



Red Hat Ansible Automation Platform 2.4

Getting started with automation hub

Configure Red Hat automation hub as your default server for Ansible collections content

Red Hat Ansible Automation Platform 2.4 Getting started with automation hub

Configure Red Hat automation hub as your default server for Ansible collections content

Legal Notice

Copyright © 2024 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 walks you through the initial steps required to use Red Hat automation hub as the default source for certified Ansible collections content.

Table of Contents

PREFACE	3
PROVIDING FEEDBACK ON RED HAT DOCUMENTATION	4
CHAPTER 1. DISTINCTIONS BETWEEN AUTOMATION HUB AND PRIVATE AUTOMATION HUB	5
CHAPTER 2. CREATING THE API TOKEN IN AUTOMATION HUB	6
2.1. CREATING THE API TOKEN IN AUTOMATION HUB	6
2.2. CREATING THE API TOKEN IN PRIVATE AUTOMATION HUB	6
2.3. KEEPING YOUR OFFLINE TOKEN ACTIVE	7
CHAPTER 3. CONFIGURING RED HAT AUTOMATION HUB AS THE PRIMARY SOURCE FOR CONTENT ...	8
3.1. USING THE CLI TO CONFIGURE RED HAT AUTOMATION HUB AS THE PRIMARY CONTENT SOURCE	8
3.2. USING THE WEB CONSOLE TO CONFIGURE RED HAT AUTOMATION HUB AS THE PRIMARY CONTENT SOURCE	9
CHAPTER 4. CONFIGURING USER ACCESS FOR YOUR PRIVATE AUTOMATION HUB	12
4.1. IMPLEMENTING USER ACCESS	12
4.1.1. Default user access for private automation hub	12
4.1.2. Creating a new group in private automation hub	12
4.1.3. Assigning permissions to groups	12
4.1.4. Creating new users and giving them permissions	12
4.1.5. Creating a super user	13
4.1.6. Adding users to existing groups	13
4.1.7. Creating a new group for content curators	13
4.1.8. Automation hub permissions	14
4.1.9. Deleting a user from private automation hub	16
4.2. ENABLE VIEW-ONLY ACCESS FOR YOUR PRIVATE AUTOMATION HUB	16
CHAPTER 5. UPLOADING CONTENT TO RED HAT AUTOMATION HUB	18
5.1. UPLOADING A COLLECTION TO AUTOMATION HUB	18
5.2. DELETING A COLLECTION ON AUTOMATION HUB	19

PREFACE

Red Hat Ansible automation hub provides a place for Red Hat subscribers to quickly find and use content that is supported by Red Hat and our technology partners to deliver automation solutions for the most demanding environments.

The Ansible Galaxy client, **ansible-galaxy**, manages roles and collections from the command line. To ensure that the **ansible-galaxy** client uses certified, supported Ansible collections whenever possible, update your **ansible.cfg** file to use Red Hat automation hub as your primary source of Ansible collections.

PROVIDING FEEDBACK ON RED HAT DOCUMENTATION

If you have a suggestion to improve this documentation, or find an error, you can contact technical support at <https://access.redhat.com> to open a request.

CHAPTER 1. DISTINCTIONS BETWEEN AUTOMATION HUB AND PRIVATE AUTOMATION HUB

Red Hat Ansible Automation Platform uses an automation hub as a central location for automation content that you can download and integrate into your Ansible automation. Two types of hubs are available:

Automation hub

Hosted by Red Hat on the [Red Hat Hybrid Cloud Console](#), it contains only Red Hat supported or certified content.

Private automation hub

This is a self-hosted content management system. You can use it to access and manage all types of Ansible content and choose which Ansible content collections and versions are made available to your automation consumers.

With a private automation hub, you can access these three types of content:

- Red Hat certified and supported content found in automation hub on the [Red Hat Hybrid Cloud Console](#).
- Community content from Ansible Galaxy.
- Private content created and curated by an organization and shared locally.

CHAPTER 2. CREATING THE API TOKEN IN AUTOMATION HUB

Before you can interact with automation hub by uploading or downloading collections, you must create an API token. The automation hub API token authenticates your **ansible-galaxy** client to the Red Hat automation hub server.

Your method for creating the API token differs according to the type of automation hub that you are using:

- Automation hub uses Offline token management. See [Creating the API token in automation hub](#).
- Private automation hub uses API token management. See [Creating the API token in private automation hub](#).

2.1. CREATING THE API TOKEN IN AUTOMATION HUB

In automation hub, you can create an API token by using **Token management**. The API token is a secret token used to protect your content.

Procedure

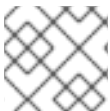
1. Navigate to [Ansible Automation Platform on the Red Hat Hybrid Cloud Console](#) .
2. From the navigation panel, select **Automation Hub** → **Connect to Hub**.
3. Under **Offline token**, click **Load Token**.
4. Click the **Copy to clipboard** icon to copy the API token.
5. Paste the API token into a file and store in a secure location.



IMPORTANT

The API token is a secret token used to protect your content. Store your API token in a secure location.

The API token is now available for configuring automation hub as your default collections server or for uploading collections by using the **ansible-galaxy** command line tool.



NOTE

The API token does not expire.

2.2. CREATING THE API TOKEN IN PRIVATE AUTOMATION HUB

In private automation hub, you can create an API token using API token management. The API token is a secret token used to protect your content.

Prerequisites

- Valid subscription credentials for Red Hat Ansible Automation Platform.

Procedure

1. Log in to your private automation hub.
2. From the navigation panel, select **Collections** → **API token**.
3. Click **Load Token**.
4. To copy the API token, click the **Copy to clipboard** icon.
5. Paste the API token into a file and store in a secure location.



IMPORTANT

The API token is a secret token used to protect your content. Store your API token in a secure location.

The API token is now available for configuring automation hub as your default collections server or uploading collections using the **ansible-galaxy** command line tool.



NOTE

The API token does not expire.

2.3. KEEPING YOUR OFFLINE TOKEN ACTIVE

Offline tokens expire after 30 days of inactivity. You can keep your offline token from expiring by periodically refreshing your offline token.

Keeping an online token active is useful when an application performs an action on behalf of the user; for example, this allows the application to perform a routine data backup when the user is offline.



NOTE

If your offline token expires, you must request a new one.

Procedure

- Run the following command to prevent your token from expiring:

```
curl https://sso.redhat.com/auth/realms/redhat-external/protocol/openid-connect/token -d grant_type=refresh_token -d client_id="cloud-services" -d refresh_token="{{ user_token }}" --fail --silent --show-error --output /dev/null
```

CHAPTER 3. CONFIGURING RED HAT AUTOMATION HUB AS THE PRIMARY SOURCE FOR CONTENT

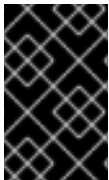
To access Ansible Certified Content Collections, configure Red Hat automation hub as your primary source of content. You can configure automation hub in the command-line interface (CLI) or the web console.

3.1. USING THE CLI TO CONFIGURE RED HAT AUTOMATION HUB AS THE PRIMARY CONTENT SOURCE

To configure automation hub, you must update the **ansible.cfg** configuration file. By default, the **ansible.cfg** configuration file is located in the **/etc/ansible/** directory. With automation hub, you have access to certified, supported collections.

Prerequisites

- You have obtained the API token for the automation hub server. See [Creating the Red Hat automation hub API token](#) for more information.



IMPORTANT

Creating a new token revokes any previous tokens generated for automation hub. Update any automation controller or scripts created with the previous token to include the new token.

Procedure

1. Open the **ansible.cfg** file.
2. Add the **server_list** option under the **[galaxy]** section and include one or more server names.
3. Create a new section for each server name:

```
[galaxy_server.<server_name>_]
```

4. Set the **url** option for each server name:

```
https://<server_fully_qualified_domain_name>/api/galaxy/
```

5. Optional: Set the **auth_url** option. The community Ansible Galaxy does not require an **auth_url**.
6. Set the API token for the automation hub server.

Example

The following **ansible.cfg** configuration file example shows how to configure multiple servers in prioritized order. Automation hub is configured as your primary source and an Ansible Galaxy server as a secondary source:

ansible.cfg

```
[galaxy]
```

```

server_list = automation_hub, my_org_hub

[galaxy_server.automation_hub]
url=https://console.redhat.com/api/automation-hub/content/published/ 1
auth_url=https://sso.redhat.com/auth/realms/redhat-external/protocol/openid-connect/token

token=my_ah_token

[galaxy_server.my_org_hub]
url=https://automation.my_org/api/galaxy/content/rh-certified/ 2
username=my_user
password=my_pass

```

- 1 Include a trailing slash / after the server URL.
- 2 Include the **/api/galaxy/content/rh-certified/** subdirectory in the automation hub server URL. You can replace **rh-certified** with **community** to reference the community repository if you prefer.



NOTE

To prevent a 301 redirect, all API URLs must end with a trailing slash /.

You have now configured automation hub as your primary server. You can begin to download and install supported collections.

Additional resources

For more information about server list configuration options and using Ansible Galaxy as an Ansible content source, see the [Ansible Galaxy User Guide](#).

3.2. USING THE WEB CONSOLE TO CONFIGURE RED HAT AUTOMATION HUB AS THE PRIMARY CONTENT SOURCE

To configure automation hub, you must create a credential and add it to the Organization's Galaxy Credentials field. With automation hub, you have access to certified, supported collections.

Prerequisites

- You have obtained the API token for the automation hub server. See [Creating the Red Hat automation hub API token](#) for more information.



IMPORTANT

Creating a new token revokes any previous tokens generated for automation hub. Update any automation controller or scripts created with the previous token.

Procedure

1. Navigate to your automation controller.
2. Create a new credential.

- a. Navigate to **Resources** → **Credentials**.
 - b. Click **Add**.
 - c. Enter the name for your new credential in the **Name** field.
 - d. Optional: Enter a description and enter or select the name of the organization with which the credential is associated.
 - e. Under **Organization**, select the organization that you want to use the credential for.
 - f. Select **Ansible Galaxy/Automation Hub API Token** as the credential type.
 - g. Under **Type Details**, enter the **Galaxy Server URL**, **Authentication Server URL**, and **API Token** previously created.
 - h. Click **Save**.
3. Select the credential that you created from the Organization's **Galaxy Credentials** field.
 - a. Navigate to **Access** → **Organizations**.
 - b. Select the organization where you want to add your Galaxy credentials.
 - c. Click **Edit**.
 - d. Under Galaxy Credentials, click the **Search** icon.
 - e. Select the credential that you created for automation hub, and place it at the beginning of the list.
 - f. Optional: If you have a secondary source of content, such as Ansible Galaxy, place this credential after the credential that you created for automation hub.
 - g. Click **Select**.
 - h. Click **Save**.

Verification

To validate the credential, update an existing source control management (SCM)-based project by selecting the project and clicking the **Sync Project** icon.

1. Navigate to your project repository.
2. Select a project that uses a **collections/requirements.yml** file.
3. Update the project by clicking the **Sync Project** icon.

If the **Status** of the project is **Successful**, then the credential is valid.

You have now configured automation hub as your primary server. You can begin to download and install supported collections.

Additional resources

- For more information about server list configuration options and using Ansible Galaxy as an Ansible content source, see the [Ansible Galaxy User Guide](#) .
- For more information about creating and using credentials, see the [Credentials](#) section of Automation Controller User Guide v4.2.1.

CHAPTER 4. CONFIGURING USER ACCESS FOR YOUR PRIVATE AUTOMATION HUB

You can manage user access to content and features in automation hub by creating groups of users that have specific permissions.

4.1. IMPLEMENTING USER ACCESS

User access is based on managing permissions to system objects (users, groups, namespaces) rather than by assigning permissions individually to specific users.

You assign permissions to the groups that you create. You can then assign users to these groups. This means that each user in a group has the permissions assigned to that group.

Groups created in private automation hub can range from system administrators responsible for governing internal collections, configuring user access, and repository management to groups with access to organize and upload internally developed content to the private automation hub.

Additional resources

- See [Automation Hub permissions](#) for information on system permissions.

4.1.1. Default user access for private automation hub

When you install automation hub, the system automatically creates the default **admin** user in the **Admin** group. The **Admin** group is assigned all permissions in the system.

The following sections describe the workflows associated with organizing your users who will access private automation hub and providing them with required permissions to reach their goals. See the permissions reference table for a full list and description of all permissions available.

4.1.2. Creating a new group in private automation hub

You can create and assign permissions to a group in private automation hub that enables users to access specified features in the system. By default, the **Admin** group in the automation hub has all permissions assigned and is available on initial login. Use the credentials created when installing private automation hub.

For more information, see [Creating a new group in private automation hub](#) in the Getting started with automation hub guide.

4.1.3. Assigning permissions to groups

By default, new groups do not have any assigned permissions. You can assign permissions to groups in private automation hub that enable users to access specific features in the system.

You can add permissions when first creating a group or edit an existing group to add or remove permissions

For more information, see [Assigning permissions to groups](#) in the Getting started with automation hub guide.

4.1.4. Creating new users and giving them permissions

After you create a user in private automation hub, you can give them permissions by adding them to groups. Each group that can access features in the system associated to the level of assigned permissions.

Prerequisites

- You have **user** permissions and can create users in private automation hub.

Procedure

1. Log in to your private automation hub.
2. From the navigation panel, select **User Access → Users**.
3. Click **Create user**.
4. Enter information in the field. **Username** and **Password** are required.
5. Optional: To assign the user to a group, click the **Groups** field and select from the list of groups.
6. Click **Save**.

The new user is now displayed in the list on the **Users** page.

4.1.5. Creating a super user

If you want to spread administration across your team, you can create a super user in private automation hub.

Prerequisites

- You must be a **Super user**.

Procedure

1. Log in to your private automation hub.
2. From the navigation panel, select **User Access → Users**.
3. Select the user that you want to make a super user. The **User details** for that user are displayed.
4. Under **User type**, select **Super User**.

The user now has **Super user** permissions.

4.1.6. Adding users to existing groups

You can add users to groups when you create a group. But, you can also manually add users to existing groups.

For more information, see [Adding users to existing groups](#) in the Getting started with automation hub guide.

4.1.7. Creating a new group for content curators

You can create a new group in private automation hub designed to support content curation in your organization. This group can contribute internally developed collections for publication in private automation hub.

To help content developers create a namespace and upload their internally developed collections to private automation hub, you must first create and edit a group and assign the required permissions.

Prerequisites

- You have administrative permissions in private automation hub and can create groups.

Procedure

1. Log in to your private automation hub.
2. From the navigation panel, select **User Access** → **Groups** and click **Create**.
3. Enter **Content Engineering** as a **Name** for the group in the modal and click **Create**. You have created the new group and the **Groups** page opens.
4. On the **Permissions** tab, click **Edit**.
5. Under **Namespaces**, add permissions for **Add Namespace**, **Upload to Namespace**, and **Change Namespace**.
6. Click **Save**.
The new group is created with the permissions that you assigned. You can then add users to the group.
7. Click the **Users** tab on the **Groups** page.
8. Click **Add**.
9. Select users and click **Add**.

4.1.8. Automation hub permissions

Permissions provide a defined set of actions each group can perform on a given object. Determine the required level of access for your groups based on the permissions described in this table.

Table 4.1. Permissions Reference Table

Object	Permission	Description
collection namespaces	Add namespace	Groups with these permissions can create, upload collections, and delete a namespace.
	Upload to namespace	
	Change namespace	
	Delete namespace	

Object	Permission	Description
collections	Modify Ansible repo content Delete collections	Groups with this permission can perform these actions: Move content between repositories by using the Approval feature. Certify or reject features to move content from the staging to published or rejected repositories. Delete collections.
users	View user Delete user Add user Change user	Groups with these permissions can manage user configuration and access in private automation hub.
groups	View group Delete group Add group Change group	Groups with these permissions can manage group configuration and access in private automation hub.
collection remotes	Change collection remote View collection remote	Groups with these permissions can configure a remote repository by navigating to Collection → Repositories .
containers	Change container namespace permissions Change containers Change image tags Create new containers Push to existing containers Delete container repository	Groups with these permissions can manage container repositories in private automation hub.
remote registries	Add remote registry Change remote registry Delete remote registry	Groups with these permissions can add, change, or delete remote registries added to private automation hub.

Object	Permission	Description
task management	Change task	Groups with these permissions can manage tasks added to Task Management in private automation hub.
	Delete task	
	View all tasks	


4.1.9. Deleting a user from private automation hub

When you delete a user account, the name and email of the user are permanently removed from private automation hub.

Prerequisites

- You have **user** permissions in private automation hub.

Procedure

- Log in to private automation hub.
- From the navigation panel, select **User Access** → **Users**.
- Click **Users** to display a list of the current users.
- Click the **More Actions** icon  icon beside the user that you want to remove, then click **Delete**.
- Click **Delete** in the warning message to permanently delete the user.

4.2. ENABLE VIEW-ONLY ACCESS FOR YOUR PRIVATE AUTOMATION HUB

By enabling view-only access, you can grant access for users to view collections or namespaces on your private automation hub without requiring them to log in. View-only access allows you to share content with unauthorized users while restricting their ability to view or download source code. They will not have permissions to edit anything on your private automation hub.

To enable view-only access for your private automation hub, you must edit the inventory file on your Red Hat Ansible Automation Platform installer.

- If you are installing a new instance of Ansible Automation Platform, add the **automationhub_enable_unauthenticated_collection_access** and **automationhub_enable_unauthenticated_collection_download** parameters to your **inventory** file along with your other installation configurations:
- If you are updating an existing Ansible Automation Platform installation to include view-only access, add the **automationhub_enable_unauthenticated_collection_access** and **automationhub_enable_unauthenticated_collection_download** parameters to your **inventory** file and then run the **setup.sh** script to apply the updates:

Procedure

- Navigate to the installer.

Bundled installer

```
$ cd ansible-automation-platform-setup-bundle-<latest-version>
```

Online installer

```
$ cd ansible-automation-platform-setup-<latest-version>
```

2. Open the **inventory** file with a text editor.
3. Add the **automationhub_enable_unauthenticated_collection_access** and **automationhub_enable_unauthenticated_collection_download** parameters to the inventory file and set both to **True**, following the example below:

```
[all:vars]
```

```
automationhub_enable_unauthenticated_collection_access = True 1  
automationhub_enable_unauthenticated_collection_download = True 2
```

- 1** Allows unauthorized users to view collections
- 2** Allows unauthorized users to download collections

4. Run the **setup.sh** script. The installer enables view-only access to your private automation hub.

Verification

After the installation is complete, verify that you have view-only access on your private automation hub by attempting to view content on your private automation hub without logging in.

1. Navigate to your private automation hub.
2. On the login screen, click **View only mode**.

Verify that you are able to view content on your automation hub, such as namespaces or collections, without having to log in.

CHAPTER 5. UPLOADING CONTENT TO RED HAT AUTOMATION HUB

Automation hub distributes certified, supported collections from partners to customers. Each collection includes content such as modules, roles, plugins and documentation. The first time you upload a collection to automation hub, our Partner Engineering team reviews it for certification.

You can manage your collections by uploading or deleting collections using the automation hub user interface or the **ansible-galaxy** client.

5.1. UPLOADING A COLLECTION TO AUTOMATION HUB

If you want to share a collection that you have created with the rest of the Ansible community, you can upload it to automation hub. When you upload a collection to automation hub, our Partner Engineering team reviews it for certification.

You can upload the collection by using either the automation hub user interface or the **ansible-galaxy** client.

Prerequisites

- You have configured the **ansible-galaxy** client for Red Hat Automation Hub.
- You have at least one namespace.
- You have run all content through **ansible-test sanity**.
- You are a Red Hat Connect Partner. Learn more at [Red Hat Partner Connect](#).

Procedure

Using the automation hub user interface:

1. From the navigation panel, select **Collections** → **Namespaces**.
2. On the **My namespaces** tab, locate the namespace to which you want to upload a collection.
3. Click **View collections** and click **Upload collection**.
4. In the **New collection** modal, click **Select file**. Locate the file on your system.
5. Click **Upload**.

Using the **ansible-galaxy** client:

- Enter the following command:

```
ansible-galaxy collection publish path/to/my_namespace-my_collection-1.0.0.tar.gz --api-key=SECRET
```

Next steps

- After you upload your collections, they enter the partner certification process. Our Partner Engineering team will contact you with the certification status of your collection.


5.2. DELETING A COLLECTION ON AUTOMATION HUB

You can further manage your collections by deleting unwanted collections, if the collection is not dependent on other collections. The **Dependencies** tab on a collection displays a list of other collections that use the current collection.

Prerequisites

- The collection being deleted does not have dependencies with other collections.
- You have **Delete Collections** permissions.

Procedure

1. Log in to your private automation hub.
2. From the navigation panel, select **Collections** → **Collections**.
3. Before deleting the collection, check to see if it has collections that are dependent on it:
 - Click the **Dependencies** tab for that collection. If it is blank, you will be able to delete the collection. If the **Dependencies** tab is not blank, you must delete these dependencies before you can delete the collection.
4. Click the collection to delete.
5. Click the **More Actions** icon , and then select an option:
 - a. **Delete entire collection** to delete all versions in this collection.
 - b. **Delete version [number]** to delete the current version of this collection. You can change versions by using the **Version** drop-down menu.



NOTE

If the selected collection has any dependencies with other collections, these actions are disabled until you delete those dependencies. Click the **Dependencies** tab to see a list of dependencies to delete.

6. When the confirmation window opens, verify that the collection or version number is correct, and then select **Delete**.