

Lab: Ansible AWX Web UI

Overview

Ansible AWX provides a Graphical User Interface for executing Ansible playbooks. It also has many other features around access control, job scheduling, and viewing job history.

In this lab, we will run AWX on our local workspace and then add the inventory and playbook we have developed so far. Then execute it from the interface.

[Overview](#)

[Instructions](#)

[Task 0: Install AWX](#)

[Step 1: Install Prerequisites](#)

[Step 2: Install AWX \(with Ansible!\)](#)

[Step 3: View AWX](#)

[Task 1: Create a Job Template](#)

[Step 1: Update the demo project](#)

[Step 2: Add our inventory](#)

[Step 3: Run ad hoc commands](#)

[Step 4: Upload your playbook](#)

[Step 5: create a template that deploys the db portion](#)

Lab: Ansible AWX Web UI

Instructions

Read this lab like a book, all text is there for a reason!

"→" denotes an action you must take

Use your favorite editor to edit files within the console. I suggest VI, nano, or emacs.

White boxes with black text denote commands and file contents

Black boxes with green text denote example output

Task 0: Install AWX

Step 1: Install Prerequisites

The ansible playbook for AWX will not install some of the required dependencies. The below script should prepare your workspace.

→ Execute the following

```
sudo amazon-linux-extras install docker
sudo service docker start
sudo usermod -a -G docker $(whoami)
exec su -l $USER
sudo yum install -y libffi-devel
pip install -y --user docker docker-compose
```

Step 2: Install AWX (with Ansible!)

→ Clone AWX

```
# Go to your home directory
cd ~

# Download awx
git clone https://github.com/ansible/awx
cd ansible/awx
```

→ Update inventory, change the awx_alternate_dns_servers line to match the following.

This will set AWX to use the google DNS servers so it can resolve websites like github.com.

```
awx_alternate_dns_servers="8.8.8.8,8.8.4.4"
```

Lab: Ansible AWX Web UI

→ Run the playbook

```
ansible-playbook -i inventory install.yml
```

Step 3: View AWX

→ Go to <http://localhost> in your browser

The default administrator is 'admin' with the password 'password'

Task 1: Create a Job Template

Step 1: Update the demo project

To ensure everything is set up correctly, take a look at the Demo Project under Projects in the navigation menu. This project is a link to the "https://github.com/ansible/ansible-tower-samples" git repository. You can use projects to allow AWX to clone repositories for use when creating inventories or job templates. For example, your playbook may live in a git repository and you simply want AWX to refer to that instead of maintaining its own copy.

→ Click the  "Get latest SCM revision" button for Demo Project

This will cause a job to run that will download the latest for the ansible-tower-samples project

→ Go to Jobs on the menu

You should see a Job named "Demo Project" with a green circle.

→ Click the latest job and view its output

Step 2: Add our inventory


→ Go to Inventories ->



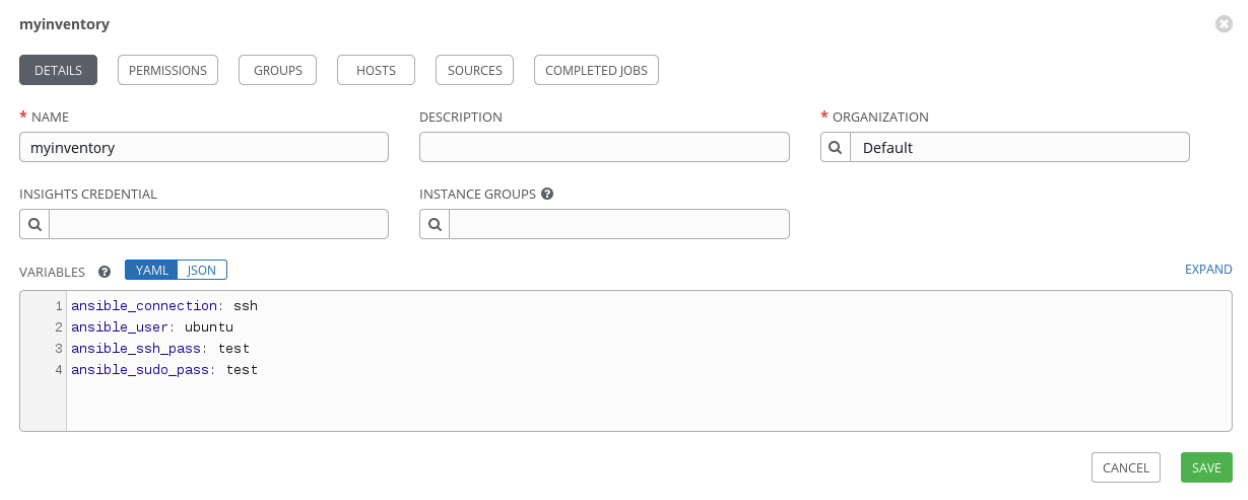
→ Give your inventory a name

Lab: Ansible AWX Web UI

→ Create two groups, "web" and "db", just like we did in our inventory file

→ For both of our hosts, use the  button under the hosts tab to add their IPs, also use the groups tab to set their correct groups.

→ Add your ansible connection variables under variables for the inventory



The screenshot shows the 'myinventory' page in the Ansible AWX Web UI. The 'VARIABLES' tab is selected, showing a list of variables in a YAML format:

```
1 ansible_connection: ssh
2 ansible_user: ubuntu
3 ansible_ssh_pass: test
4 ansible_sudo_pass: test
```

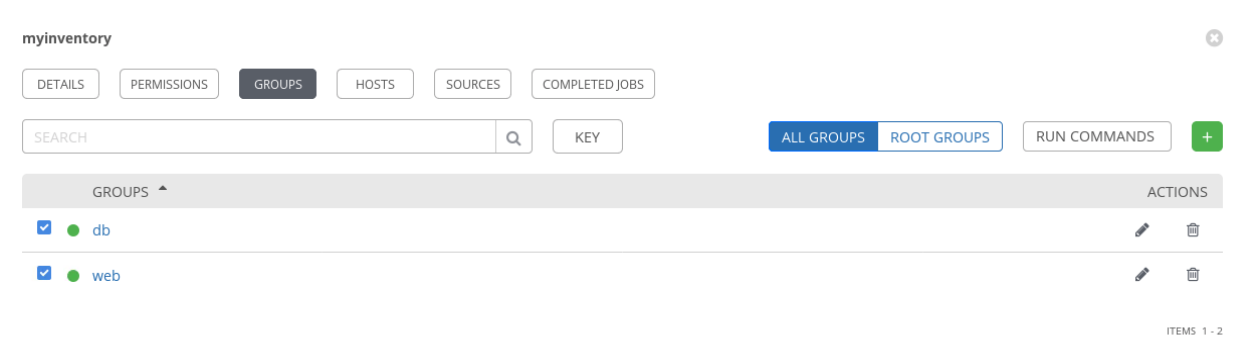
At the bottom right, there are 'CANCEL' and 'SAVE' buttons.

Step 3: Run ad hoc commands





You can run ad hoc commands via AWX just like using the *ansible* command, but it is a little hard to find.

→ Inventories → Groups tab

→ Check the boxes for both groups



The screenshot shows the 'myinventory' page in the Ansible AWX Web UI, with the 'GROUPS' tab selected. It displays a table of groups:

GROUPS	ACTIONS
<input checked="" type="checkbox"/> db	 
<input checked="" type="checkbox"/> web	 

At the top right, there is a 'RUN COMMANDS' button with a green plus icon. At the bottom right, it says 'ITEMS 1 - 2'.

→ Click the RUN COMMANDS button in the top right

→ Choose "ping" as the module and click LAUNCH

This will create a new job running a single tasks for all selected hosts. Try it again but select DEBUG=3. You will see a lot more information in the Job log.

Lab: Ansible AWX Web UI

Step 4: Upload your playbook

****STOP HERE**** Ask your instructor where and how to put your playbook in a repository

→ Go back to "Projects"

→ Click the green plus sign to add a project

→ Ensure you have committed all of the changes in your playbook and pushed it to a repository

→ Put in the repository information

Step 5: Create a template that deploys the db portion

→ Templates →  → Select "Job Template" in the dropdown

→ Update the following values. For inventory, use the one you created.

TEMPLATES / Deploy database

Deploy database

DETAILS PERMISSIONS NOTIFICATIONS COMPLETED JOBS SCHEDULES ADD SURVEY

* NAME	DESCRIPTION	* JOB TYPE ?	<input type="checkbox"/> PROMPT ON LAUNCH
Deploy database		Run	
* INVENTORY ?	* PROJECT ?	* PLAYBOOK ?	
<input type="text" value="myinventory"/>	<input type="text" value="Guestbook"/>	<input type="text" value="playbook.yml"/>	
CREDENTIAL ?	FORKS ?	LIMIT ?	<input type="checkbox"/> PROMPT ON LAUNCH
<input type="text"/>	0	db	
* VERBOSITY ?	JOB TAGS ?	SKIP TAGS ?	<input type="checkbox"/> PROMPT ON LAUNCH
0 (Normal)			
LABELS ?	INSTANCE GROUPS ?	JOB SLICING ?	
	<input type="text"/>	1	
TIMEOUT ?	SHOW CHANGES ?	OPTIONS	
0	<input type="checkbox"/> OFF	<input checked="" type="checkbox"/> ENABLE PRIVILEGE ESCALATION ?	
		<input type="checkbox"/> ALLOW PROVISIONING CALLBACKS ?	
		<input type="checkbox"/> ENABLE CONCURRENT JOBS ?	
		<input type="checkbox"/> USE FACT CACHE ?	
EXTRA VARIABLES ?		<input type="checkbox"/> PROMPT ON LAUNCH	
<div><div>YAML JSON</div><div><pre>1 ---</pre></div></div>			

LAUNCH CANCEL SAVE

Lab: Ansible AWX Web UI

In the Job output, you should see some familiar text begin to appear as the job runs