ONESTOP

Red Hat Ansible Automation Platform

Enablement Team

Workshop Ansible Tower Enablement Specialist

Workshop Description

Ansible Tower will enable you to create playbooks, while building in security. Automation features will save time, empower junior staff, offload senior staff and automate your most tedious tasks!

Minimal Requirements.

Minimal VM Machine RHEL 8.0

4096 Gb RAM

20Gb HD

1 NIC Interface

RHEL subscription and Ansible Tower License ONLY FOR CHAMPIONS
https://mojo.redhat.com/docs/DOC-1094354

Who should attend

- Architects
- Developers
- Technical Leads
- Operations Engineers

What you will learn

- How to run ad hoc commands
- How to write a playbook
- How to install and configure Red Hat Ansible Tower
- How to use templates to save time
- How to enhance security, using Ansible Tower

Overview

The Ansible Tower Workshop is meant for anyone who has any exposure to Ansible, whether you have used it or not. We are going to start with a short overview and then we'll get into the lab as soon as possible. That is where we will spend most of our time.

Your Responsibilities

Have a Discussion. This will be boring if it's just us, up here talking for over 4 hours.

Participate. We are going to cut you loose with Ansible, in just a little while. Have questions. Have opinions.

Hopefully you have your laptop with you. If not, please find a shoulder-surfing buddy. See? Not only can we dig into Ansible but you can make a new friend!

Exercises

Exercise 2.0 - Installing Ansible Tower

Exercise Description

In this exercise, we are going to get Ansible Tower installed on your control node.

Step 1: Change directories

Change directories to /tmp

```
cd /tmp
```

Step 2: Download Red Hat Ansible Tower

Download the latest Ansible Tower package

```
curl -O
https://releases.ansible.com/ansible-tower/setup/ansible-tower-setup
-latest.tar.gz
```

Step 3: Untar and unzip the package file

```
tar xvfz /tmp/ansible-tower-setup-latest.tar.gz
```

Step 4: Change directories

Change directories into the Ansible Tower setup package

```
cd /tmp/ansible-tower-setup-*/
```

Step 5: Open inventory file

Using an editor of your choice, open the inventory file

```
vim inventory
```

Step 6: Identify variables

Fill a few variables out in an inventory file: `admin_password`, `pg_password`, `rabbitmq_password`

```
[tower]
localhost ansible_connection=local

[database]

[all:vars]
admin_password='ansibleWS'

pg_host=''
pg_port=''

pg_database='awx'
pg_username='awx'
pg_password='ansibleWS'

rabbitmq_username=tower
rabbitmq_password='ansibleWS'
rabbitmq_cookie=cookiemonster

# Isolated Tower nodes automatically generate an RSA key for
authentication;
# To disable this behavior, set this value to false
# isolated_key_generation=true
```

~

Step 7: Run setup

Run the Ansible Tower setup script

```
sudo ./setup.sh
```

> Step 7 will take approx. 10-15 minutes to complete. This may be a good time to take a break.

Step 8: Confirm results

At this point, your Ansible Tower installation should be complete. You can access your Tower workshop (not forgetting that `workshopname` is the name of your workshop, and # is your student number) at:

```
example.tower.0.redhatgov.io
```

Ensuring Installation Success

You know you were successful if you are able to browse to your Ansible Tower's url (*control node's IP address*) and get something like this

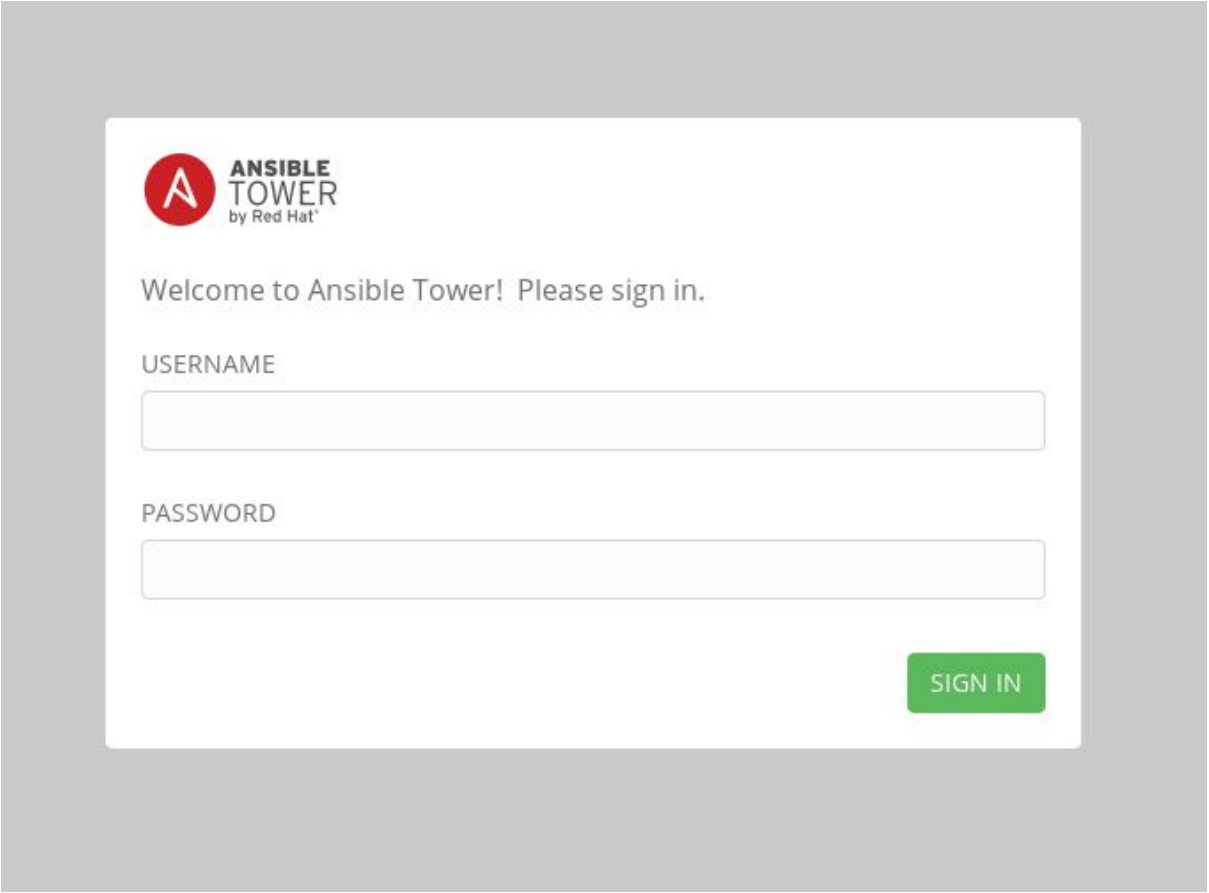


Figure 2: Ansible Tower Login Screen

Exercise 2.1 - Configuring Ansible Tower

Exercise Description

In this exercise, we are going to configure Red Hat Ansible Tower, so that we can run a playbook.

There are a number of constructs in the Ansible Tower UI that enable multi-tenancy, notifications, scheduling, etc. However, we are only going to focus on the key concepts required for this workshop today.

- Credentials
- Projects
- Inventory
- Job Template

Section 1: Logging into Ansible Tower and installing the license key

Step 1: Log in

To log in, use the username `admin` and and the password `ansibleWS`.

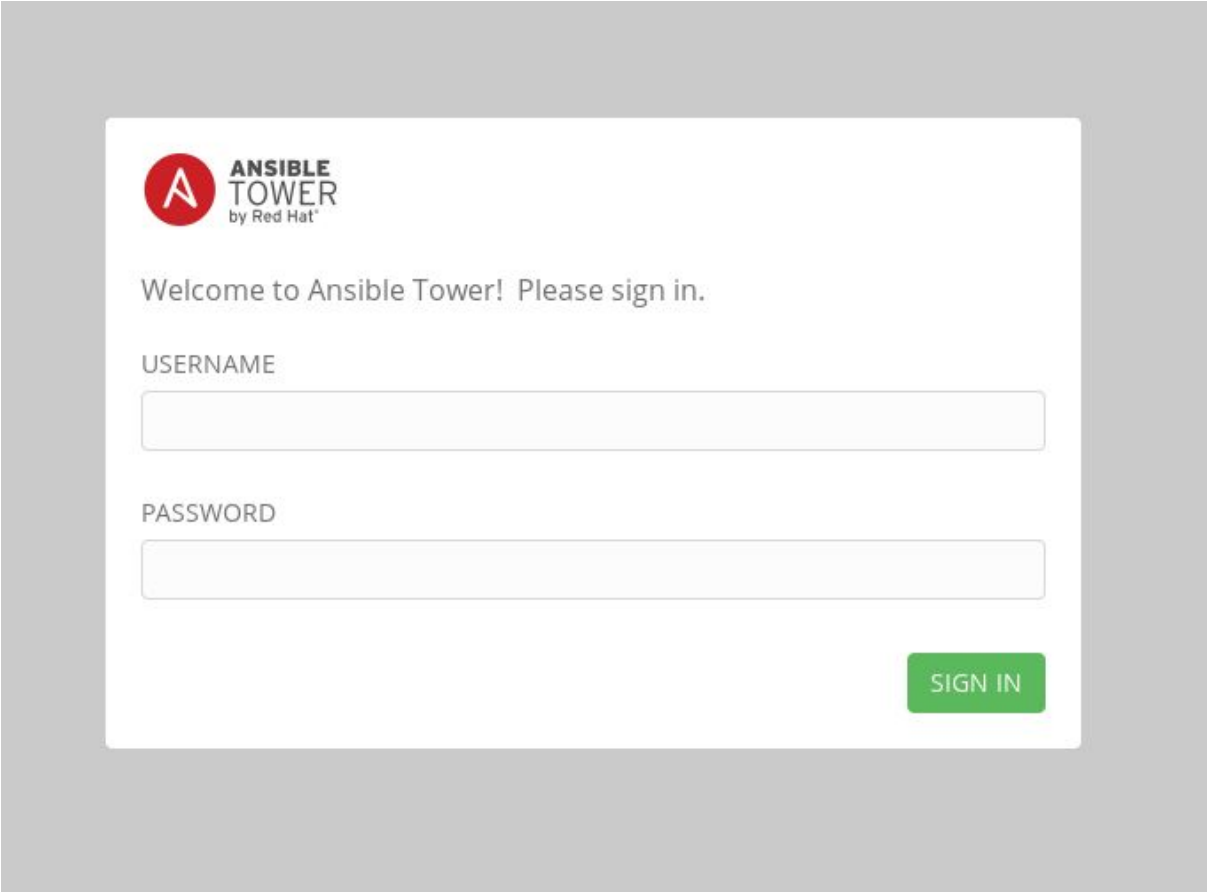


Figure 3: Ansible Tower Login Screen

After you login, you will be prompted to request an Ansible Tower license, or browse for an existing license file

https://mojo.redhat.com/docs/DOC-1094354 Get a License here !!

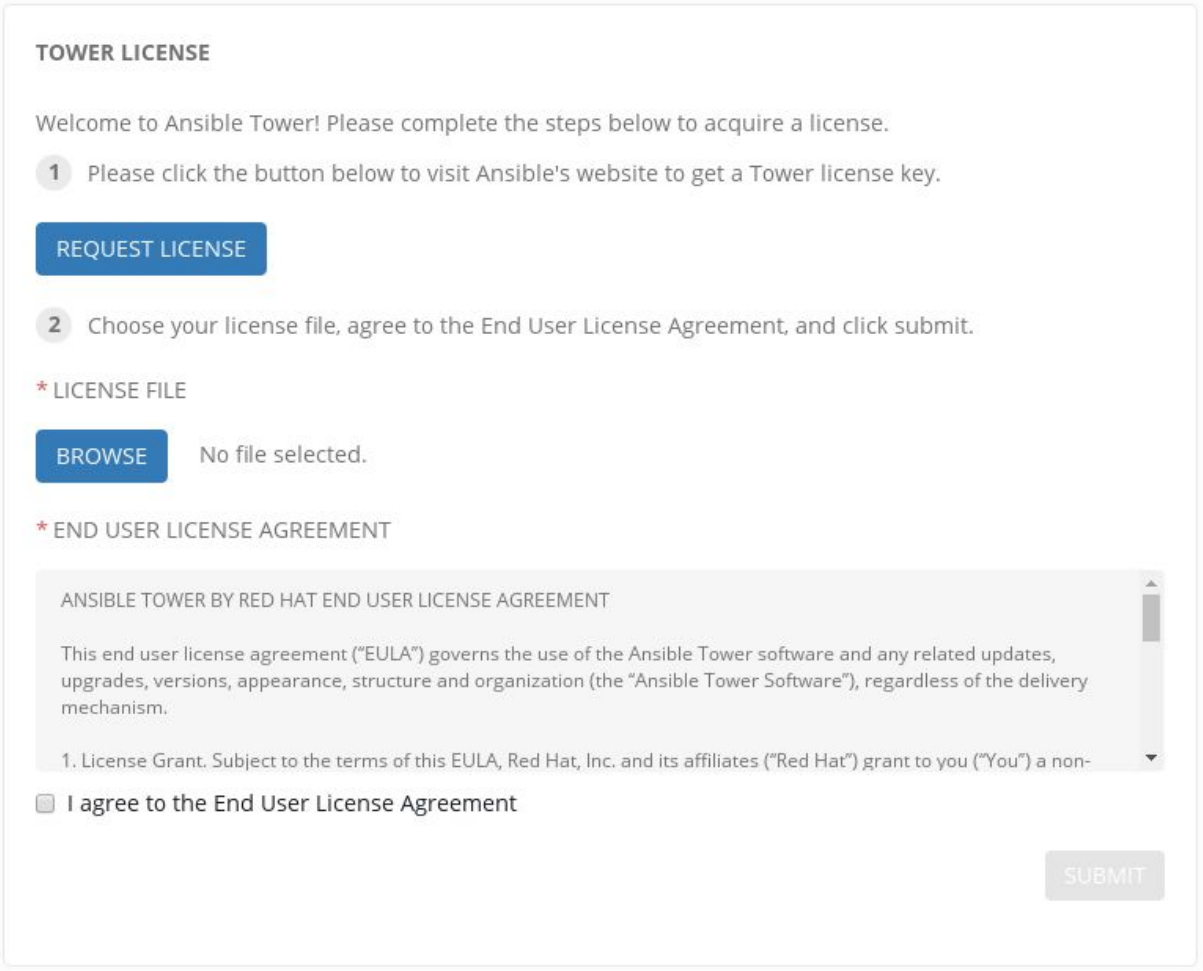


Figure 4: Uploading a License

Step 2: Request workshop license

In a separate browser tab, browse to https://www.ansible.com/workshop-license to request a workshop license.

Step 3: Upload Ansible Tower workshop license

Back in the Ansible Tower UI, choose BROWSE  and upload your recently downloaded license file into Ansible Tower.

Step 4: Accept license agreement

Select "*I agree to the End User License Agreement*".

Step 5: Submit

Select SUBMIT 

Section 2: Creating a credential

Credentials are utilized by Ansible Tower for authentication when launching jobs against machines, synchronizing with inventory sources, and importing project content from a version control system.

There are many types of credentials including machine, network, and various cloud providers. In this workshop, we are using a `machine` credential.

Step 1: Select Credentials

Select CREDENTIALS, from the left-side menu bar. 

Step 2: Select Add

Select ADD 

Step 3: Complete the Ansible Tower Credentials form

Complete the form, using the following entries and your private Ansible Tower SSH key. When you paste the key in the `PRIVATE KEY` field, make sure to include the `-----BEGIN RSA PRIVATE KEY-----` and `-----BEGIN RSA PRIVATE KEY-----` lines.

```
cat ~/.ssh/example-tower && echo
```

NAME	Ansible Workshop Credential
DESCRIPTION	Machine credential for run job templates during workshop
ORGANIZATION	Default
TYPE	Machine
USERNAME	ec2-user
PRIVILEGE ESCALATION	Sudo

PRIVATE KEY	paste in secret key

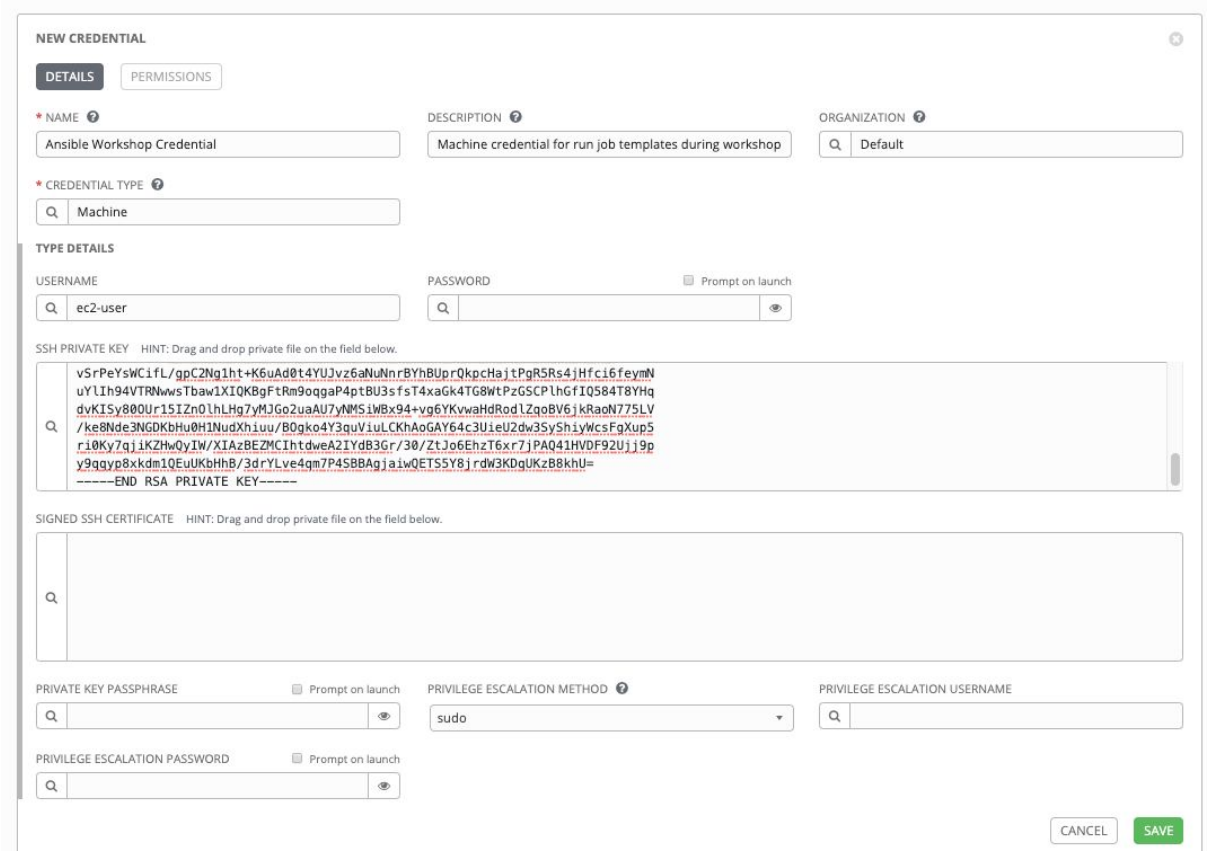


Figure 5: Adding a Credential

Step 4: Save

Select SAVE 

Section 3: Creating a Project

A Project is a logical collection of Ansible playbooks, represented in Ansible Tower. You can manage playbooks and playbook directories, by either placing them manually under the Project Base Path on your Ansible Tower server, or by placing your playbooks into a source code management (SCM) system supported by Ansible Tower, including Git, Subversion, and Mercurial.

Step 1: Open a new project

Select PROJECTS 

.

Step 2: Add the project

Select ADD 

Step 3: Complete the Project form

Complete the form using the following entries:

NAME	Ansible Workshop Project
DESCRIPTION	workshop playbooks
ORGANIZATION	Default
SCM TYPE	Git
SCM URL	https://github.com/ansible/lightbulb
SCM BRANCH	
SCM UPDATE OPTIONS	Clean Delete on Update Update Revision on Launch

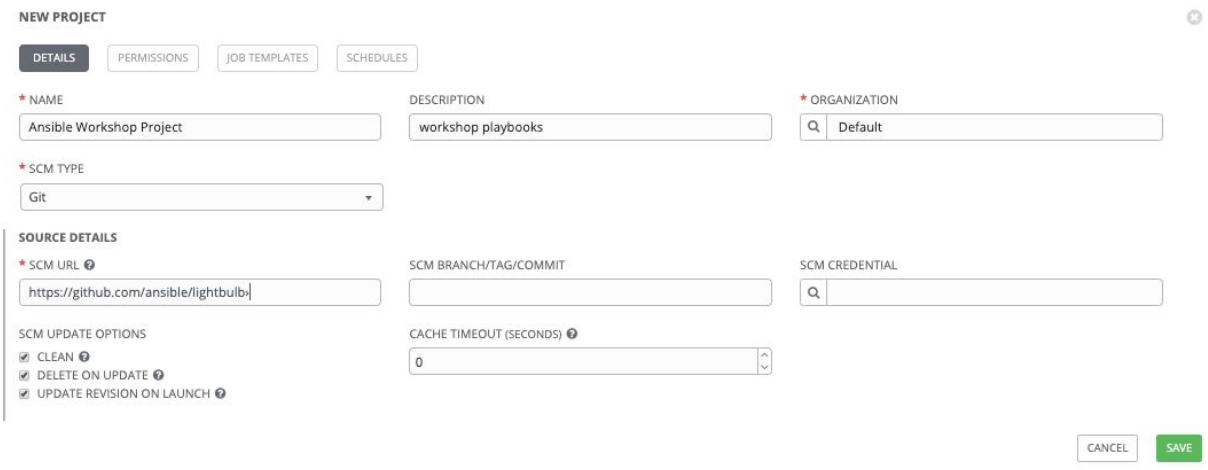


Figure 6: Defining a Project

Step 4: Save

Select SAVE 

Section 4: Creating an Inventory

An inventory is a collection of hosts, against which jobs may be launched. Inventories are divided into groups and these groups contain the actual hosts.

Groups may be sourced `manually`, by entering host names into Ansible Tower, or from one of Ansible Tower's supported cloud providers.

An Inventory can also be `imported` into Ansible Tower using the `tower-manage` command. This is how we are going to add an inventory for this workshop.

Step 1: Navigate to Inventories main link

Select INVENTORIES 

Step 2: Add a new inventory

Select ADD  , and then select Inventory.

Step 3: Complete the Inventory form

Complete the form, using the following entries:

NAME	Ansible Workshop Inventory
DESCRIPTION	workshop hosts
ORGANIZATION	Default

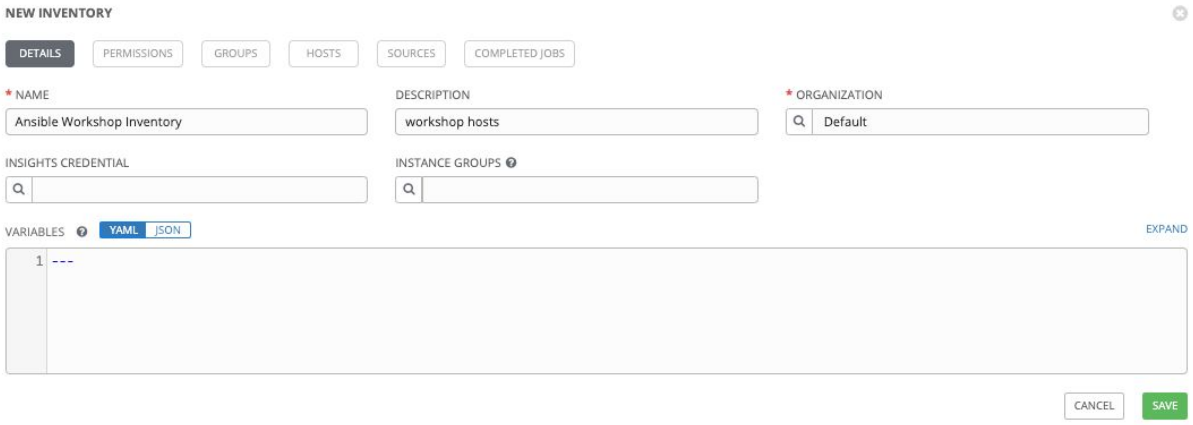


Figure 7: Create an Inventory

Step 4: Save

Select SAVE

Step 5: Switch back to your terminal session

Switch back to your terminal session. If by any chance you closed the wetty browser window, open a new one with the URL shown, below:

```
https://example.tower.0.redhatgov.io:8888/wetty
```

Step 6: Import an existing inventory

Use the `tower-manage` command to import an existing inventory. (*Be sure to replace <username> with your actual username.*)

```
sudo tower-manage inventory_import --source=/home/ec2-user/hosts
--inventory-name="Ansible Workshop Inventory"
```

You should see output similar to the following:

```
  1.646 INFO     Updating inventory 2: Ansible Workshop Inventory
  2.071 INFO     Reading Ansible inventory source: /home/ec2-user/hosts
  2.073 INFO     Using VIRTUAL_ENV: /var/lib/awx/venv/ansible
  2.073 INFO     Using PATH: /var/lib/awx/venv/ansible/bin:/var/lib/awx/venv/awx/bin:/opt/rh/rh-python36/root/usr/bin:/sbin:/bin:/usr/sbin:/usr/bin
  2.073 INFO     Using PYTHONPATH: /var/lib/awx/venv/ansible/lib/python2.7/site-packages:
  2.396 ERROR    ansible-inventory 2.8.1
  2.396 ERROR      config file = /home/ec2-user/ansible.cfg
  2.396 ERROR      configured module search path = [u'/root/.ansible/plugins/modules', u'/usr/share/ansible/plugins/modules']
  2.396 ERROR      ansible python module location = /usr/lib/python2.7/site-packages/ansible
  2.396 ERROR      executable location = /bin/ansible-inventory
  2.396 ERROR      python version = 2.7.5 (default, Jun 11 2019, 12:19:05) [GCC 4.8.5 20150623 (Red Hat 4.8.5-36)]
  2.396 ERROR    Using /home/ec2-user/ansible.cfg as config file
  2.396 ERROR    host_list declined parsing /home/ec2-user/hosts as it did not pass it's verify_file() method
  2.396 ERROR    script declined parsing /home/ec2-user/hosts as it did not pass it's verify_file() method
  2.396 ERROR    auto declined parsing /home/ec2-user/hosts as it did not pass it's verify_file() method
  2.396 ERROR    Parsed /home/ec2-user/hosts inventory source with ini plugin
  2.397 INFO     Processing JSON output...
  2.397 INFO     Loaded 1 groups, 1 hosts
  2.503 INFO     Inventory import completed for  (Ansible Workshop Inventory - 7) in 0.9s
```

Figure 8: Importing an inventory with tower-manage

Feel free to browse your inventory in Ansible Tower, by selecting `Hosts`.

You should now notice that the inventory has been populated with each each of hosts and corresponding inventory.

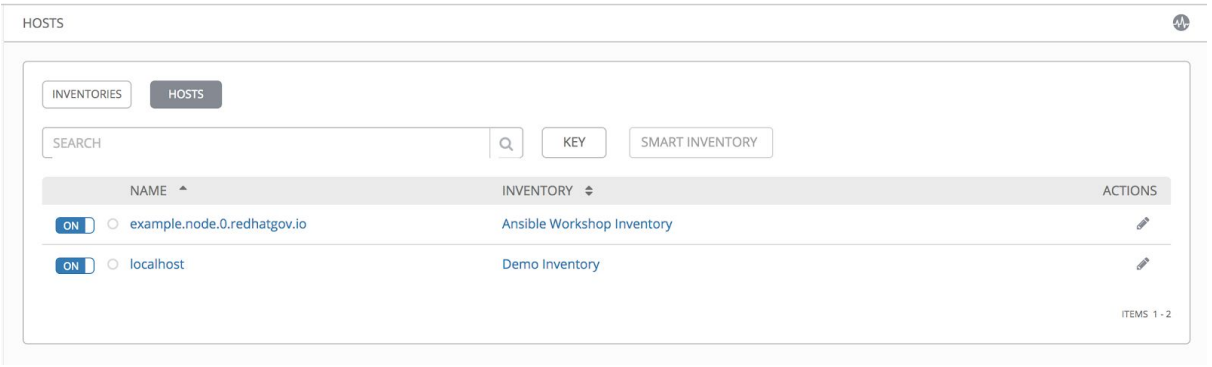


Figure 9: Inventory with Groups

End Result

At this point, we are working with our basic configuration of Ansible Tower. In Exercise 2.2, we will be solely focused on creating and running a job template so you can see Ansible Tower in action.

Exercise 2.2 - Creating and Running a Job Template

Exercise Description

This exercise will walk you through the steps required to create a job template and run it. A job template is a definition and set of parameters for running an Ansible job. In other words, a template combines an Ansible project playbook and the settings required to launch it, into one package.

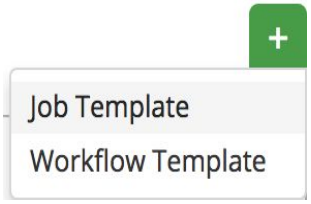
Templates save setup time, for jobs that are launched repetitively. Once the template is set, it can be edited for future jobs, with different settings. Templates also drive uniformity, by running templates exactly the same way every time. And, because we're using templates, production responsibilities can be delegated to less experienced production personnel.

Section1: Creating a job template

Step 1: Navigate to the Templates tab

Select TEMPLATES 

Step 2: Create a new template

Click + and select JOB TEMPLATE 

Step 3: Complete the job Template form

Complete the template form, by entering the following values:

NAME	Apache Basic Job Template
DESCRIPTION	Template for the apache-basic-playbook
JOB TYPE	Run
INVENTORY	Ansible Workshop Inventory
PROJECT	Ansible Workshop Project
PLAYBOOK	examples/apache-basic-playbook/site.yml
MACHINE CREDENTIAL	Ansible Workshop Credential
LIMIT	web
OPTIONS	Enable Privilege Escalation

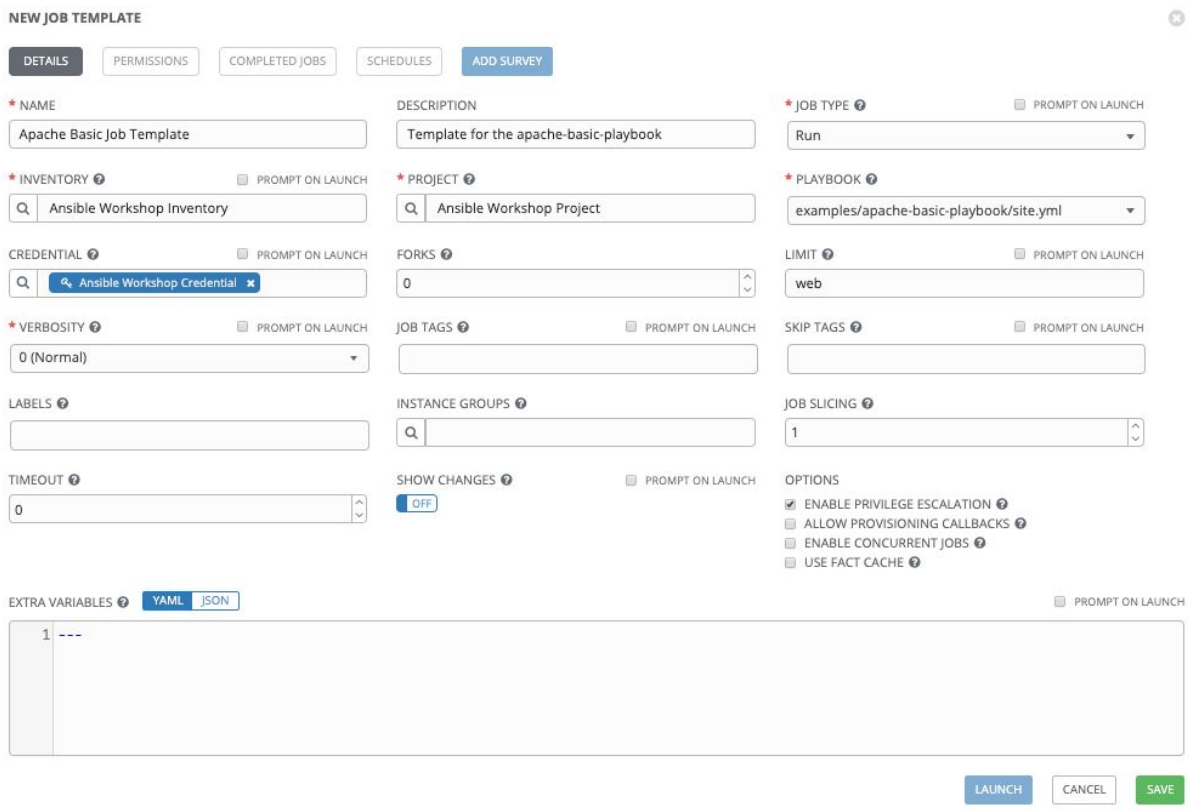


Figure 11: Job Template Form

Step 4: Save

Click SAVE  and then select ADD SURVEY 

Step 5: Complete survey

Job surveys provide extra playbook variables and present them in a user-friendly way. They also enable validation of the input you provide.

Complete the survey form with following values.

PROMPT	Please enter a test message for your new website
DESCRIPTION	Website test message prompt
ANSWER VARIABLE NAME	apache_test_message

ANSWER TYPE	Text
MINIMUM/MAXIMUM LENGTH	Use the defaults
DEFAULT ANSWER	Be creative, keep it clean, we’re all professionals here

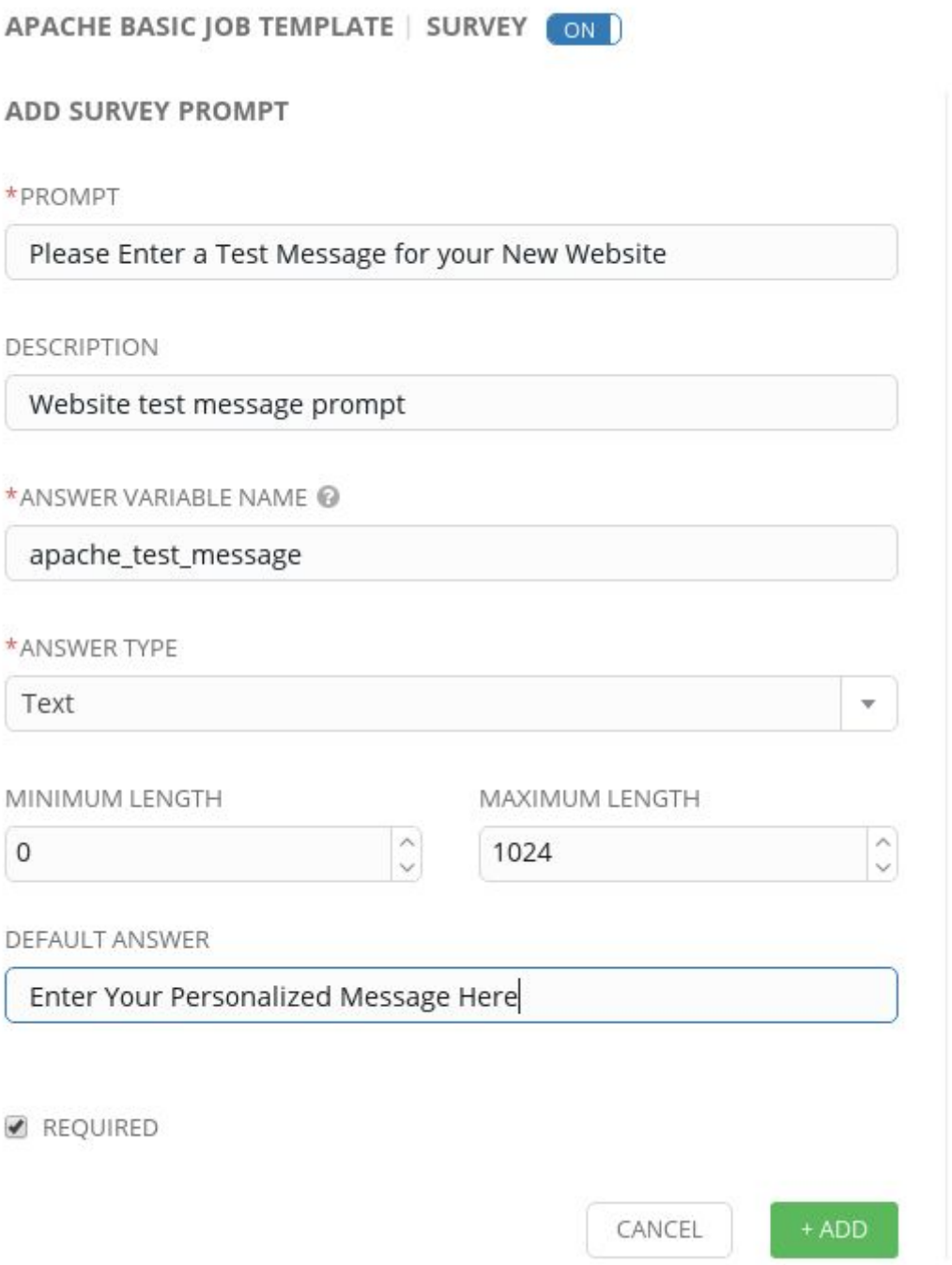


Figure 12: Survey Form

Step 6: Add the survey input

Select ADD +

Step 7: Save

Select SAVE SAVE

Step 8: Save at the main page

Back on the main Job Template page, select SAVE SAVE again.

Section 2: Running a job template

Now that you've sucessfully created your Job Template, you are ready to launch it. You will be redirected to a job screen, which refreshes in realtime and displays the status of the job.

Step 1: Select the Template tab

Select TEMPLATES

> Alternatively, if you haven't navigated away from the job templates creation page, you can scroll down to see all existing job templates

Step 2: Access Apache Basic Job Template

Click the rocketship icon for the `Apache Basic Job Template`

Step 3: Enter test message

When prompted, enter your desired test message

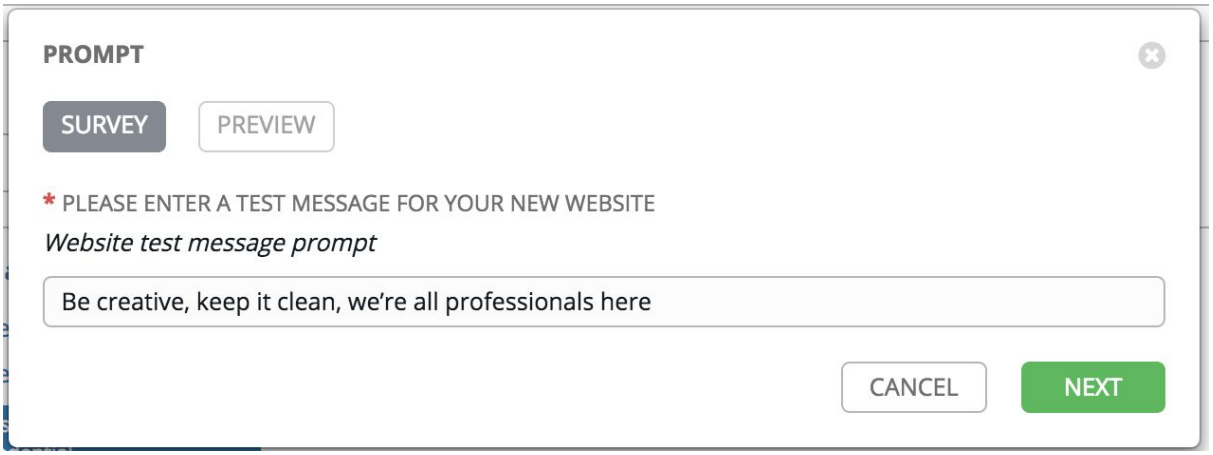


Figure 13: Survey Prompt

Step 4: Launch the job

Select LAUNCH

Step 5: View job summary

Sit back, watch the magic happen

One of the first things you will notice is the RESULTS section. This section provides job details, such as who launched it, what playbook it's running, what the status is, i.e. Pending, Running, or Complete.

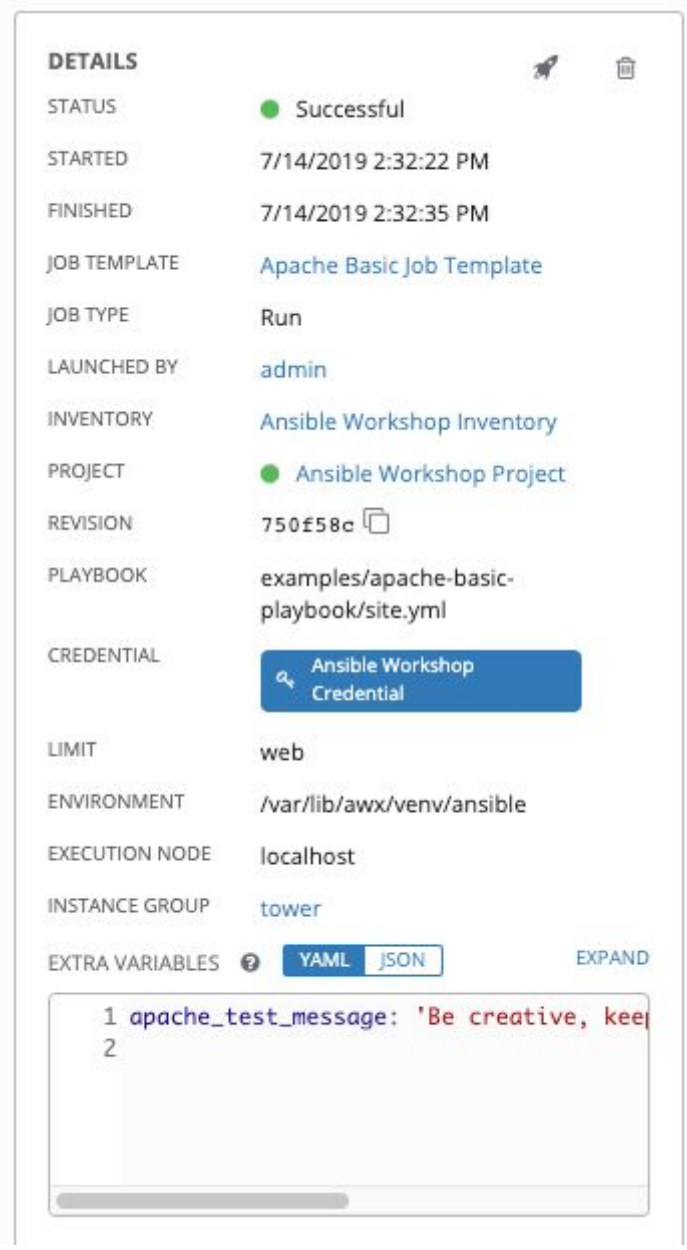


Figure 14: Job Summary

To the right, you can view standard output; the same way you could if you were running Ansible Core from the command line.

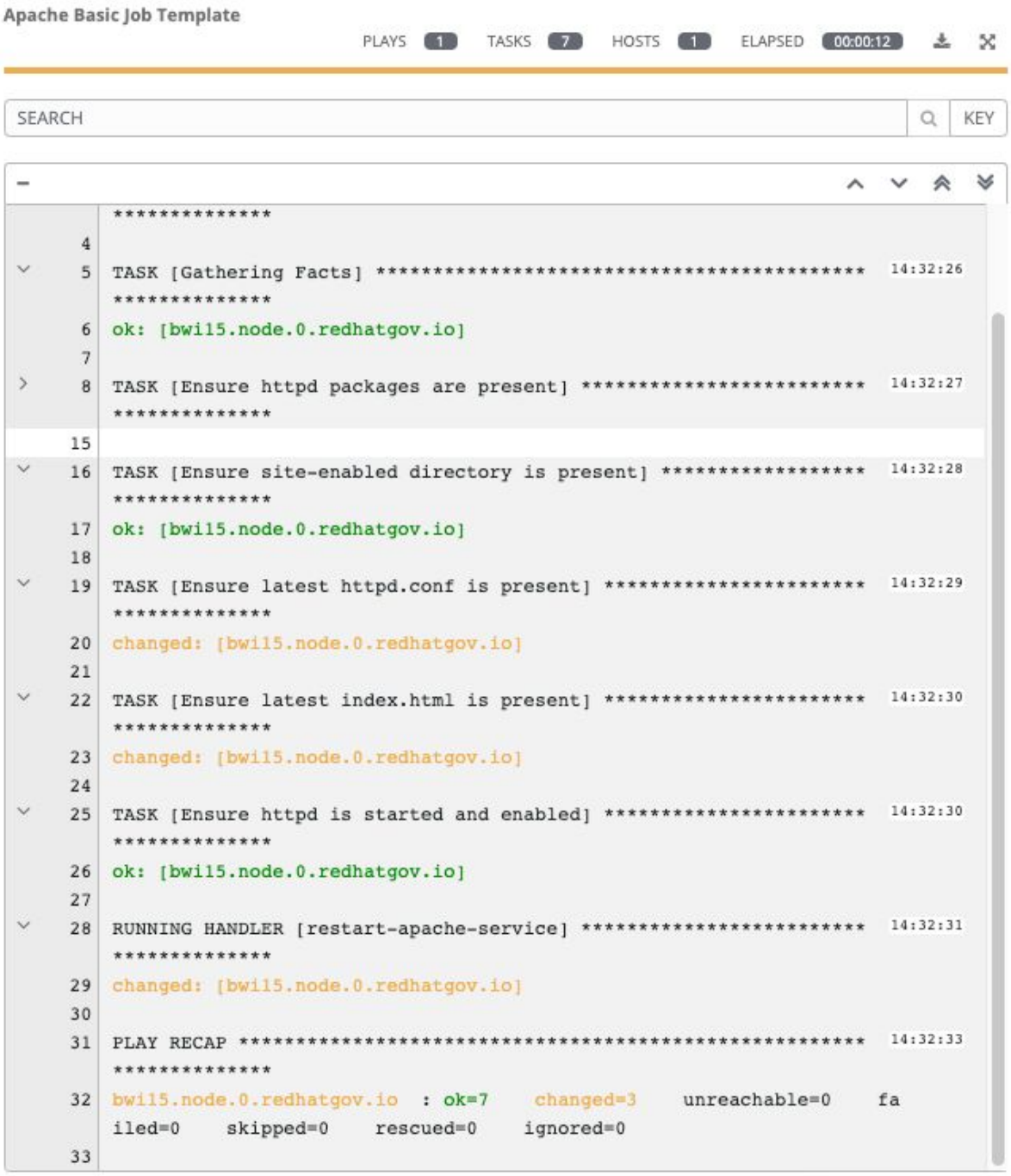


Figure 16: Job Standard Output

Step 6: View the new website

Once your job is sucessful, navigate to your new website, where `workshopname` is the name of your workshop, and # is your student number:

`http://example.node.0.redhatgov.io`

If all went well, you should see something like this, but with your own custom message:

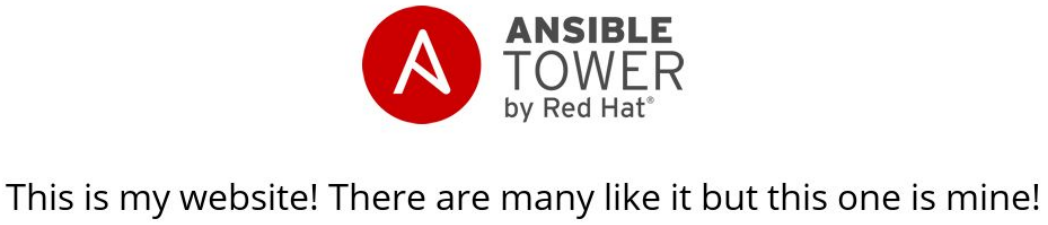


Figure 17: New Website with Personalized Test Message

End Result

At this point in the workshop, you've experienced the core functionality of Ansible Tower. But wait… there's more! You've just begun to explore the possibilities of Ansible Core and Tower. Take a look at the resources page in this guide, to explore some more features.

Exercise 3.0 - Using Ansible to Implement Security

Exercise Description

In this exercise, we are going to use Red Hat Ansible Tower to run a `DISA STIG evaluation` of our environment.

- DISA STIG controls https://galaxy.ansible.com/redhatofficial/rhel7_stig

Step 1: Download role to Ansible roles directory

In your wetty window (if you closed it, see the SETUP step, in your workbook), type the following:

```
sudo ansible-galaxy install redhatofficial.rhel7_stig -p
/etc/ansible/roles
```

The image below illustrates that the role has been downloaded to your system-wide Ansible roles directory, `/etc/ansible/roles`:

```
- downloading role 'rhel7_stig', owned by redhatofficial
- downloading role from
https://github.com/RedHatOfficial/ansible-role-rhel7-stig/archive/0.
1.44.tar.gz
- extracting redhatofficial.rhel7_stig to
/etc/ansible/roles/redhatofficial.rhel7_stig
- redhatofficial.rhel7_stig (0.1.44) was installed successfully
```

Step 2: Select Projects

Click the Projects tab, in the Ansible Tower UI. 

Step 3: Click Add

Next, Select 

Step 4: Complete the Project form

Complete the project form, using the following entries:

NAME	Ansible STIG Project
DESCRIPTION	STIG Role Playbook
ORGANIZATION	Default
SCM TYPE	Git
SCM URL	https://github.com/ajacocks/rhel7_disa_stig
SCM BRANCH	
SCM UPDATE OPTIONS	Clean Delete on Update Update Revision on Launch

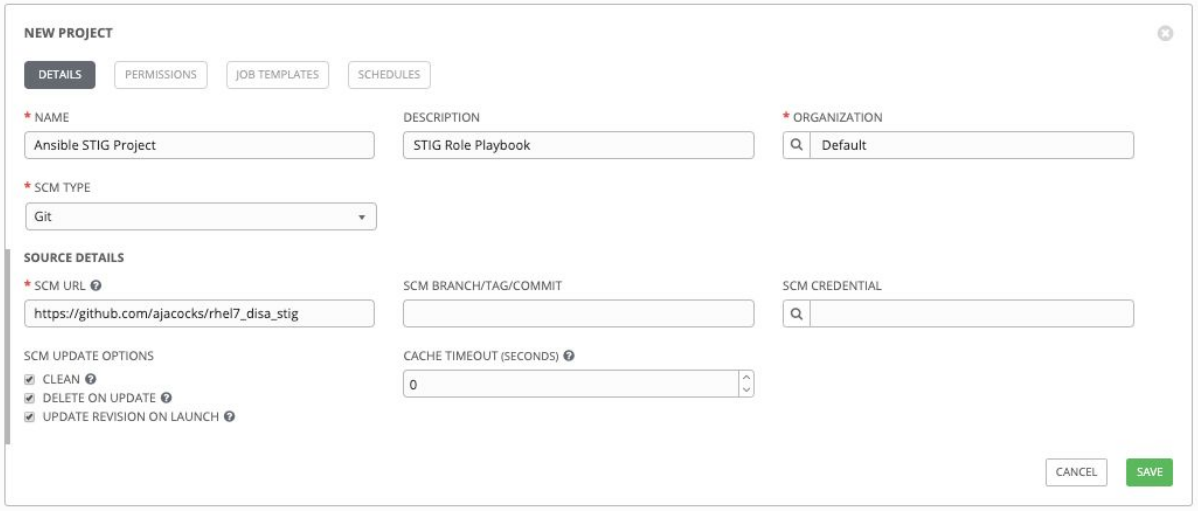


Figure 1: Defining a Project

Step 5: Save

Select SAVE 

Step 6: Select Template tab

In your Tower window, click `TEMPLATES`

Step 7: Add the job template

Click ADD 

, and select `JOB TEMPLATE`

Step 8: Complete the job Template form

Complete the form using the following values. Note that the `PLAYBOOK` field should offer `main.yml` as an option, when clicked.

NAME	STIG Job Template
DESCRIPTION	Template for security playbooks
JOB TYPE	Run
INVENTORY	Ansible Workshop Inventory
PROJECT	Ansible STIG Project
PLAYBOOK	main.yml
MACHINE CREDENTIAL	Ansible Workshop Credential
LIMIT	web

SKIP TAGS	• CCE-27361-5 • CCE-27485-2 • CCE-27311-0 • CCE-80546-5 • CCE-80998-8 • CCE-80224-9 • CCE-80226-4 • CCE-27413-4 • CCE-27445-6 • CCE-80221-5 • CCE-27433-2 • CCE-80222-3 • CCE-80220-7 • CCE-80223-1 • CCE-27455-5 • CCE-80436-9 • CCE-80240-5 • CCE-27458-9
OPTIONS	Enable Privilege Escalation

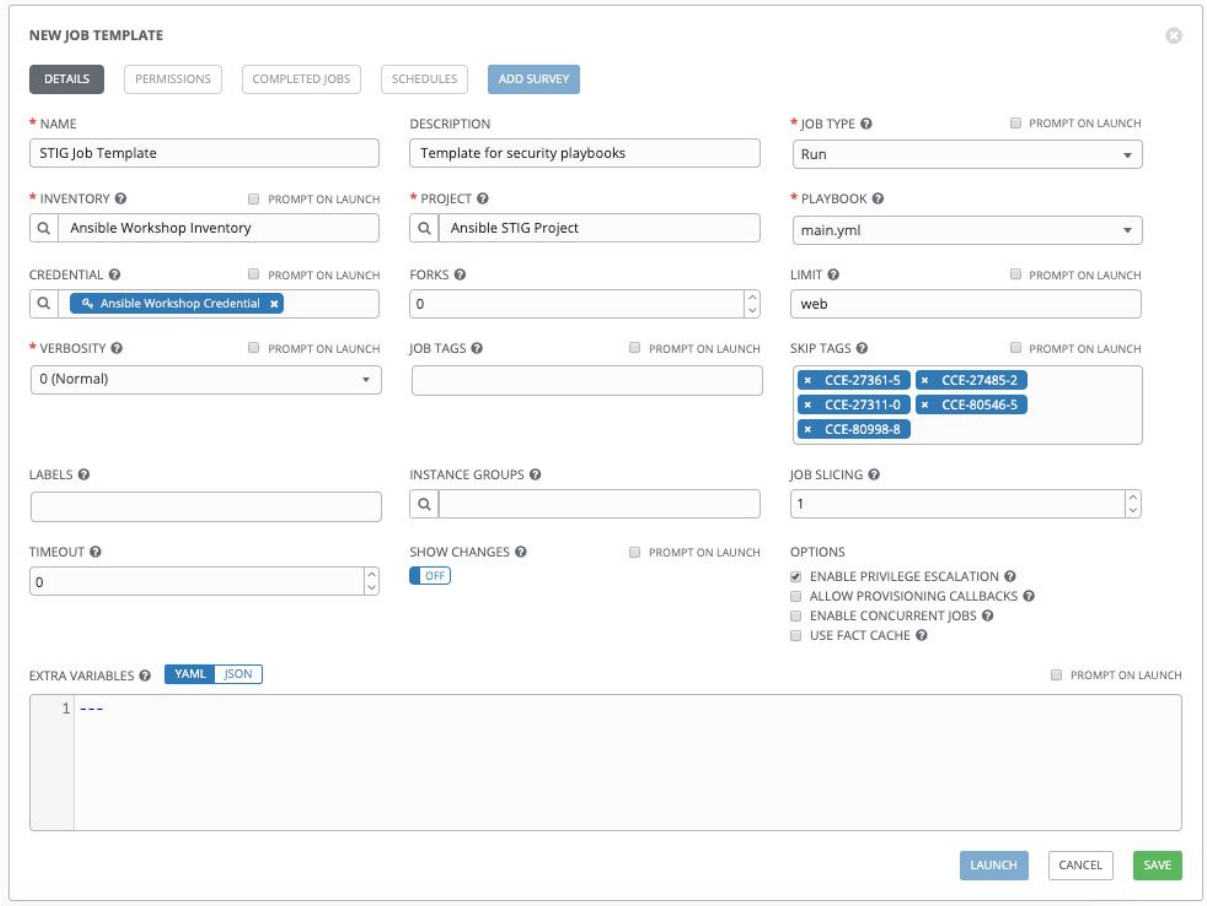


Figure 2: Defining a Job

Step 9: Save the template and run it

Click SAVE 

, to store your new template, and we are ready to run it.

Click the rocketship icon  next to the `STIG Job Template` entry, to launch the job.

View what the job looks like as it is executing, as well as what the SCAP results look like, when uploaded to your second node, in the panel, below.

Job Status

Results

Step 10: Observe the scanning process and view reports

You can watch the scan run against your managed node. Note that each compliance check is named and detailed.

Once the check is complete, you can open a new tab in your web browser, and navigate to the following URL, where `workshopname` is the workshop prefix, and `#` is the number that your instructor gave you:

`http://example.node.0.redhatgov.io/scap`

Click the link called `scan-xccdf-report-…` to review the SCAP report that was generated. Note the failures in the report; look at the machines, if you want, via your Wetty ssh session, to analyze what the problems might be.

Wrapup

That wraps up what we have planned for today. We hope you've learned something valuable about Ansible and Red Hat Ansible Tower that you can apply in your daily role.

What do you think? How can we help you understand Ansible Tower better?

Before you leave, check out the `Resources` page that is part of this guide. There, you can find a ton of links that will further your Ansible education.

This `Participant Guide` will remain active for the next two weeks. Please take advantage of it by creating your own Ansible and Ansible Tower environment back at your organization.

Thank you for your time and participation!