Thank you for your interest in Ansible Tower, the open source IT orchestration engine. Whether sharing operations tasks with your team or integrating with Ansible through the Tower REST API, Tower provides powerful tools to make your automation life easier.

The *Ansible Tower Quick Setup Guide* covers basic steps for using Ansible Tower and running your first playbook. This document has been updated to include information for the latest release of Ansible Tower 2.4.3.

Ansible Tower Version 2.4.3; December 18, 2015; [https://access.redhat.com/](https://access.redhat.com/)
You can expect the Quick Start process to take less than thirty minutes. At the end of the Quick Start, you will have a functioning Tower application that you can use to launch more sophisticated playbooks.

To begin, you must install Tower and you must choose a target system where the initial playbook deploys. This first playbook executes simple Ansible tasks, while teaching you how to use Tower, as well as ensuring its proper setup. This can be any sort of system manageable by Ansible, as described at: http://docs.ansible.com/intro_installation.html.

Note: Ansible Tower is available as a Basic license, Enterprise license, or Premium license. These licenses vary in price, support-levels, and features. Enterprise and Premium license users will have access to some extended Tower features that Basic users will not have available. For more information about licenses and features, refer to Licensing, Updates, and Support in the Tower Installation and Reference Guide.

This Quick Start Guide covers setting up with a Basic license in mind. Features only available to Enterprise or Premium license users are noted as you go along.
Using the login information provided by the configuration wizard, open a web browser and log in to Tower by browsing to the Tower server URL at: https://<Tower server name>/

**Note:** Tower installs a self-signed certificate for HTTPS communication which may require acceptance in your browser. Refer to the General Installation Notes in the Tower Installation and Reference Guide for help with replacing this certificate if needed.

Log in using the username and password set during the installation process. By default, this is username: *admin* and password: *password*.

These defaults can be changed later by clicking on the Setup Menu button and selecting the Users navigational link.
Tower requires a valid license to run. If you did not receive a license from Ansible directly or via email, or have issues with the license you received, refer to http://www.ansible.com/license for free and paid license options (including free trial licenses) or contact Ansible via the Red Hat Customer Portal at https://access.redhat.com/.

Paste in the license you received from Ansible, agree to the End User License Agreement, and click Submit.

**Note:** Users of older versions of Ansible Tower, prior to 2.2, can review their license from the ‘View License’ drop-down of the Tower user’s menu at the top right of the screen.

Once accepted, your license allows you to continue on to the main Ansible interface. For later reference, you can view this license from the Setup ( ) Menu’s ‘View License’ link.
The Ansible Tower license was updated. To review or update the license, choose View License from the Setup menu.
EXAMINE THE TOWER DASHBOARD

Note: Ansible Tower 2.2 provides a streamlined interface, with the button offering access to administrative configuration needs. Users of older versions of Ansible Tower (pre-2.2) can access most of these through the top-level navigational menu.

The Tower Dashboard offers a friendly graphical framework for your IT orchestration needs. Across the top-left side of the Tower Dashboard, administrators can quickly navigate to their Projects, Inventories, Job Templates, and Jobs. Across the top-right side of this interface, administrators can access the tools they need to configure organizations, users, groups, and permissions.

On the main Tower Dashboard screen, a summary appears listing your current Hosts, Inventories, and Projects. You can view charts and graphs for Job Status and Host Status by clicking on their tabs. Also available for review are summaries of Recent Used Job Templates and Recent Run Jobs.

Note: Keep in mind that the goal of this Quick Start is to launch a simple playbook. To do this, a number of configuration options must be setup. Completing the quick start configuration tasks now ensures that Tower is configured
properly and allows for easier executions of more involved playbooks later on.
To enter the Setup Menu screen for Ansible Tower, click the button. This screen allows you to create your organization, add credentials, add users and teams, schedule management jobs, and more.

**Note:** The Setup Menu screen is new for Ansible Tower 2.2. Users of older versions can click on the self-titled links from the main navigation menu to access Organizations, Teams, Users, Credentials, etc.

<table>
<thead>
<tr>
<th>A T O W E R</th>
<th>Projects</th>
<th>Inventories</th>
<th>Job Templates</th>
<th>jobs</th>
</tr>
</thead>
</table>

**Credentials**
Add passwords, SSH keys, etc. for Tower to use when launching jobs against machines, or when syncing inventories or projects.

**Users**
Allow others to sign into Tower and own the content they create.

**Teams**
Split up your organization to associate content and control permissions for groups.

**Organizations**
Group all of your content to manage permissions across departments in your company.

**Management Jobs**
Manage the cleanup of old job history, activity streams, data marked for deletion, and system tracking info.

**Inventory Scripts**
Create and edit scripts to dynamically load hosts from any source.

**View Your License**

**About Tower**
CHAPTER SIX

REVIEW THE ORGANIZATION

An organization is a logical collection of users, teams, projects, and inventories. Consider it the highest level object in the Tower object hierarchy.

From the Setup Menu screen, click on Organizations.

Note: Ansible Tower 2.2 creates a default organization automatically. Users of older versions of Tower will not see
this default organization.

A default organization has been automatically created and is available to all users of Ansible Tower. It can be used as is or edited later as needed.

For the purpose of this Quick Start Guide, leave the default organization as is and click **Save**.

**Note:** If you are using Ansible Tower with a Basic license, you must use the default organization. Only Enterprise or Premium Tower licenses have the ability to add new organizations beyond the default.

To edit the default organization later, expand its ‘Properties’ and enter the appropriate details. Then, click **Save**.

Enterprise and Premium Tower license users who want to add a new organization should refer to the Organizations section in the Tower User Guide.
ADD A USER TO THE ORGANIZATION

Expand the Users section at the bottom of the default organization you just saved (not from the Setup [ ] Screen’s User menu item).

To add a user, click the button.

Because you have not created any users yet, the “admin” user is the only user listed. Click the button again to create a new user and enter the user’s details. Leave the Organization field set to default for now.
Click the Save button. The organization’s user information becomes available for viewing and the new user you created appears on the list.
CREATE A NEW INVENTORY AND ADD IT TO THE ORGANIZATION

An inventory is a collection of hosts managed by Tower. Inventories are assigned to organizations, while permissions to launch playbooks against inventories are controlled at the user and/or team level. For more information, refer to Inventories, Users - Permissions, and Teams - Permissions in the “Tower User Guide”.

Create a new inventory by browsing to the Inventories main navigation link and clicking the button.

Enter the values for Name and Description. Leave the organization set as default in the Organization field.

For now, ignore the Variables field. Click the Save button at the bottom of the page to create the inventory.
After clicking **Save**, ‘Scan Job Templates’ becomes available at the bottom of the screen. For now, ignore this section. For more information, refer to **Job Templates** in the “Tower User Guide”. Click **Save** again and a populated **Inventories** screen appears.

Click on the ‘Web Servers’ link for the inventory you created.

Note that inventories are divided into groups and hosts. A group might represent a particular environment (e.g. “Datacenter 1” or “Stage Testing”), a server type (e.g. “Application Servers” or “DB Servers”), or any other representation of your environment.

The left side of the screen displays the groups that belong to the Web Servers inventory. The right side displays hosts.

---

**Note:** Prior to Ansible Tower 2.2, hosts could not be added to the Web Servers inventory before adding a group. If
you are using an older version of Tower, click the button above the ‘Groups’ section to add a group before adding a host.

For this example, suppose that the organization you created earlier has a group of web server hosts supporting the corporate CMS application. To add these hosts to the Web Servers inventory, create a “CMS Web” group. Click the Save button to create the group.

Beginning with Ansible Tower 2.2, hosts can be added directly to the inventory root or created under a group. In the “Web Servers” inventory you created, select the button above ‘Hosts’ to create the new host. (To add a host to a group, click on the group to which the host should be assigned and add the host from there.)

**Note:** For the purposes of this Quick Start and to test that Tower is setup properly, add a host that you can reach via SSH and manage using Ansible (i.e. that meets the Ansible requirements). The simple Ansible playbook launched will not harm or alter the target in any way. If you cannot reach the host via SSH, the playbook launched later in this Quick Guide will fail.

Enter the “Host Name”, either the DNS resolvable name of the host or its IP address. The **Description** is arbitrary. Click Save to finish adding the host.

Click Save to finish adding the host.
### Web Servers

<table>
<thead>
<tr>
<th>Groups</th>
<th>Actions</th>
<th>Hosts</th>
<th>Search</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

No records matched your search.
CHAPTER

NINE

CREATE A CREDENTIAL

Credentials authenticate the Tower user to launch Ansible playbooks against inventory hosts, which can include passwords and SSH keys. You can also require the Tower user to enter a password or key phrase when a playbook launches using the credentials feature of Tower.

Create a new credential by browsing to and clicking on the **Credentials** link from the button.

**Note:** When setting up your credential, keep in mind that the user you assign must have root access or be able to use SSH to connect to the host machine.

Click the **button to create a new credential.

Enter an arbitrary **Name** and **Description** for this credential based on the user you created earlier. Select the “User” radio button.
Click the button to find the user you created earlier, then select that user.

Next, select credential type **Machine**.

Enter the details of the appropriate authentication mechanism to use for the host you added to Tower earlier. For testing purposes, remember to use credentials for a real host that you can access via SSH. To keep things simple, rather than entering the password here, you will enter it later when a playbook launches using these credentials. Check the box **Ask at runtime?**.

**Note:** Tower supports different options for what you want to store for credentials in this box. Uploading a locked SSH key is recommended and Tower can prompt you for the SSH unlock password for use with ssh-agent when launching the job. Tower encrypts passwords and key information in the Tower database and never makes secret information
visible via the API.

Click **Save** and your newly created credential appears.
There are two ways to access a simple playbook for this Quick Start example—manually or by specifying a link from a Github repository.

**Note:** It is recommended that, whenever possible, you use source control to manage your playbooks. This type of best practice provides the ability to treat your infrastructure as code and is in line with DevOps ideals.

### 10.1 Using Github for Playbook Access

**Linking a playbook from Github**

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](https://github.com/ansible/tower-example.git)

This link offers a very similar playbook to the one created manually in the instructions below. Using it will not alter or harm your system in anyway.

**Create the project**

Using your web browser, create the new project by clicking on the **Projects** link at the top of the Tower Dashboard.

Click the **button, then enter a **Name** and **Description** for the project.

The **Project Base Path** displays the value entered during the installation of Tower. You cannot edit it from this dialog. (Refer to the **Administration Guide** for more information on how to alter this value.)

Set the **SCM Type** *Git*.

**Playbook Setup:**

Under **SCM URL**, provide the Github address to the playbook. For the purposes of this Quick Start, ignore the **SCM Branch**, **SCM Credential**, and **SCM Update Options** input fields.
Select **Save** and the new project displays.

### 10.2 Accessing a Manually Created Playbook

**Creating a playbook**

Using a command line console as the root user, create a subdirectory for your project on the Tower server filesystem, in which to store your Ansible playbooks for this project.
Make a new project directory by creating it on the Tower filesystem under the **Project Base Path** directory, located by default in “/var/lib/awx/projects/”. For this example, the directory is *helloworld*.

```
root@localhost:~$ mkdir /var/lib/awx/projects/helloworld
```

Next, use your preferred editor to create a simple Ansible playbook. Make a file called “helloworld.yml” inside of the directory you just created.

```
root@localhost:~$vi /var/lib/awx/projects/helloworld/helloworld.yml
```

The contents of the file are below:

```yaml
---
- name: Hello World!
  hosts: all
  tasks:
  - name: Hello World!
    shell: echo "Hi! Tower is working!"
```

Please take note of the indentation as it plays an important role. Once saved, this playbook file can test Tower running a playbook against the host in the inventory.

**Note:** If you have trouble navigating to the directories mentioned above or creating the *helloworld.yml* file due to permissions, use *sudo* as you execute your commands if you do not have root access.

### Create the project

Using your web browser, create the new project by clicking on the **Projects** link at the top of the Tower Dashboard.

Click the **+** button, then enter a **Name** and **Description** for the project.

The **Project Base Path** displays the value entered during the installation of Tower. You cannot edit it from this dialog. (Refer to the Administration Guide for more information on how to alter this value.)

Set the **SCM Type** to **Manual**.

**Playbook Setup:**

For the **Playbook Directory**, select a value which corresponds to the subdirectory you created.
Warning: If the helloworld/ project directory and file are incorrectly created, and if the permissions are incorrect, you will encounter the following message:

```
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.
```

Double check your settings and use the command `chown -R awx` against the project directory, if necessary. If SE Linux is enabled, check the directory and file context.

Select Save and the new project displays.

**10.2. Accessing a Manually Created Playbook**
CREATE A NEW JOB TEMPLATE

A job template combines an Ansible playbook from a project and the settings required to launch it. Create a new job template by browsing to **Job Templates**.

Click the **+** button.
Enter values for the **Name** and **Description**. Jobs can be of type **Run**, **Check**, or **Scan**. Select **Run** for this Quick Start.

Choose the **Inventory**, **Project**, and **Credential** from those you created earlier. The playbook drop-down menu automatically populates from the project path and playbook we created earlier. Choose the “**helloworld**” playbook.

You can leave the other values, such as **Cloud Credential**, **Forks**, **Limit**, and **Job Tags** set to their default values or blank. Select the **Verbosity** level to control the output ansible produces as the playbook executes. Refer to **Job Templates** for more information.

Click **Save** and the new project displays.
To launch the playbook, browse to the **Job Templates** tab and click the button on the template.

Tower prompts you for the any passwords you configured when setting up the credential. Enter and confirm your secure password(s).

Tower redirects the browser to the status page for this job from the **Jobs** link, where you can watch this job as it runs.
This page automatically refreshes using Tower’s Live Event feature until the job is complete.
To view the standard output for the job once it has finished, click the button.

You can also drill down into individual tasks.
Select the 127.0.0.1 host entry under **Events Summary** to see the events for that host.

To view an event’s details, click on any event in the list:
Click **Results**.

Congratulations! Your Tower installation is officially setup and running properly. To learn more about these Tower features or to learn about administration tasks, the Tower API, etc., refer to the following documentation sets:

- Ansible Tower User Guide
- Ansible Tower Installation and Reference Guide
- Ansible Tower Administration Guide
- Ansible Tower API Guide
• Ansible Documentation.
CHAPTER

THIRTEEN

INDEX

• genindex

This document is Copyright © 2015 Red Hat, Inc. All rights reserved.

Ansible, Ansible Tower, Red Hat, and Red Hat Enterprise Linux are trademarks of Red Hat, Inc., registered in the United States and other countries.

If you distribute this document, or a modified version of it, you must provide attribution to Red Hat, Inc. and provide a link to the original version.

Third Party Rights

Ubuntu and Canonical are registered trademarks of Canonical Ltd.

The CentOS Project is copyright protected. The CentOS Marks are trademarks of Red Hat, Inc. (“Red Hat”).

Microsoft, Windows, Windows Azure, and Internet Explore are trademarks of Microsoft, Inc.

VMware is a registered trademark or trademark of VMware, Inc.

Rackspace trademarks, service marks, logos and domain names are either common-law trademarks/service marks or registered trademarks/service marks of Rackspace US, Inc., or its subsidiaries, and are protected by trademark and other laws in the United States and other countries.

Amazon Web Services™, “AWS”, “Amazon EC2”, and “EC2”, are trademarks of Amazon Web Services, Inc. or its affiliates.

OpenStack™ and OpenStack logo are trademarks of OpenStack, LLC.

Chrome™ and Google Compute Engine™ service registered trademarks of Google Inc.

Safari® is a registered trademark of Apple, Inc.

Firefox® is a registered trademark of the Mozilla Foundation.

All other trademarks are the property of their respective owners.
C
credentials
  create, 17

D
dashboard, Tower
  examine, 6

I
inventory
  add to organization, 13
  create new, 13

J
job templates
  creation, 25
  launch, 27

L
launch
  job templates, 27
license
  import, 4
log in
  Tower, 3

O
organization
  default, 9, 10
organizations, 9

P
project
  create, 21, 23
  github playbook linking, 21
  manual playbook creation, 22
projects, 21

Q
quick start, intro, 2

S
setup menu, 8

T
Tower
  log in, 3

U
users
  add new, 11
  add to organization, 11