

14. Projects

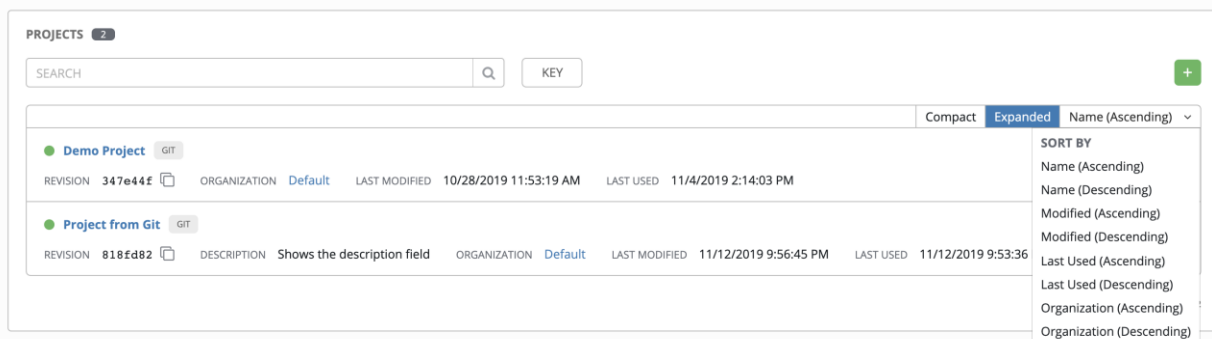
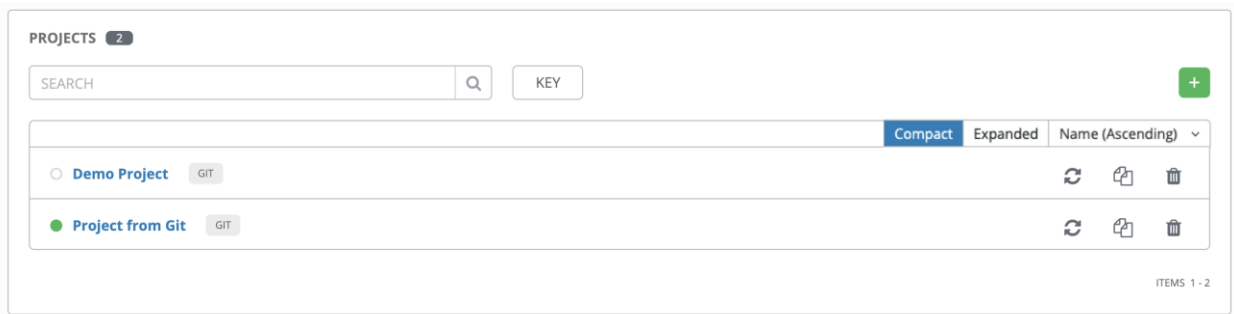
A [Project](#) is a logical collection of Ansible playbooks, represented in Tower.




You can manage playbooks and playbook directories by either placing them manually under the Project Base Path on your Tower server, or by placing your playbooks into a source code management (SCM) system supported by Tower, including Git, Subversion, Mercurial, and Red Hat Insights. To create a Red Hat Insights project, refer to [Setting up an Insights Project](#).

Note

By default, the Project Base Path is `/var/lib/awx/projects`, but this may have been modified by the Tower administrator. It is configured in `/etc/tower/conf.d/custom.py`. Use caution when editing this file, as incorrect settings can disable your installation.

This menu displays the list of the projects that are currently available. The default view is collapsed (**Compact**) with project name and its status, but you can expand to see more information. You can sort this list by various criteria, and perform a search to filter the projects of interest.



For each project listed, you can get the latest SCM revision () , copy the project attributes () , or delete () the project, using the respective icons next to each project. Starting with

Ansible Tower 3.7, projects are allowed to be updated while a related job is running. In cases where you have a big project (around 10 GB), disk space on /tmp may be an issue.

Status indicates the state of the project and may be one of the following (note that you can also filter your view by specific status types):

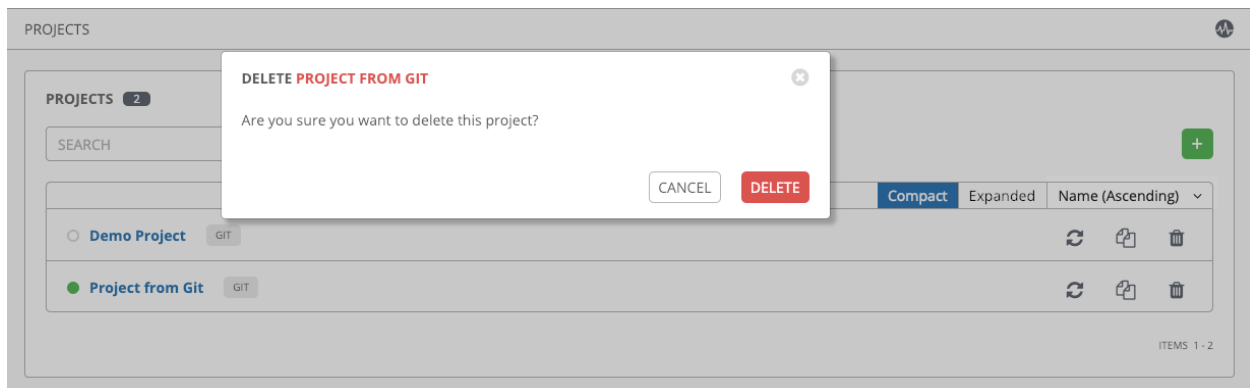
- **Pending** - The source control update has been created, but not queued or started yet. Any job (not just source control updates) will stay in pending until it's actually ready to be run by the system. Reasons for it not being ready because it has dependencies that are currently running so it has to wait until they are done, or there is not enough capacity to run in the locations it is configured to.
- **Waiting** - The source control update is in the queue waiting to be executed.
- **Running** - The source control update is currently in progress.
- **Successful** - The last source control update for this project succeeded.
- **Failed** - The last source control update for this project failed.
- **Error** - The last source control update job failed to run at all. (To be deprecated.)
- **Canceled** - The last source control update for the project was canceled.
- **Never updated** - The project is configured for source control, but has never been updated.
- **OK** - The project is not configured for source control, and is correctly in place. (To be deprecated.)
- **Missing** - Projects are absent from the project base path of /var/lib/awx/projects (applicable for manual or source control managed projects).

Note

Projects of credential type Manual cannot update or schedule source control-based actions without being reconfigured as an SCM type credential.


Note

If deleting items that are used by other work items, a message opens listing the items are affected by the deletion and prompts you to confirm the deletion. Some screens will contain items that are invalid or previously deleted, so they will fail to run. Below is an example of such a message:



14.1. Add a new project

To create a new project:

1. Click the  button, which launches the **Create Project** window.

2. Enter the appropriate details into the following required fields:
 - **Name**
 - **Description** (optional)
 - **Organization** - A project must have at least one organization. Pick one organization now to create the project, and then after the project is created you can add additional organizations.
 - **Ansible Environment** (optional) - Select from the drop-down menu list a custom virtual environment on which to run this project. This field is only present if custom environments were previously created. See [Using virtualenv with Ansible Tower](#) in the *Ansible Tower Upgrade and Migration Guide*.
 - **SCM Type** - Select from the drop-down menu list an SCM type associated with this project. The options in the subsequent section become available depend on the type you choose. Refer to [Manage playbooks manually](#) or [Manage playbooks using source control](#) in the subsequent sections for more detail.

Note

If adding a manual project, each project path inside of the project root folder can only be assigned to one project. If you receive the following message, ensure that you have not already assigned the project path to an existing project:

All of the project paths have been assigned to existing projects, or there are no directories found in the base path. You will need to add a project path before creating a new project.

3. Click **Save** when done.

14.1.1. Manage playbooks manually

- Create one or more directories to store playbooks under the Project Base Path (for example, `/var/lib/awx/projects/`)
- Create or copy playbook files into the playbook directory.
- Ensure that the playbook directory and files are owned by the same UNIX user and group that the Tower service runs as.
- Ensure that the permissions are appropriate for the playbook directories and files.

If you have trouble adding a project path, check the permissions and SELinux context settings for the project directory and files.

Warning

If you have not added any Ansible playbook directories to the base project path, you will receive the following message from Tower:

NEW PROJECT

DETAILS PERMISSIONS JOB TEMPLATES SCHEDULES

* NAME: Example DESCRIPTION: Ansible example playbook * ORGANIZATION: Honey Dog, Inc.

ANSIBLE ENVIRONMENT: Select Ansible Environment * SCM TYPE: Manual

WARNING: There are no available playbook directories in `/var/lib/awx/projects`. Either that directory is empty, or all of the contents are already assigned to other projects. Create a new directory there and make sure the playbook files can be read by the "awx" system user, or have Tower directly retrieve your playbooks from source control using the SCM Type option above.

PROJECT BASE PATH: `/var/lib/awx/projects`

CANCEL SAVE

Correct this issue by creating the appropriate playbook directories and checking out playbooks from your SCM or otherwise copying playbooks into the appropriate playbook directories.

14.1.2. Manage playbooks using source control

- [SCM Types - Git, Mercurial and Subversion](#)
- [SCM Type - Red Hat Insights](#)
- [SCM Type - Remote Archive](#)


14.1.2.1. SCM Types - Git, Mercurial and Subversion

To configure playbooks to use source control, in the Project **Details** tab:

1. Select the appropriate option (Git, Subversion, or Mercurial) from the **SCM Type** drop-down menu list.

The screenshot shows the 'NEW PROJECT' form with the 'DETAILS' tab selected. The 'SCM TYPE' dropdown menu is highlighted with a red box and shows 'Git' as the selected option. Other fields include NAME (Example), DESCRIPTION (Ansible example playbook), ORGANIZATION (Honey Dog, Inc.), ANSIBLE ENVIRONMENT (Select Ansible Environment), SCM URL (https://github.com/ansible/tower-example), SCM CREDENTIAL, SCM BRANCH/TAG/COMMIT, SCM REFSPEC, and SCM UPDATE OPTIONS (CLEAN, DELETE ON UPDATE, UPDATE REVISION ON LAUNCH, ALLOW BRANCH OVERRIDE).

2. Enter the appropriate details into the following fields:

- **SCM URL** - See an example in the help  text.
- **SCM Branch/Tag/Commit** - Optionally enter the SCM branch, tags, commit hashes, arbitrary refs, or revision number (if applicable) from the source control (Git, Subversion, or Mercurial) to checkout. Some commit hashes and refs may not be available unless you also provide a custom refspec in the next field.
- **SCM Refspec** - This field is an option specific to git source control and only advanced users familiar and comfortable with git should specify which references to download from the remote repository. For more detail, see [job branch overriding](#).
- **SCM Credential** - If authentication is required, select the appropriate SCM credential

3. In the **SCM Update Options**, optionally select the launch behavior, if applicable.

- **Clean** - Removes any local modifications prior to performing an update.

- **Delete on Update** - Deletes the local repository in its entirety prior to performing an update. Depending on the size of the repository this may significantly increase the amount of time required to complete an update.
- **Update Revision on Launch** - Updates the revision of the project to the current revision in the remote source control, as well as cache the roles directory from [Galaxy](#) or [Collections](#). Tower ensures that the local revision matches and that the roles and collections are up-to-date with the last update. Also, to avoid job overflows if jobs are spawned faster than the project can sync, selecting this allows you to configure a Cache Timeout to cache prior project syncs for a certain number of seconds.
- **Allow Branch Override** - Allows a job template that uses this project to launch with a specified SCM branch or revision other than that of the project's. For more detail, see [job branch overriding](#).

SCM UPDATE OPTIONS

☐ CLEAN ?

☐ DELETE ON UPDATE ?

☐ UPDATE REVISION ON LAUNCH ?

☒ ALLOW BRANCH OVERRIDE ?

3. Click **Save** to save your project.

Tip

Using a GitHub link offers an easy way to use a playbook. To help get you started, use the helloworld.yml file available at: <https://github.com/ansible/tower-example.git>

This link offers a very similar playbook to the one created manually in the instructions found in the [Ansible Tower Quick Start Guide](#). Using it will not alter or harm your system in anyway.

14.1.2.2. SCM Type - Red Hat Insights

To configure playbooks to use Red Hat Insights, in the Project **Details** tab:

1. Select **Red Hat Insights** from the **SCM Type** drop-down menu list.
2. Red Hat Insights requires a credential for authentication. Select from the **Credential** field the appropriate credential for use with Insights.

3. In the **SCM Update Options**, optionally select the launch behavior, if applicable.
 - **Clean** - Removes any local modifications prior to performing an update.
 - **Delete on Update** - Deletes the local repository in its entirety prior to performing an update. Depending on the size of the repository this may significantly increase the amount of time required to complete an update.
 - **Update Revision on Launch** - Updates the revision of the project to the current revision in the remote source control, as well as cache the roles directory from [Galaxy](#) or [Collections](#). Tower ensures that the local revision matches and that the roles and collections are up-to-date with the last update. Also, to avoid job overflows if jobs are spawned faster than the project can sync, selecting this allows you to configure a Cache Timeout to cache prior project syncs for a certain number of seconds.

NEW PROJECT

DETAILS PERMISSIONS JOB TEMPLATES SCHEDULES

* NAME: Red Hat Insights Project DESCRIPTION: ORGANIZATION: Honey Dog, Inc.

* SCM TYPE: Red Hat Insights

SOURCE DETAILS

* CREDENTIAL: Insights Credential

SCM UPDATE OPTIONS

☐ CLEAN ☐ DELETE ON UPDATE ☐ UPDATE REVISION ON LAUNCH

CANCEL SAVE

3. Click **Save** to save your project.

14.1.2.3. SCM Type - Remote Archive

Playbooks using a remote archive allow projects to be provided based on a build process that produces a versioned artifact, or release, containing all the requirements for that project in a single archive.

To configure playbooks to use a remote archive, in the Project **Details** tab:

1. Select **Remote Archive** from the **SCM Type** drop-down menu list.
2. Enter the appropriate details into the following fields:
 - **SCM URL** - requires a URL to a remote archive, such as a *GitHub Release* or a build artifact stored in *Artifactory* and unpacks it into the project path for use

- **SCM Credential** - If authentication is required, select the appropriate SCM credential
3. In the **SCM Update Options**, optionally select the launch behavior, if applicable.
- **Clean** - Removes any local modifications prior to performing an update.
 - **Delete on Update** - Deletes the local repository in its entirety prior to performing an update. Depending on the size of the repository this may significantly increase the amount of time required to complete an update.
 - **Update Revision on Launch** - Not recommended, as this option updates the revision of the project to the current revision in the remote source control, as well as cache the roles directory from [Galaxy](#) or [Collections](#).
 - **Allow Branch Override** - Not recommended, as this option allows a job template that uses this project to launch with a specified SCM branch or revision other than that of the project's.

NEW PROJECT

DETAILS PERMISSIONS JOB TEMPLATES SCHEDULES

* NAME Remote Archived Project DESCRIPTION ORGANIZATION Honey Dog, Inc.

* SCM TYPE Remote Archive

SOURCE DETAILS

* SCM URL https://github.com/ansible/product-docs SCM CREDENTIAL

SCM UPDATE OPTIONS

☐ CLEAN ☐ DELETE ON UPDATE ☐ UPDATE REVISION ON LAUNCH ☐ ALLOW BRANCH OVERRIDE


CANCEL SAVE

Note

Since this SCM type is intended to support the concept of unchanging artifacts, it is advisable to disable Galaxy integration (for roles, at minimum).

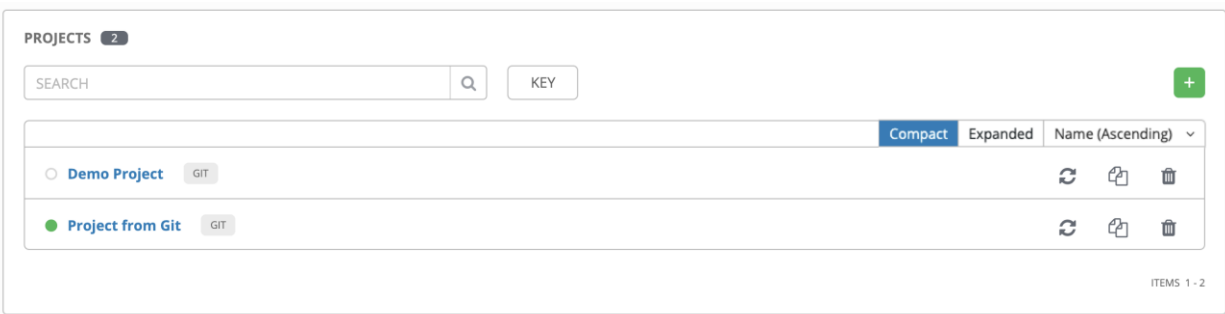
3. Click **Save** to save your project.

14.2. Updating projects from source control

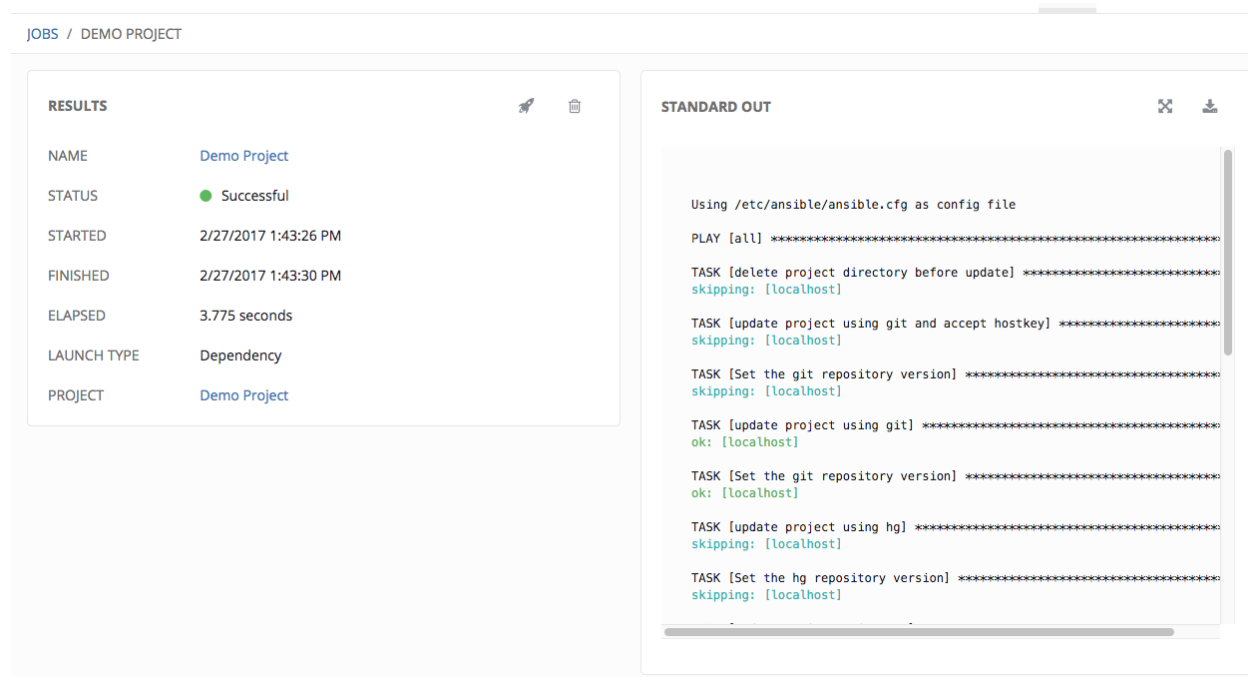
1. Update an existing SCM-based project by selecting the project and clicking the  button.

Note

Please note that immediately after adding a project setup to use source control, a “Sync” starts that fetches the project details from the configured source control.



2. Click on the dot under **Status** (far left, beside the name of the Project) to get further details about the update process.



14.3. Work with Permissions

The set of Permissions assigned to this project (role-based access controls) that provide the ability to read, modify, and administer projects, inventories, job templates, and other Tower elements are Privileges.

You can access the project permissions via the **Permissions** tab next to the **Details** tab. This screen displays a list of users that currently have permissions to this project. The list may be sorted and searched by **User**, **Role**, or **Team Role**.

PROJECTS / New project / PERMISSIONS

New project

DETAILS PERMISSIONS NOTIFICATIONS JOB TEMPLATES SCHEDULES


SEARCH Q KEY

| USER | ROLE | TEAM ROLES |
|-------|----------------------|------------|
| admin | SYSTEM ADMINISTRATOR | |
| jdoge | UPDATE | |

ITEMS 1 - 2

14.3.1. Add Permissions

The **Permissions** tab allows you to review, grant, edit, and remove associated permissions for users as well as team members. To assign permissions to a particular user for this resource:

1. Click the **Permissions** tab.
2. Click the  button to open the Add Users/Teams window.

DEMO EXAMPLE / PERMISSIONS

MPLE

PERMISSIONS

ES HOSTS

NAME

Database Servers

DEMO EXAMPLE

DEMO EXAMPLE | ADD USERS / TEAMS

1 Please select Users / Teams from the lists below.

USERS TEAMS

SEARCH Q KEY

| USERNAME | FIRST NAME | LAST NAME |
|-----------------------------------|------------|-----------|
| <input type="checkbox"/> althea | Althea | Bully |
| <input type="checkbox"/> austin78 | Austin | Texas |
| <input type="checkbox"/> gdoge | Gerry | Doge |
| <input type="checkbox"/> jdoge | Josie | Doge |
| <input type="checkbox"/> jgarcia | Jerry | Garcia |

< 1 2 > PAGE 1 OF 2

ITEMS 1 - 5 OF 6

CANCEL SAVE

3. Specify the users or teams that will have access then assign them specific roles:
 1. Click to select one or multiple check boxes beside the name(s) of the user(s) or team(s) to select them.

Note

You can select multiple users and teams at the same time by navigating between the **Users** and **Teams** tabs without saving.

After selections are made, the window expands to allow you to select a role from the drop-down menu list for each user or team you chose.

DEMO EXAMPLE | ADD USERS / TEAMS

1 Please select Users / Teams from the lists below.

USERS TEAMS

SEARCH [] Q KEY

| USERNAME ^ | FIRST NAME ^ | LAST NAME ^ |
|--|--------------|-------------|
| <input checked="" type="checkbox"/> althea | Althea | Bully |
| <input type="checkbox"/> austin78 | Austin | Texas |
| <input type="checkbox"/> gdodge | Gerry | Dodge |
| <input type="checkbox"/> jdodge | Josie | Dodge |
| <input type="checkbox"/> jgarcia | Jerry | Garcia |

< 1 2 > PAGE 1 OF 2 ITEMS 1 - 5 OF 6

2 Please assign roles to the selected users/teams KEY

Althea Bully USER

SELECT ROLES

- Admin
- Update
- Ad Hoc
- Use
- Read

SAVE

The example above shows options associated with inventories. Different resources have different options available:

- **Admin** allows read, run, and edit privileges (applies to all resources)
- **Use** allows use of a resource in a job template (applies all resources except job templates)
- **Update** allows updating of project via the SCM Update (applies to projects and inventories)

- **Ad Hoc** allows use of Ad Hoc commands (applies to inventories)
- **Execute** allows launching of a job template (applies to job templates)
- **Read** allows view-only access (applies to all resources)

Tip

Use the **Key** button in the roles selection pane to display a description of each of the roles. For more information, refer to the [Roles](#) section of this guide.

2. Select the role to apply to the selected user or team.

Note

You can assign roles to multiple users and teams by navigating between the **Users** and **Teams** tabs without saving.

DEMO EXAMPLE | ADD USERS / TEAMS

1 Please select Users / Teams from the lists below.

USERS TEAMS

SEARCH Q KEY

| USERNAME ^ | FIRST NAME ^ | LAST NAME ^ |
|--|--------------|-------------|
| <input checked="" type="checkbox"/> althea | Althea | Bully |
| <input type="checkbox"/> austin78 | Austin | Texas |
| <input type="checkbox"/> gdoge | Gerry | Doge |
| <input checked="" type="checkbox"/> jdoge | Josie | Doge |
| <input type="checkbox"/> jgarcia | Jerry | Garcia |

< 1 2 > PAGE 1 OF 2 ITEMS 1 - 5 OF 6

2 Please assign roles to the selected users/teams KEY

Althea Bully USER SELECT ROLES ×

Josie Doge USER SELECT ROLES ×

Production Operatio... TEAM SELECT ROLES ×

CANCEL SAVE

4. Review your role assignments for each user and team.

DEMO EXAMPLE / PERMISSIONS

MPLE

PERMISSIONS

ES HOSTS

NAME ^

Database Servers

DEMO EXAMPLE

Demo Inventory

King PLC

DEMO EXAMPLE | ADD USERS / TEAMS

1 Please select Users / Teams from the lists below.

USERS TEAMS

SEARCH Q KEY

| USERNAME ^ | FIRST NAME ^ | LAST NAME ^ |
|--|--------------|-------------|
| <input checked="" type="checkbox"/> althea | Althea | Bully |
| <input type="checkbox"/> austin78 | Austin | Texas |
| <input type="checkbox"/> gdoge | Gerry | Doge |
| <input checked="" type="checkbox"/> jdoge | Josie | Doge |
| <input type="checkbox"/> jgarcia | Jerry | Garcia |

< 1 2 > PAGE 1 OF 2 ITEMS 1 - 5 OF 6

2 Please assign roles to the selected users/teams

KEY

Althea Bully USER

Josie Doge USER

Production Operatio... TEAM

CANCEL SAVE

5. Click **Save** when done, and the Add Users/Teams window closes to display the updated roles assigned for each user and team.

| USER ^ | ROLE | TEAM ROLES |
|----------|--|---|
| admin | SYSTEM ADMINISTRATOR | |
| althea | <input type="text" value="AD HOC"/> <input type="text" value="SYSTEM AUDITOR"/> <input type="text" value="USE"/> | |
| jdoge | <input type="text" value="UPDATE"/> <input type="text" value="USE"/> | |
| mags3707 | SYSTEM ADMINISTRATOR | <input type="text" value="AD HOC"/> <input type="text" value="ADMIN"/> <input type="text" value="USE"/> |
| yser | SYSTEM AUDITOR | |

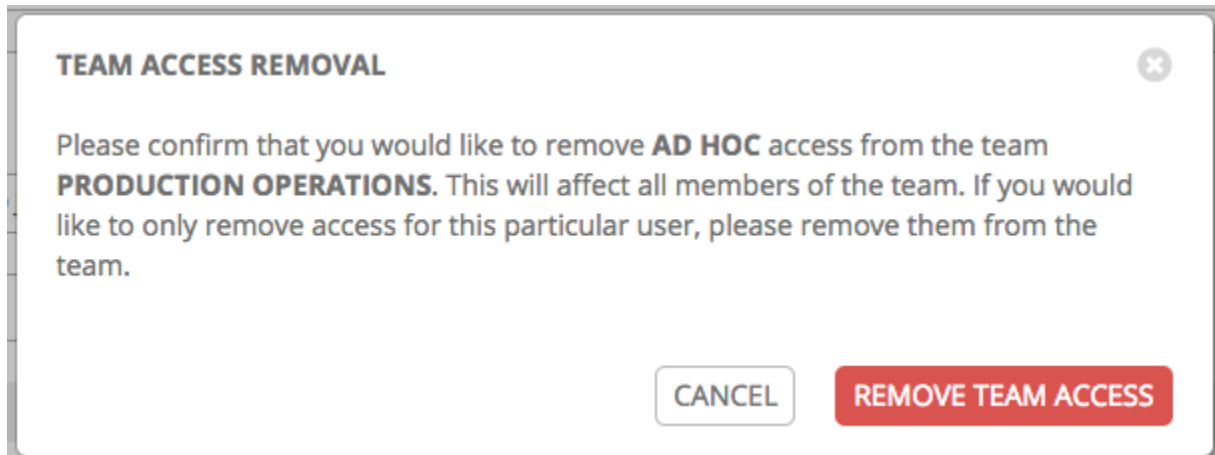
ITEMS 1 - 5

To remove Permissions for a particular user, click the Disassociate (x) button next to its resource.

| USER ^ | ROLE | TEAM ROLES |
|----------|--|---|
| admin | SYSTEM ADMINISTRATOR | |
| althea | <input type="text" value="AD HOC"/> <input type="text" value="SYSTEM AUDITOR"/> <input type="text" value="USE"/> | |
| jdoge | <input type="text" value="UPDATE"/> <input type="text" value="USE"/> | |
| mags3707 | SYSTEM ADMINISTRATOR | <input type="text" value="AD HOC"/> <input type="text" value="ADMIN"/> <input type="text" value="USE"/> |
| yser | SYSTEM AUDITOR | |

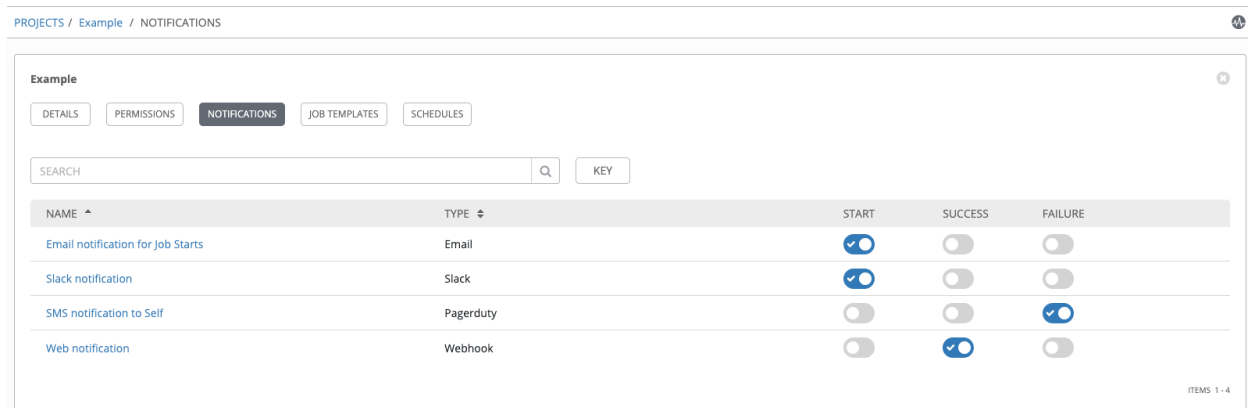
ITEMS 1 - 5

This launches a confirmation dialog, asking you to confirm the disassociation.



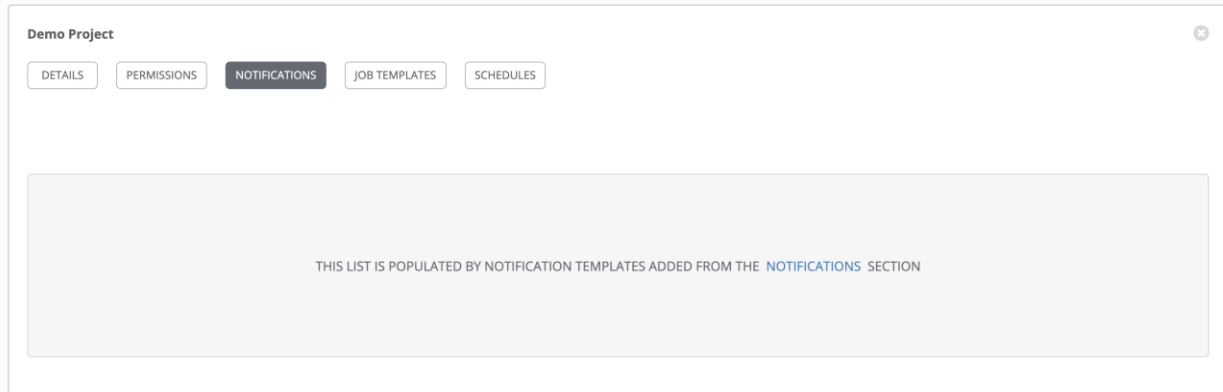
14.4. Work with Notifications

Clicking the **Notifications** tab allows you to review any notification integrations you have setup.



Use the toggles to enable or disable the notifications to use with your particular project. For more detail, see [Enable and Disable Notifications](#).

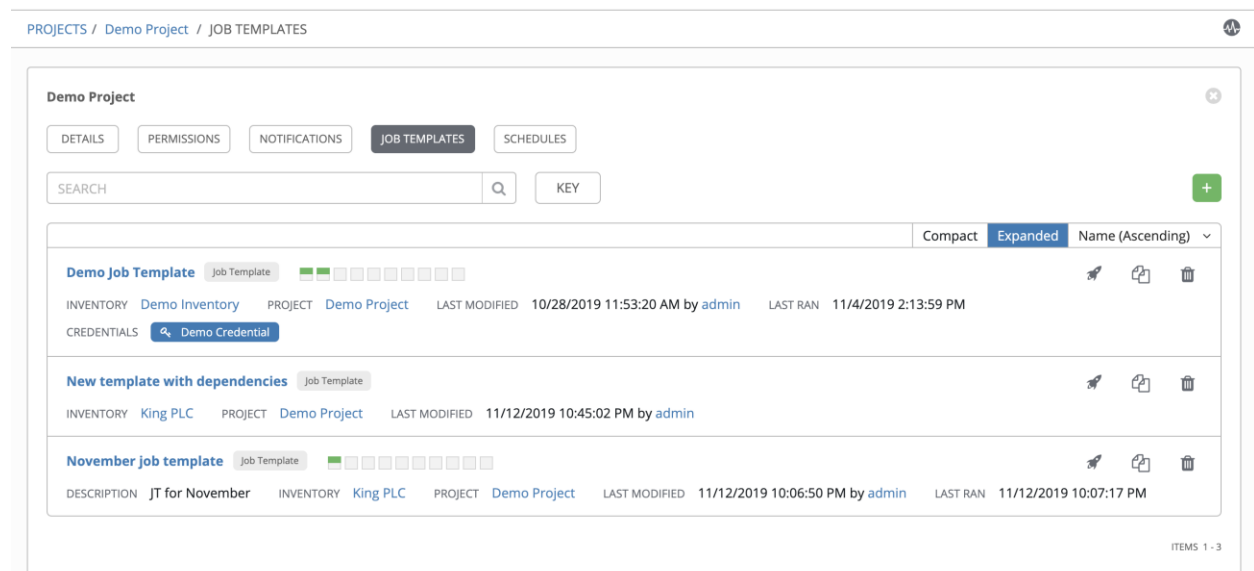
If no notifications have been set up, click the **NOTIFICATIONS** link from inside the gray box to create a new notification.



Refer to [Notification Types](#) for additional details on configuring various notification types.

14.5. Work with Job Templates

Clicking on **Job Templates** allows you to add and review any job templates or workflow templates associated with this project. Click **Expanded** to view details about each template, including the statuses of the jobs that ran using that template, and other useful information. You can sort this list by various criteria, and perform a search to filter the templates of interest.



From this view, you can also launch (), copy (), or delete () the template configuration. Note, the example above shows the expanded view.

14.6. Work with Schedules

Clicking on **Schedules** allows you to review any schedules set up for this project.

Demo Project

DETAILS

PERMISSIONS

NOTIFICATIONS

JOB TEMPLATES

SCHEDULES

SEARCH

KEY

+

| NAME ^ | FIRST RUN ↕ | NEXT RUN ↕ | FINAL RUN ↕ | ACTIONS |
|--|------------------------|------------------------|------------------------|---------|
| <input checked="" type="checkbox"/> Run Once | 11/13/2019 12:00:00 AM | 11/13/2019 12:00:00 AM | 11/13/2019 12:00:00 AM | |
| <input checked="" type="checkbox"/> Schedule 1 | 11/15/2019 12:00:00 AM | 11/15/2019 12:00:00 AM | | |
| <input checked="" type="checkbox"/> Schedule 2 | 12/1/2019 12:02:00 AM | 12/1/2019 12:02:00 AM | 3/1/2020 12:02:00 AM | |
| <input checked="" type="checkbox"/> Schedule 3 | 12/25/2019 12:03:00 AM | 12/25/2019 12:03:00 AM | 1/1/2020 12:03:00 AM | |

ITEMS 1 - 4

14.6.1. Schedule a Project

To schedule a project run, click the **Schedules** tab.

- If schedules are already set up; review, edit, or enable/disable your schedule preferences.
- If schedules have not been set up, refer to [Schedules](#) for more information.

14.7. Ansible Galaxy Support

At the end of a Project update, Tower searches for a file called requirements.yml in the roles directory, located at <project-top-level-directory>/roles/requirements.yml. If this file is found, the following command automatically runs:

```
ansible-galaxy role install -r roles/requirements.yml -p <project-specific cache location>/requirements_roles -vvv
```

This file allows you to reference Galaxy roles or roles within other repositories which can be checked out in conjunction with your own project. The addition of this Ansible Galaxy support eliminates the need to create git submodules for achieving this result. Given that SCM projects (along with roles/collections) are pulled into and executed from a private job environment, a <private job directory> specific to the project within /tmp is created by default. However, you can specify another **Job Execution Path** based on your environment in the Jobs Settings tab of the Configure Tower window:

JOBS

ANSIBLE MODULES ALLOWED FOR AD HOC JOBS REVERT

- * command
- * shell
- * yum
- * apt
- * apt_key
- * apt_repository
- * apt_rpm
- * service
- * group
- * user
- * mount
- * ping
- * selinux
- * setup
- * win_ping
- * win_service
- * win_updates
- * win_group
- * win_user

* JOB EXECUTION PATH REVERT

/tmp

* MAXIMUM SCHEDULED JOBS REVERT

10

PATHS TO EXPOSE TO ISOLATED JOBS REVERT

ANSIBLE CALLBACK PLUGINS REVERT

PATHS TO HIDE FROM ISOLATED JOBS REVERT

The cache directory is a subdirectory inside the global projects folder. The content may be copied from the cache location to <job private directory>/requirements_roles location.

By default, Ansible Tower has a system-wide setting that allows roles to be dynamically downloaded from the roles/requirements.yml file for SCM projects. You may turn off this

setting in the **Jobs** tab of the Settings () menu by switching the **Enable Role Download** toggle button to **OFF**.

JOBS

ANSIBLE MODULES ALLOWED FOR AD HOC JOBS REVERT

- * command
- * shell
- * yum
- * apt
- * apt_key
- * apt_repository
- * apt_rpm
- * service
- * group
- * user
- * mount
- * ping
- * selinux
- * setup
- * win_ping
- * win_service
- * win_updates
- * win_group
- * win_user

* JOB EXECUTION PATH REVERT

/tmp

* MAXIMUM SCHEDULED JOBS REVERT

10

PATHS TO EXPOSE TO ISOLATED JOBS REVERT

ANSIBLE CALLBACK PLUGINS REVERT

PATHS TO HIDE FROM ISOLATED JOBS REVERT

* ENABLE JOB ISOLATION REVERT

☒

DEFAULT PROJECT UPDATE TIMEOUT REVERT

0

* RUN PROJECT UPDATES WITH HIGHER VERBOSITY REVERT

☐

ENABLE COLLECTION(S) DOWNLOAD REVERT

☒

* ISOLATED STATUS CHECK INTERVAL REVERT

30

ENABLE DETAILED RESOURCE PROFILING ON ALL PLAYBOOK RUNS REVERT

☐

EXTRA ENVIRONMENT VARIABLES REVERT

1 {}

REVERT ALL TO DEFAULT

* ENABLE ROLE DOWNLOAD REVERT

☒

DEFAULT JOB TIMEOUT REVERT

0

PER-HOST ANSIBLE FACT CACHE TIMEOUT REVERT

0

IGNORE ANSIBLE GALAXY SSL CERTIFICATE VERIFICATION REVERT

☒

FOLLOW SYMLINKS REVERT

☐

ISOLATED HOST KEY CHECKING REVERT

☐

* ISOLATED LAUNCH TIMEOUT REVERT

600

ISOLATED CONNECTION TIMEOUT REVERT

10

CANCEL SAVE

Whenever a project sync runs, Tower determines if the project source and any roles from Galaxy and/or Collections are out of date with the project. Project updates will download the roles inside the update.

If jobs need to pick up a change made to an upstream role, updating the project will ensure this happens. A change to the role means that a new commit was pushed to the *provision-role* source control. To make this change take effect in a job, you do not need to push a new commit to the *playbooks* repo, but you **do need** to update the project, which downloads roles to a local cache. For instance, say you have two git repositories in source control. The first one is *playbooks* and the project in Tower points to this URL. The second one is *provision-role* and it is referenced by the roles/requirements.yml file inside of the *playbooks* git repo.

In short, jobs would download the most recent roles before every job run. Roles and collections are locally cached for performance reasons, and you will need to select **Update Revision on Launch** in the project SCM Update Options to ensure that the upstream role is re-downloaded before each job run:

SCM UPDATE OPTIONS

- ☐ CLEAN ?
- ☐ DELETE ON UPDATE ?
- ☒ UPDATE REVISION ON LAUNCH ?
- ☐ ALLOW BRANCH OVERRIDE ?

The update happens much earlier in the process than the sync, so this surfaces errors and details faster and in a more logic place.

The screenshot shows the Ansible Tower interface. On the left, a sidebar displays job details for a job named 'feature test' (ID 3320), which is 'Successful'. The main panel shows the job's execution progress. A modal window titled 'localhost' is open, displaying the output of the 'fetch galaxy roles' task. The output shows that roles 'timezone' and 'debug-variable' were successfully downloaded from their respective sources. A red arrow points from the 'fetch galaxy roles' task in the job's task list to the modal window.

```
1 - downloading role 'timezone', owned by yatesr
2 - downloading role from https://github.com/yatesr/ansible-timezone/archive/1.2.0.tar.gz
3 - extracting yatesr.timezone to /var/lib/awx/projects/_awx_cache/_3241__feature_test/stage/requirements_roles/yatesr.timezone
4 - yatesr.timezone (1.2.0) was installed successfully
5 - downloading role 'debug-variable', owned by grog
6 - downloading role from https://github.com/GROG/ansible-role-debug-variable/archive/v1.2.0.tar.gz
```

For more information and examples on the syntax of the requirements.yml file, refer to the [role requirements section](#) in the Ansible documentation.

If there are any directories that should specifically be exposed, you can specify those in the Configure Tower screen in the **Paths to Expose to Isolated Jobs** or by updating the following entry in the settings file:

```
AWX_PROOT_SHOW_PATHS = ['/list/of/', '/paths']
```

Note

The primary file you may want to add to AWX_PROOT_SHOW_PATHS is /var/lib/awx/.ssh, if your playbooks need to use keys or settings defined there.

If you made changes in the settings file, be sure to restart services with the ansible-tower-service restart command after your changes have been saved.

In the Tower User Interface, you can configure these settings in the Jobs settings window.

SETTINGS / JOBS

JOBS

ANSIBLE MODULES ALLOWED FOR AD HOC JOBS REVERT

- command
- shell
- yum
- apt
- apt_key
- apt_repository
- apt_rpm
- service
- group
- user
- mount
- ping
- selinux
- setup
- win_ping
- win_service
- win_updates
- win_group
- win_user

PATHS TO EXPOSE TO ISOLATED JOBS REVERT

JOB EXECUTION PATH REVERT

/tmp

MAXIMUM SCHEDULED JOBS REVERT

10

ANSIBLE CALLBACK PLUGINS REVERT

PATHS TO HIDE FROM ISOLATED JOBS REVERT

Note


The **Primary Galaxy Server Username** and **Primary Galaxy Server Password** fields are no longer configurable in Ansible Tower 3.8. We recommend using tokens to access Galaxy or Automation Hub instead.

14.8. Collections Support

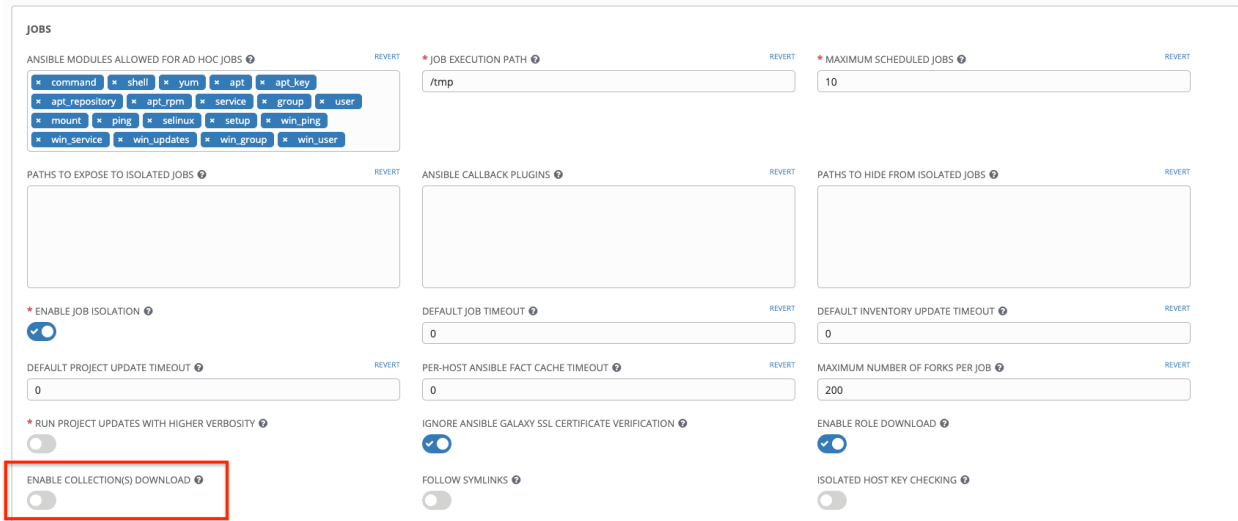
Tower supports project-specific Ansible collections in job runs. If you specify a collections requirements file in the SCM at collections/requirements.yml, Tower will install collections in that file in the implicit project sync before a job run. To specify a collections requirement:

```
ansible-galaxy collection install -r requirements.yml -p <job tmp location>
```

By default, Ansible Tower has a system-wide setting that allows collections to be dynamically downloaded from the collections/requirements.yml file for SCM projects. You

may turn off this setting in the **Jobs** tab of the Settings () menu by switching the **Enable Collections Download** toggle button to **OFF**.

SETTINGS / JOBS



The screenshot shows the 'JOBS' settings page in Ansible Tower. The 'ENABLE COLLECTION(S) DOWNLOAD' toggle is located in the bottom left section and is currently turned off. Other settings visible include 'ANSIBLE MODULES ALLOWED FOR AD HOC JOBS', 'JOB EXECUTION PATH', 'MAXIMUM SCHEDULED JOBS', 'PATHS TO EXPOSE TO ISOLATED JOBS', 'ANSIBLE CALLBACK PLUGINS', 'PATHS TO HIDE FROM ISOLATED JOBS', 'ENABLE JOB ISOLATION', 'DEFAULT JOB TIMEOUT', 'DEFAULT INVENTORY UPDATE TIMEOUT', 'DEFAULT PROJECT UPDATE TIMEOUT', 'PER-HOST ANSIBLE FACT CACHE TIMEOUT', 'MAXIMUM NUMBER OF FORKS PER JOB', 'RUN PROJECT UPDATES WITH HIGHER VERBOSITY', 'IGNORE ANSIBLE GALAXY SSL CERTIFICATE VERIFICATION', 'FOLLOW SYMLINKS', 'ENABLE ROLE DOWNLOAD', and 'ISOLATED HOST KEY CHECKING'.

Roles and collections are locally cached for performance reasons, and you will need to select **Update Revision on Launch** in the project SCM Update Options to ensure this:

SCM UPDATE OPTIONS

- ☐ CLEAN ?
- ☐ DELETE ON UPDATE ?
- ☒ UPDATE REVISION ON LAUNCH ?
- ☐ ALLOW BRANCH OVERRIDE ?

14.8.1. Using Collections in Tower

Before Tower can use Automation Hub as the default source for collections content, you need to create an API token in the Automation Hub UI so that it could be specified in Ansible Tower. You may connect to a private Automation Hub or a public Automation Hub collection, the only difference is which URL you specify.

1. Navigate to <https://cloud.redhat.com/ansible/automation-hub/token> and click **Load token**.
2. Click the copy icon to copy the API token to the clipboard.

Repo Management

Local Remote

| Distribution name | Repository name | Content c... | Last updated | Sync URL | Ansible CLI URL |
|-------------------|-----------------|--------------|--------------|---|---|
| community | community | 34 | 17 days ago | https://10.10.94.209/api/galaxy/conte... | https://10.10.94.209/api/galaxy/conte... |
| published | published | 6 | 5 days ago | https://10.10.94.209/api/galaxy/conte... | https://10.10.94.209/api/galaxy/conte... |
| red-hat-certified | rh-certified | 195 | an hour ago | https://10.10.94.209/api/galaxy/conte... | https://10.10.94.209/api/galaxy/conte... |

Get token

Repo Management

Local Remote

| Distribution name | Repository name | Content c... | Last updated | Sync URL | Ansible CLI URL |
|-------------------|-----------------|--------------|--------------|---|---|
| community | community | 34 | 17 days ago | https://10.10.94.209/api/galaxy/conte... | https://10.10.94.209/api/galaxy/conte... |
| published | published | 6 | 5 days ago | https://10.10.94.209/api/galaxy/conte... | https://10.10.94.209/api/galaxy/conte... |
| red-hat-certified | rh-certified | 195 | an hour ago | https://10.10.94.209/api/galaxy/conte... | https://10.10.94.209/api/galaxy/conte... |

You can create different repos with different namespaces/collections in them. But for each repo in Automation Hub you need to create a different Automation Hub credential. Copy the **Ansible CLI URL** from the Automation Hub UI in the format of `https://$<hub_url>/api/galaxy/content/<repo you want to pull from>` into the **Galaxy Server URL** field of the *Tower Create Credential* form:

NEW CREDENTIAL

DETAILS PERMISSIONS

* NAME Automation Hub DESCRIPTION ORGANIZATION Default

* CREDENTIAL TYPE Ansible Galaxy/Automation Hub API Token

TYPE DETAILS

* GALAXY SERVER URL <https://10.10.94.209/api/galaxy/content/published/> AUTH SERVER URL API TOKEN

CANCEL SAVE

Note

The token generated in private Automation Hub is used for all content. Repositories such as rh-certified and community do not have dedicated tokens. You should use the token generated for all appropriate ansible-galaxy definitions in the `ansible.cfg` file. See [Locate](#)

[and configure the Ansible configuration file](#) in the *Ansible Tower Administration Guide* for detail.

Refer to [Managing Red Hat Certified and Ansible Galaxy Collections in Ansible Hub](#) for Automation Hub UI-specific instructions.

5. Navigate to the organization for which you want to be able to sync content from Automation Hub and add the new Automation Hub credential to the organization. This step allows you to associate each organization with the Automation Hub credential (i.e. repo) that you want to be able to use content from.



The screenshot shows the 'Default' organization page in Ansible Tower. The page has tabs for 'DETAILS', 'USERS', 'PERMISSIONS', and 'NOTIFICATIONS'. The 'DETAILS' tab is active. It shows fields for 'NAME' (Default), 'DESCRIPTION', 'INSTANCE GROUPS', 'GALAXY CREDENTIALS', and 'MAX HOSTS'. The 'GALAXY CREDENTIALS' section shows two credentials: 'Ansible Galaxy' and 'Automation Hub'. A red arrow points to the 'Automation Hub' credential. The 'MAX HOSTS' field is set to 0. There are 'CANCEL' and 'SAVE' buttons at the bottom right.

Note

Suppose you have two repos:

- *Prod*: Namespace 1 and Namespace 2, each with collection A and B so: namespace1.collectionA:v2.0.0 and namespace2.collectionB:v2.0.0
- *Stage*: Namespace 1 with only collection A so: namespace1.collectionA:v1.5.0 on Automation Hub, you will have a repo URL for *Prod* and *Stage*.

You can create an Automation Hub credential for each one. Then you can assign different levels of access to different organizations. For example, you can create a Developers organization has access to both repos, while an Operations organization just has access to the Automation Hub **Prod** repo only.

Refer to [Managing User Access in Ansible Hub](#) for Automation Hub UI-specific instructions.

6. If the Automation Hub has self-signed certificates, click the toggle to enable the Tower setting **Ignore Ansible Galaxy SSL Certificate Verification**. For **public Automation Hub**, which uses a signed certificate, click the toggle to disable it instead. Note this is a global setting:

JOBS

ANSIBLE MODULES ALLOWED FOR AD HOC JOBS [?](#) REVERT

- * command
- * shell
- * yum
- * apt
- * apt_key
- * apt_repository
- * apt_rpm
- * service
- * group
- * user
- * mount
- * ping
- * selinux
- * setup
- * win_ping
- * win_service
- * win_updates
- * win_group
- * win_user

PATHS TO EXPOSE TO ISOLATED JOBS [?](#) REVERT

ANSIBLE CALLBACK PLUGINS [?](#) REVERT

PATHS TO HIDE FROM ISOLATED JOBS [?](#) REVERT

* ENABLE JOB ISOLATION [?](#) ☒

DEFAULT PROJECT UPDATE TIMEOUT [?](#) REVERT 0

* RUN PROJECT UPDATES WITH HIGHER VERBOSITY [?](#) ☐

ENABLE COLLECTION(S) DOWNLOAD [?](#) ☒

* ISOLATED STATUS CHECK INTERVAL [?](#) REVERT 30

ENABLE DETAILED RESOURCE PROFILING ON ALL PLAYBOOK RUNS [?](#) ☐

* JOB EXECUTION PATH [?](#) /tmp

* MAXIMUM SCHEDULED JOBS [?](#) 10

DEFAULT JOB TIMEOUT [?](#) REVERT 0

PER-HOST ANSIBLE FACT CACHE TIMEOUT [?](#) REVERT 0

IGNORE ANSIBLE GALAXY SSL CERTIFICATE VERIFICATION [?](#) ☒

FOLLOW SYMLINKS [?](#) ☐

* ISOLATED LAUNCH TIMEOUT [?](#) REVERT 600

DEFAULT INVENTORY UPDATE TIMEOUT [?](#) REVERT 0

MAXIMUM NUMBER OF FORKS PER JOB [?](#) REVERT 200

ENABLE ROLE DOWNLOAD [?](#) ☒

ISOLATED HOST KEY CHECKING [?](#) ☐

ISOLATED CONNECTION TIMEOUT [?](#) REVERT 10

7. Create a project, where the source repository specifies the necessary collections in a requirements file located in the collections/requirements.yml file. Refer to the syntax described in the Ansible documentation: https://docs.ansible.com/ansible/latest/user_guide/collections_using.html#install-multiple-collections-with-a-requirements-file.

PROJECTS / CREATE PROJECT 🔊

NEW PROJECT ⓘ

DETAILS | PERMISSIONS | JOB TEMPLATES | SCHEDULES

* NAME Collections Project

DESCRIPTION

* ORGANIZATION [?](#) Default

* SCM TYPE Git

SOURCE DETAILS

* SCM URL [?](#) https://github.com/ansible-collections/

SCM BRANCH/TAG/COMMIT [?](#)


SCM REFSPEC [?](#)

SCM CREDENTIAL

SCM UPDATE OPTIONS

- ☐ CLEAN [?](#)
- ☐ DELETE ON UPDATE [?](#)
- ☐ UPDATE REVISION ON LAUNCH [?](#)
- ☐ ALLOW BRANCH OVERRIDE [?](#)

CANCEL SAVE

8. In the Projects list view, click  to run an update against this project. Tower fetches the Galaxy collections from the collections/requirements.yml file and report it as changed; and the collections will now be installed for any job template using this project.

Note

If updates are needed from Galaxy or Collections, a sync is performed that downloads the required roles, consuming that much more space in your /tmp file. In cases where you have a big project (around 10 GB), disk space on /tmp may be an issue.

For more information on collections, refer to [Using Collections](#). For more information on how Red Hat itself publishes one of these official collections, which can be used to automate your Tower install directly, refer to the [AWX Ansible Collection](#) documentation. This page is accessible with your Red Hat customer credentials as part of your Red Hat Ansible Automation Platform subscription.