATION

To test a problematic CloudForms LDAP configuration, run the following command to see if the user been pulled from LDAP with the right group. For example:

```
# ldapsearch -x -H ldap://ldap-example:389 -LLL \ -b "ou=people,dc=example,dc=com" -s sub \ -D
"ui=:userid,ou=People,dc=example,dc=com" -w :password \ "(objectclass=organizationalPerson)
```

To test if the user belongs to right group, include one of the following lines in the **ldapsearch** command above:

```
(&(objectClass=user)(sAMAccountName=yourUserName)
(memberof=CN=YourGroup,OU=Users,DC=YourDomain,DC=com))
```

or

```
-b "ou=groups, dc=example,dc=com"
```

CHAPTER 3. CONFIGURING AWS IDENTITY AND ACCESS MANAGEMENT (IAM) AUTHENTICATION

If you choose Amazon AWS Identity and Access Management (IAM) as your authentication mode, required parameters are exposed under **Amazon Primary AWS Account Settings for IAM**(Identity and Access Management). Be sure to validate your settings before saving them.

> **NOTE**
>
> For further information on AWS Identity and Access Management settings, see the AWS documentation.

To configure CloudForms to use AWS IAM authentication:

1. Click ⚙ (**Configuration**).
2. Select your server in the **Settings** accordion.
3. Select the **Authentication** tab.
4. Use **Session Timeout** to set the period of inactivity before a user is logged out of the console.
5. Select **Amazon** as the authentication method in the **Mode** list.
6. Type in an **Access Key** provided by your Amazon account.
7. Type in a **Secret Key** provided by your Amazon account.
8. Optionally, select **Get User Groups from Amazon** to retrieve the user's group membership from Amazon. This is used for mapping a user's authorization to a CloudForms role.
9. Click **Validate** to verify your settings.
10. Click **Save**.

Users logging into CloudForms with Amazon authentication enter their own IAM access key as the username and IAM secret access key as the password.

Amazon users must be added as a CloudForms user or belong to an IAM user group added to the list of CloudForms groups.

CHAPTER 4. CONFIGURING IDENTITY MANAGEMENT (EXTERNAL AUTHENTICATION) WITH CLOUDFORMS

You can configure CloudForms to use system authentication methods such as Red Hat Identity Management (IdM) or IPA, Red Hat Single Sign-On (SSO), or Active Directory (AD).

This method uses **apache** (**httpd**) modules with web browsers to control authentication to CloudForms. It is the recommended authentication method to connect CloudForms with most identity management services.

4.1. CONFIGURING AUTHENTICATION WITH IPA

You can configure CloudForms to use IPA with the **External Authentication (httpd)** option in CloudForms.

When external authentication is enabled, users can log in to the CloudForms appliance using their IPA server credentials. The appliance creates user accounts automatically and imports relevant information from the IPA server.

The appliance contains IPA client software for connecting to IPA servers, but it is not configured by default. External authentication is enabled by configuring it with the appliance console and enabling it the web interface.

Disabling external authentication and returning to internal database authentication also requires steps in both the appliance console and the web user interface.

4.1.1. External Authentication Requirements

- For an appliance to leverage an IPA server on the network, both the appliance and the IPA server must have their clocks synchronized or Kerberos and LDAP authentication fail.

- The IPA server must be known by DNS and accessible by name. If DNS is not configured accordingly, the hosts files need to be updated to reflect both IPA server and the appliance on both virtual machines.

- For users to log in to the appliance using IPA server credentials, they must be members of at least one group on the IPA server which is also defined in the appliance. Navigate to the settings menu, then **Configuration** → **Access Control** → **Groups** to administer groups.

4.1.2. Configuring the Appliance for External Authentication

To configure the appliance for external authentication, set up authentication using the appliance console, then select the **External Authentication** option in the web user interface.

Using the appliance console:

1. Log in to the appliance console using the user name **admin**.

2. The summary screen displays:

   ```
   External Auth:  not configured
   ```

3. Press Enter.

4. Select **Configure External Authentication (httpd)**.
5. Enter the fully qualified host name of the IPA server, for example *ipaserver.test.company.com*.
6. Enter the IPA server domain, for example *test.company.com*.
7. Enter the IPA server realm, for example *TEST.COMPANY.COM*.
8. Enter the IPA server principal, for example *admin*.
9. Enter the password of the IPA server principal.
10. Enter **y** to proceed.

NOTE

If any of the following conditions are true, configuration fails:

- The IPA server is not reachable by its FQDN
- The IPA server cannot reach the appliance by its FQDN
- The time is not synchronized between the appliance and the IPA server
- The IPA server admin password is entered incorrectly

Alternatively, you can configure external authentication using the **appliance_console_cli** command instead of using the appliance console menu:

Example 4.1. Configuring External Authentication

```
$ ssh root@appliance.test.company.com
[appliance]# /bin/appliance_console_cli --host appliance.test.company.com \
                                        --ipaserver ipaserver.test.company.com \
                                        --iparealm TEST.COMPANY.COM \
                                        --ipaprincipal admin \
                                        --ipapassword smartvm1
```

Finish configuring external authentication using the web user interface:

1. Log in to the web user interface as an administrative user.
2. Navigate to the settings menu, then **Configuration → Settings → Zone → Server → NTP Servers** or use the hosting provider of the virtual machine to synchronize the appliance’s time with an NTP server.
3. Click ⚙ (**Configuration**).
4. Select your server in the **Settings** accordion.
5. Select the **Authentication** tab.
6. Select a **Session Timeout** if required.

7. Select **External (httpd)** in the **Mode list**.

8. Select **Enable Single Sign-On** to allow single sign-on using Kerberos tickets from client machines that authenticate to the same IPA server as the appliance.

9. In the **Role Settings** area, select **Get User Groups** from **External Authentication (https)**.

10. Click **Save**.

4.1.3. Reverting to Internal Database Authentication

To revert to internal database authentication, first configure authentication using the web user interface, then using the appliance console.

Using the web user interface:

1. Click ⚙ (**Configuration**).

2. Select your server in the **Settings** accordion.

3. Select the **Authentication** tab.

4. Select **Database** in the **Mode** list.

5. Click **Save**.

Using the appliance console:

1. Log in to the appliance console using the user name **admin**.

2. The summary screen displays:

   ```
   External Auth:  ld.server.FQDN
   ```

3. Press **Enter**.

4. Select **Configure External Authentication (httpd)**. The currently configured IPA server host name and domain are displayed.

5. Enter **y** to remove configuration details for the IPA client.

Example 4.2. Reverting to Internal Database Authentication

```
$ ssh root@appliance.test.company.com
[appliance]# /bin/appliance_console_cli --uninstall-ipa
```

4.1.4. Optional Configuration Using the Appliance Console CLI

In addition to using the appliance console, external authentication can optionally be configured and reverted using the appliance console command line interface.

Appliance console CLI command and relevant options include:

```
/bin/appliance_console_cli --host <appliance_fqdn>
                           --ipaserver <ipa_server_fqdn>
                           --iparealm <realm_of_ipa_server>
                           --ipaprincipal <ipa_server_principal>
                           --ipapassword <ipa_server_password>
                           --uninstall-ipa4.5
```

--host

Updates the host name of the appliance. If you performed this step using the console and made the necessary updates made to **/etc/hosts** if DNS is not properly configured, you can omit the **--host** option.

--iparealm

If omitted, the **iparealm** is based on the domain name of the **ipaserver**.

--ipaprincipal

If omitted, defaults to admin.

4.2. CONFIGURING AUTHENTICATION WITH ACTIVE DIRECTORY

This procedure outlines how to configure CloudForms to authenticate against an existing Active Directory (AD) configuration using external HTTP authentication. This provides Active Directory users access to the CloudForms appliance user interface, as well as the REST API.

4.2.1. Connecting CloudForms to an Active Directory Domain

To use an Active Directory domain to authenticate users to CloudForms, configure the following on CloudForms:

1. Connect to the CloudForms appliance using SSH.

2. Run **realm discover** to determine what Active Directory realms are available:

   ```
   # realm discover example.com
     type: kerberos
     realm-name: EXAMPLE.COM
     domain-name: example.com
     configured: kerberos-member
     server-software: active-directory
     client-software: sssd
     required-package: oddjob
     required-package: oddjob-mkhomedir
     required-package: sssd
     required-package: adcli
     required-package: samba-common
     login-formats: %U@example.com
     login-policy: allow-realm-logins
   ```

3. Using the above information for your realm, join the Active Directory realm with a user that has enough permissions to be able to browse the directory:

   ```
   # realm join example.com -U user
   Password for user: ******
   ```

4. Allow all realm users to log in using **realm permit**:

   ```
   # realm permit --all
   ```

5. Edit the **/etc/sssd/sssd.conf** file with your Active Directory domain details. Refer to the following example for formatting:

   ```
   [domain/example.com]
     ad_domain = example.com
     krb5_realm = EXAMPLE.COM
     realmd_tags = manages-system joined-with-samba
     cache_credentials = True
     id_provider = ad
     krb5_store_password_if_offline = True
     default_shell = /bin/bash
     ldap_id_mapping = True
     use_fully_qualified_names = True
     fallback_homedir = /home/%d/%u
     access_provider = ad
    ldap_user_extra_attrs = mail, givenname, sn, displayname, domainname

   [sssd]
   domains = example.com
   config_file_version = 2
   services = nss, pam, ifp
   default_domain_suffix = example.com

   [nss]
   homedir_substring = /home

   [pam]
   default_domain_suffix = example.com

   [ifp]
   default_domain_suffix = example.com
   allowed_uids = apache, root
   user_attributes = +mail, +givenname, +sn, +displayname, +domainname
   ```

6. Restart and enable the **sssd** service:

   ```
   # systemctl restart sssd
   # systemctl enable sssd
   ```

7. Make sure the Kerberos keytab created by **realm join** above is readable by Apache:

   ```
   # chgrp apache /etc/krb5.keytab
   # chmod 640 /etc/krb5.keytab
   ```

8. Copy the following **httpd** configuration files into the correct respective directories with the following commands:

   ```
   # TEMPLATE_DIR="/opt/rh/cfme-appliance/TEMPLATE"
   # cp ${TEMPLATE_DIR}/etc/pam.d/httpd-auth /etc/pam.d/httpd-auth
   # cp ${TEMPLATE_DIR}/etc/httpd/conf.d/manageiq-remote-user.conf /etc/httpd/conf.d/
   ```

```
# cp ${TEMPLATE_DIR}/etc/httpd/conf.d/manageiq-external-auth.conf.erb
/etc/httpd/conf.d/manageiq-external-auth.conf
```

9. Edit the **/etc/httpd/conf.d/manageiq-external-auth.conf** file to point to the Kerberos domain hosted by your Active Directory domain by adding or editing the lines for **KrbAuthRealms**, **Krb5KeyTab** and **KrbServiceName** for your environment:

```
<Location /dashboard/kerberos_authenticate>
  AuthType           Kerberos
  AuthName           "Kerberos Login"
  KrbMethodNegotiate On
  KrbMethodK5Passwd  Off
  KrbAuthRealms      example.com
  Krb5KeyTab         /etc/krb5.keytab
  KrbServiceName     Any
  Require            pam-account httpd-auth

  ErrorDocument 401  /proxy_pages/invalid_sso_credentials.js
</Location>
```

10. Set the following SELinux booleans:

```
# setsebool -P allow_httpd_mod_auth_pam on
# setsebool -P httpd_dbus_sssd          on
```

11. Restart and enable the **httpd** service:

```
# systemctl restart httpd
# systemctl enable httpd
```

Complete authentication setup by configuring the following on each appliance with the **user_interface** or **web_services** server roles enabled.

From the CloudForms user interface:

1. Log in to the user interface as an administrative user.
2. Navigate to the settings menu, then **Configuration → Authentication**.
3. Select a **Session Timeout** if required.
4. Select **External (httpd)** as the authentication **Mode**.
5. Select **Enable Single Sign-On** to allow single sign-on using Kerberos tickets from client machines that authenticate to the same Active Directory domain as the appliance.
6. In the **Role Settings** area, select **Get User Groups from External Authentication (httpd)**
7. Click **Save**.

IMPORTANT

Make sure the user's Active Directory groups for the appliance are created and appropriate roles assigned to those groups. See Roles in *General Configuration* for more information.

CloudForms is now configured to use authentication from your Active Directory domain.

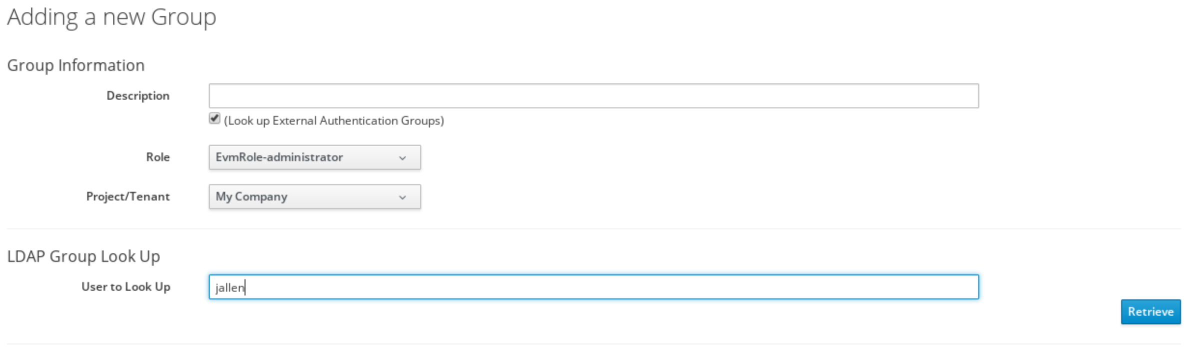
4.2.2. Mapping Active Directory Users to CloudForms User Roles

This section provides instructions for mapping your existing Active Directory (AD) groups to user account roles in CloudForms.

This is done by assigning a CloudForms role to an AD group. When an AD user who is a part of that AD group attempts to log in to CloudForms, they get that role assigned automatically and inherit the permissions from that role. As a result, all users in that AD group will then be assigned the CloudForms role(s) associated with that group.

After configuring CloudForms to connect to an Active Directory domain in Section 4.2.1, “Connecting CloudForms to an Active Directory Domain”, complete the following steps:

1. Click ⚙ (**Configuration**).
2. Click the **Access Control** accordion, then click **Groups**.
3. Click ⚙ (**Configuration**), and ⊕ (**Add a new Group**) to create a group.
4. In **Group Information**, select **Look Up External Authentication Groups**to find a list of groups assigned to a specific user in Active Directory.
5. In **User to Look Up**, enter the user name for a user in your AD group.



Adding a new Group

Group Information

Description

☑ (Look up External Authentication Groups)

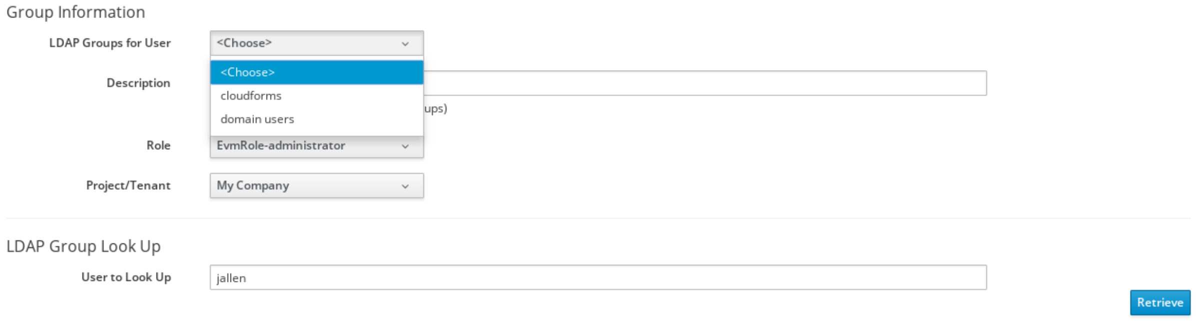
Role: EvmRole-administrator

Project/Tenant: My Company

LDAP Group Look Up

User to Look Up: jallen

6. Click **Retrieve** to look up details for the user in Active Directory and pull group information for the user. As a result, the AD groups will appear in the **LDAP Groups for User**drop-down list.
7. From the list in **LDAP Groups for User**, select the group you want to associate a CloudForms role with; for example, the **cloudforms** group.



Group Information

LDAP Groups for User: <Choose>

Description

ups)

Role: EvmRole-administrator

Project/Tenant: My Company

LDAP Group Look Up

User to Look Up: jallen

8. Select a **Role** to map to the group.
9. Select a **Project/Tenant** to map to the group.

10. Click **Add**.

Any user who is part of **cloudforms** AD group can now log in to CloudForms with their AD username and password, and they will automatically inherit the permissions for the role you assigned earlier.

To confirm this is configured correctly, log in to the CloudForms user interface with a user in the **cloudforms** AD group.

To grant an additional user access to the CloudForms server, create the user in Active Directory, then add that user to the **cloudforms** AD group. When that user attempts to log in to CloudForms, they will automatically inherit the correct permissions for the group.

4.3. CONFIGURING FEDERATED AUTHENTICATION WITH SAML

This procedure outlines how to manually configure an appliance to use SAML for federation authentication.

To enable external authentication using SAML, complete the following steps to configure your HTTP server, then your CloudForms appliance.

NOTE

The current SAML implementation only secures the CloudForms appliance's web user interface with SAML. The REST API and self service user interface do not currently support SAML.

SAML Requirements

The following is required in order to enable SAML authentication to the appliance:

- A CloudForms appliance
- A SAML identity provider (e.g. Red Hat Single Sign-On 7.0 or later)

4.3.1. Configuring Authentication with SAML and Red Hat Single Sign-On (RH-SSO)

While other SAML identity providers can be used with CloudForms, this example procedure covers using Red Hat Single Sign-On (RH-SSO) 7.0, which is implemented using the Apache HTTP server's **mod_auth_mellon** module.

NOTE

For more information about Red Hat Single Sign-On (RH-SSO), see the Red Hat Single Sign-On documentation.

4.3.1.1. Configuring the HTTP Server for SAML Authentication with Red Hat Single Sign-On (RH-SSO)

The Apache HTTP server first must be configured to work with SAML authentication. All SAML-related certificates and keys are accessed from **/etc/httpd/saml2/**.

1. Log into the CloudForms appliance as root using SSH, and create the **/etc/httpd/saml2/** directory:

```
# mkdir -p /etc/httpd/saml2
```

2. Copy the **httpd** remote user and SAML template configuration files to the appliance:

```
# TEMPLATE_DIR="/opt/rh/cfme-appliance/TEMPLATE"
# cp ${TEMPLATE_DIR}/etc/httpd/conf.d/manageiq-remote-user.conf /etc/httpd/conf.d/
# cp ${TEMPLATE_DIR}/etc/httpd/conf.d/manageiq-external-auth-saml.conf /etc/httpd/conf.d/
```

NOTE

The following are notable SAML configuration defaults in the **manageiq-external-auth-saml.conf** file:

- Identity Provider Files (i.e. Red Hat SSO)
    - Metadata File: **/etc/httpd/saml2/idp-metadata.xml**
- Service Provider Files (i.e. **mod_auth_mellon**)
    - Private Key File: **/etc/httpd/saml2/miqsp-key.key**
    - Certificate File: **/etc/httpd/saml2/miqsp-cert.cert**
    - Metadata File: **/etc/httpd/saml2/miqsp-metadata.xml**

Other **mod_auth_mellon** parameters, such as endpoints and protected URLs, must not be modified as the appliance expects them to be defined as such.

3. Generate the service provider files on the appliance using the Apache HTTP server's **mod_auth_mellon** command **mellon_create_metadata.sh**:

```
# cd /etc/httpd/saml2
# /usr/libexec/mod_auth_mellon/mellon_create_metadata.sh https://<miq-appliance>
https://<miq-appliance>/saml2
```

The **mellon_create_metadata.sh** script creates file names based on the appliance URL.

NOTE

If your appliance is behind a load balancer or uses a virtual IP address, use the hostname associated with the VIP. For example: **# /usr/libexec/mod_auth_mellon/mellon_create_metadata.sh https://my-haproxy-ka https://my-haproxy-ka/saml2**

4. Rename the files created by the **mellon_create_metadata.sh** script to match the expected file names from the **manageiq-external-auth-saml.conf** file:

```
# mv https_<miq-appliance>.key  miqsp-key.key
# mv https_<miq-appliance>.cert miqsp-cert.cert
# mv https_<miq-appliance>.xml  miqsp-metadata.xml
```

5. Now that the service provider's **metadata.xml** file has been generated, the service provider definition can be defined in the SAML identity provider. For Red Hat SSO, a realm can be created for one or more appliances with individual clients defined one per appliance, where the

client ID is specified as the URL of the appliance.
To add a client in the Red Hat SSO CloudForms realm:

a. Select and import the **miqsp-metadata.xml** file created for **mod_auth_mellon**.

b. Set the client ID as **https://<miq-appliance>**.

c. Set the client protocol as **saml**.

6. Update the client definition for the appliance in Red Hat SSO with the following:

Setting	Value
Name ID Format	username
Valid Redirect URIs	https://<miq-appliance>/saml2/postResponse
Assertion Consumer Service POST Binding URL	https://<miq-appliance>/saml2/postResponse
Logout Service Redirect Binding URL	https://<miq-appliance>/saml2/logout

7. Obtain the identity provider's **idp-metadata.xml** metadata file as follows:

```
# cd /etc/httpd/saml2
# curl -s -o idp-metadata.xml \
  http://<redhatSSO-server>:8080/auth/realms/<miq-realm>/protocol/saml/descriptor
```

8. In CloudForms, the following change is necessary to the **idp-metadata.xml** file for SAML logout to work between **mod_auth_mellon** and Red Hat SSO:

```
# vi idp-metadata.xml

  ...
  <SingleLogoutService
<   Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
---
>   Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect"
    Location=
  ...
```

9. Restart the HTTP server on the appliance:

```
# systemctl restart httpd
```

4.3.1.2. Configuring SAML Authentication in the CloudForms User Interface

After configuring the HTTP server for SAML, update the CloudForms appliance so that user interface can be accessed using SAML authentication.

1. Click ⚙ (**Configuration**).

2. Select your server in the **Settings** accordion.

3. Select the **Authentication** tab.

4. Select a **Session Timeout** to set the period of inactivity before a user is logged out of the console.

5. Set the mode to **External (httpd)**.

6. Check **Enable SAML**. This enables the SAML login button on the appliance login screen, then redirects to the SAML protected page for authentication, and supports the SAML logout process.

7. Check **Enable Single Sign-On**. With this option enabled, initial access to the appliance's user interface redirects to the SAML identity provider authentication screen. Logging out from the appliance returns the user to the appliance login screen, allowing them to log in as **admin** unless **Disable Local Login** is also checked.

8. Optional: Check **Disable Local Login** to disable the **admin** login to appliance and only allow SAML based authentication. Note that if there are issues with the identity provider or you require **admin** access to the appliance, you cannot log in through the appliance login screen until you re-enable local login as described in Section 4.3.3, "Re-enabling Local Login *(Optional)*".

9. Check **Get User Groups from External Authentication (httpd)**

10. Click **Save**.

IMPORTANT

Ensure the user's groups are created on the appliance and appropriate roles are assigned to those groups. See *SAML Assertions* in Section 4.3.2, "SAML Assertions" for more information on the parameters used by the CloudForms appliance.

For example, to configure user groups from your SAML identity provider to work with CloudForms:

1. In your SAML identity provider, specify your existing user groups in similar format to the following: **REMOTE_USER_GROUPS=Administrators;CloudAdministrators;Users**

2. On your CloudForms appliance, create the equivalent groups. See Creating a User Group in *General Configuration*.

3. On your CloudForms appliance, assign EVM roles to the groups. See Creating a Role in *General Configuration*.

Complete the above steps on each appliance in the settings menu, then navigate to **Configuration → Access Control**.

You can now log into your CloudForms appliance using your SAML credentials.

4.3.2. SAML Assertions

To authenticate to the CloudForms appliance using SAML, the following remote user parameters are looked at by the appliance upon a successful login and redirect from the identity provider. These parameters are used by the appliance to obtain group authentication information.

HTTP Environment	SAML Assertion
REMOTE_USER	username
REMOTE_USER_EMAIL	email
REMOTE_USER_FIRSTNAME	firstname
REMOTE_USER_LASTNAME	lastname
REMOTE_USER_FULLNAME	fullname
REMOTE_USER_GROUPS	groups

For Red Hat SSO, the above SAML assertions can be defined for the appliance client in Red Hat SSO as mappers.

Name	Category	Type	Property
username	AttributeStatement Mapper	User Property	username
email	AttributeStatement Mapper	User Property	email
firstname	AttributeStatement Mapper	User Property	firstName
lastname	AttributeStatement Mapper	User Property	lastName
fullname	AttributeStatement Mapper	User Attribute	fullName
groups	Group Mapper	Group List	groups

IMPORTANT

The **fullName** attribute was not available in the default database as of this writing and was added as a user attribute.

4.3.3. Re-enabling Local Login *(Optional)*

If you disabled local login in the CloudForms user interface but need the ability to log in as **admin**, you can re-enable local login using one of the following methods:

Re-enabling Local Login from the Appliance User Interface

This method requires the identity provider to be available, and the ability to log in as a user with enough administrative privileges to update CloudForms authentication settings.

1. Log in to the appliance user interface as the administrative user.
2. From the settings menu, select **Configuration → Authentication**.

3. Uncheck **Disable Local Login**.
4. Click **Save**.

Re-enabling Local Login from the Appliance Console:

1. Use SSH to log into the appliance as **root**.
2. Run the **appliance_console** command.
3. Select **Update External Authentication Options**
4. Select **Enable Local Login**.
5. Apply the updates.

Alternatively, log into the appliance as root using SSH, and run the following command:

```
# appliance_console_cli --extauth-opts local_login_disabled=false
```

4.4. TROUBLESHOOTING EXTERNAL AUTHENTICATION CONFIGURATION

The following are common errors that you may encounter when integrating with IPA:

```
Error -1 : Kerberos authentication failed: kinit: Cannot contact any KDC for realm
```

Resolution: Verify on IPA server if you are able to log into the IPA server using same user.

```
Error-2: [----] W, [2017-09-28T10:05:29.157980 #28902:fa8bc4] WARN -- Failure:
MIQ(Authenticator.authenticate) userid: [jdoe] - Authentication failed for userid jdoe: Authentication
token is no longer valid; new one required
```

Resolution: The password has expired. Go to IPA/LDAP server and update your password using **kinit <username>**.

```
Error-3: [----] W, [2017-09-28T10:13:10.083614 #28902:fa8bc4] WARN -- Failure:
MIQ(Authenticator.block in authorize) userid: [jdoe] - Authentication failed for userid jdoe, unable to
match user's group membership to an EVM role
```

Resolution: You should add the same group to the CloudForms group. Make sure you select **Look up External Authentication Groups** and enter the group name and user to look up.

4.4.1. Additional Troubleshooting Tips

- Make an IPA/LDAP host entry in **/etc/hosts** file on the CloudForms appliance.
- Make sure NTP synchronization exists between the CloudForms appliance and the IPA server.
- Check **/var/log/krb5kdc.log** on the IPA server end and **/var/www/miq/vmdb/log/audit.log** on the CloudForms appliance end for any other exception.