**Lesson 06 Demo 5**

**Creating and working with Ansible roles**

**Objective:** To create and work with Ansible Roles.

**Pre-requisites:** You need to have Ansible installed in order to proceed with this demo. If you don’t have it installed, refer to demo 1 of lesson 7.

**Tools required:** Ansible.

**Steps to be performed:**

1. Install Ansible on Ubuntu, and establish connectivity between Ansible controller and node machine (You can skip this step in case Ansible is already installed)

2. Create Ansible Role

3. Create Ansible tasks

4. Create Ansible template

5. Create Ansible variable

6. Remove unwanted directory

7. Create Ansible role playbook

8. Deploy Ansible role playbook

**Step 1: Install Ansible on the Ubuntu, and establish connectivity between Ansible controller and node machine**

1.1 Use the below commands on Ubuntu system to install ansible software

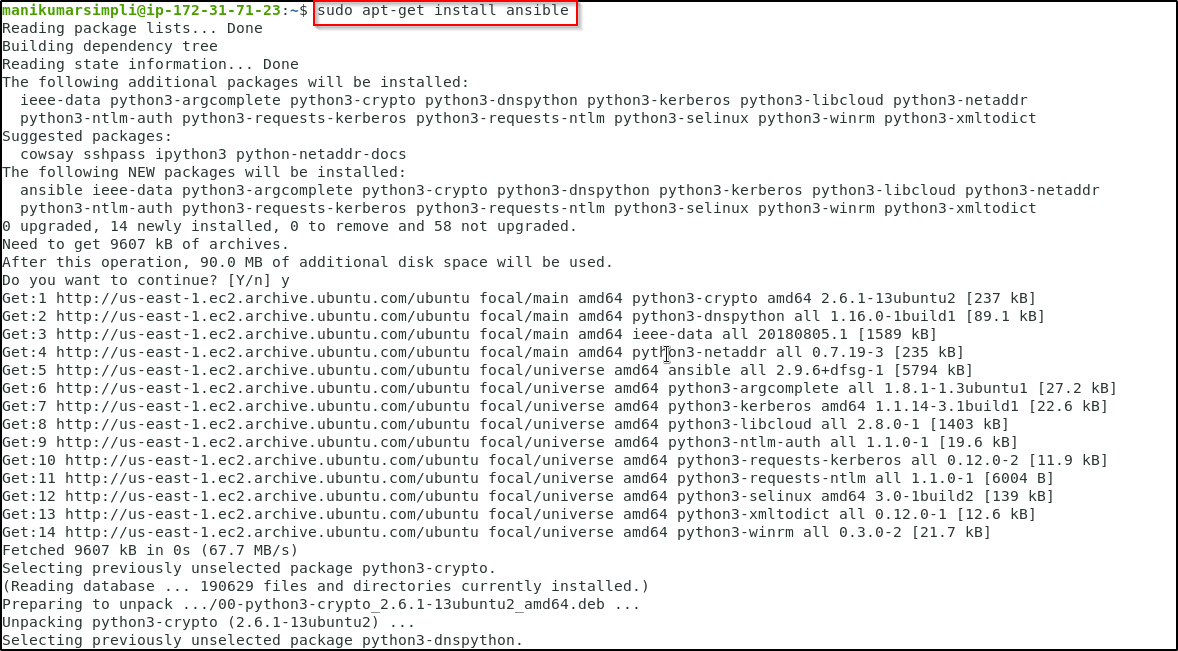
***sudo apt-get install -f***

***sudo apt-get install software-properties-common***

***sudo apt-add-repository ppa:ansible/ansible***

***sudo apt-get update***

***sudo apt-get install ansible***

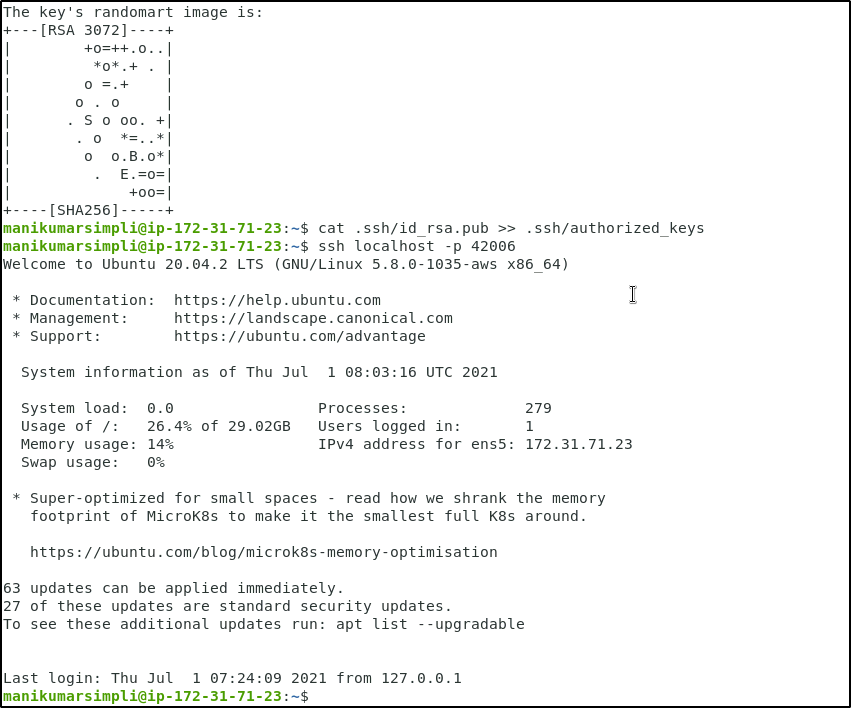


1.2 Establish SSH key pair in Linux system to establish SSH connectivity with the localhost using the following commands:

***ssh-keygen -t rsa***

***cat .ssh/id\_rsa.pub >> .ssh/authorized\_keys***

***ssh localhost -p 42006***



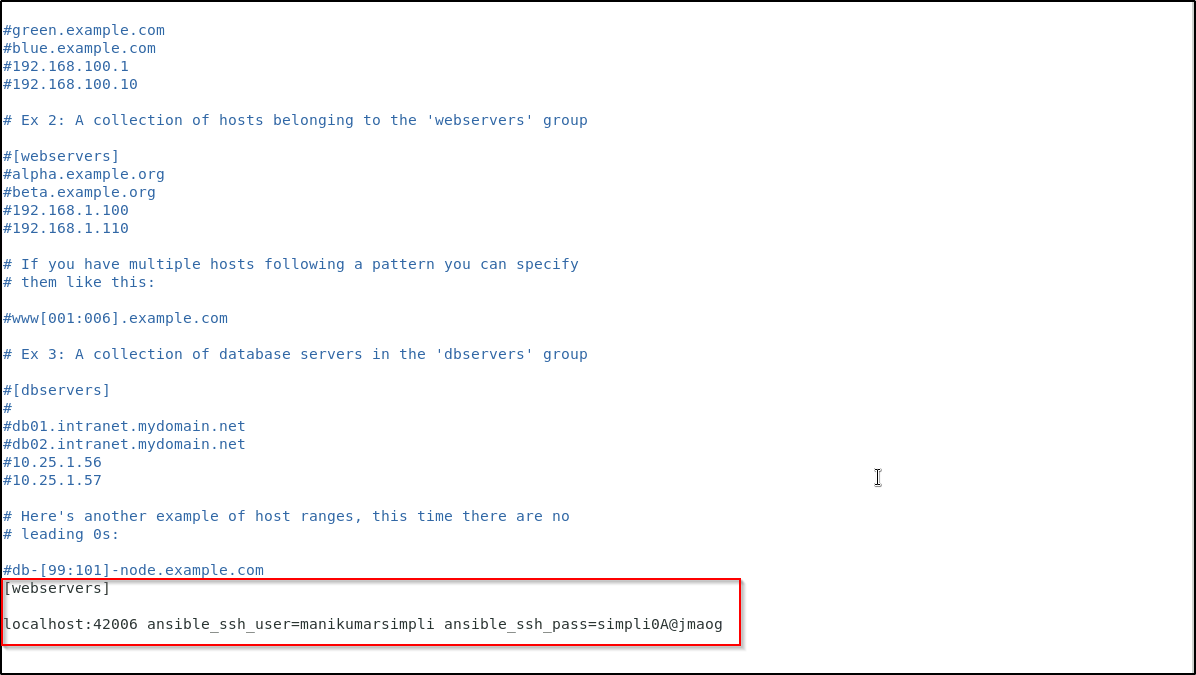
1.3 Now, add the host localhost in ansible host file ***/etc/ansible/hosts***

***sudo vi /etc/ansible/hosts***

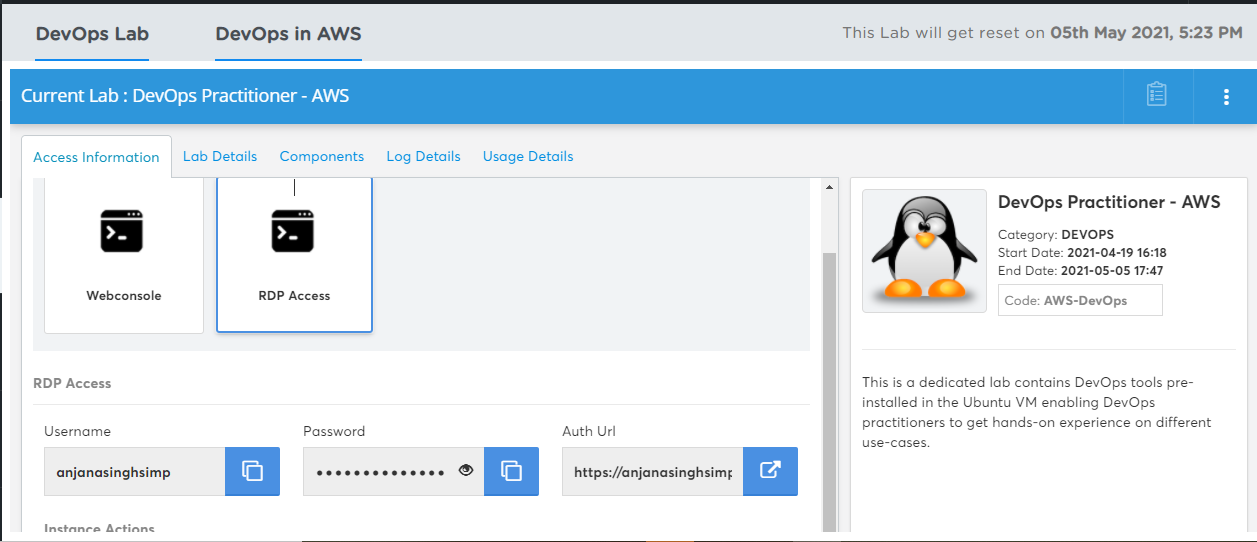
1.4 When the file opens, add the below lines of code under [webservers]:

***[webservers]***

***localhost:42006 ansible\_ssh\_user=<Your user name> ansible\_ssh\_pass=<Your Passowrd>***

****

**Note**: Add the username and password of the lab you are using. You can access those using the RDP access tab.



The reason to explicitly add username and password is - in the main yaml file, the owner is mentioned as root. This requires a master user password to authenticate it. Once added in the format, just run ansible commands via the same provided user.

You can verify this by using the below commands:

***sudo su***

***nano /etc/sudoers***

1.5 Save and exit the file using **ctrl+x**

**Step 2: Create Ansible Role**

2.1 Once we have our Ansible environment ready, next create a project. I will create a

new project base to demonstrate Ansible roles example

***mkdir base***

2.2 To create an ansible role, use ansible-galaxy init <role\_name> to create the role

directory structure

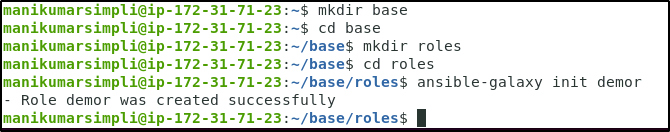
2.3 Let’s create the role in our <project>/roles directory i.e. ~/base/roles/motd

***cd base***

***mkdir roles***

***cd roles***

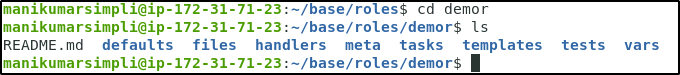
***ansible-galaxy init demor***



2.4 You can use the ***ls*** command to list the ansible role directory structure

***cd demor***

***ls***



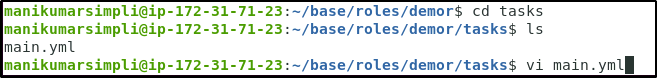
**Step 3: Create Ansible Tasks**

3.1 Now update the /etc/motd file using ansible playbook roles. Create tasks to use the main.yml file present inside the tasks folder.

***cd tasks***

***ls***

***vi main.yml***



3.2 Enter the below code

*---*

***# tasks file for demor***

***- name: copy demor file***

***template:***

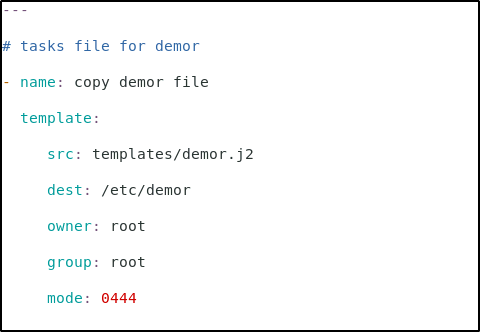
***src: templates/demor.j2***

***dest: /etc/demor***

***owner: root***

***group: root***

***mode: 0444***



**Step 4: Create Ansible Template**

4.1 Create the template content which will be used to update /etc/motd in our ansible roles examples. I will create a new template file under the templates directory using some variables:

***cd ..***

***cd templates***

***vi demor.j2***



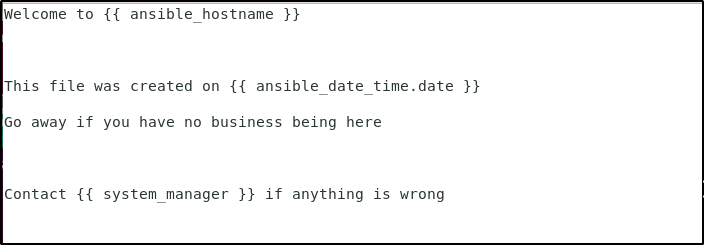
4.2 Enter the below details:

***Welcome to {{ ansible\_hostname }}***

***This file was created on {{ ansible\_date\_time.date }}***

***Go away if you have no business being here***

***Contact {{ system\_manager }} if anything is wrong***



**Step 5: Create Ansible Variable**

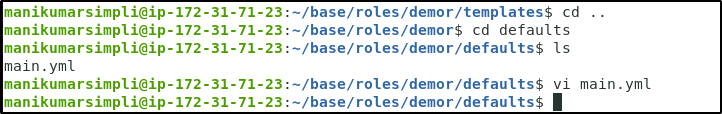
5.1 We will use the defaults folder to define custom variables which are used in our template file templates/demor.j2

***cd ..***

***cd defaults***

***ls***

***vi main.yml***

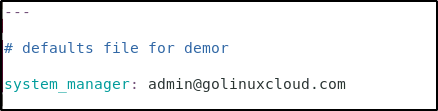


5.2 Enter the below details in the file:

***---***

***# defaults file for demor***

***system\_manager:*** [***admin@golinuxcloud.com***](mailto:admin@golinuxcloud.com)



**Step 6: Remove unwanted directories (Optional)**

6.1 This step is completely optional. In this ansible roles example, we will not use other directories so we are deleting them. After deleting the additional directories you can use the tree command to list the directory structure of motd roles

***cd ..***

***rm -rf handlers tests vars***



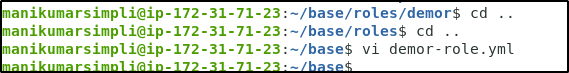
**Step 7: Create an ansible-role playbook**

7.1 Now after you create an ansible role structure, we need a playbook file that will deploy the role to our managed hosts. I will create my playbook file demor-role.yml under base project directory

***cd ..***

***cd ..***

***sudo vi demor-role.yml***



7.2 Enter the below code in the file

***---***

***- name: use demor role playbook***

***hosts: webservers***

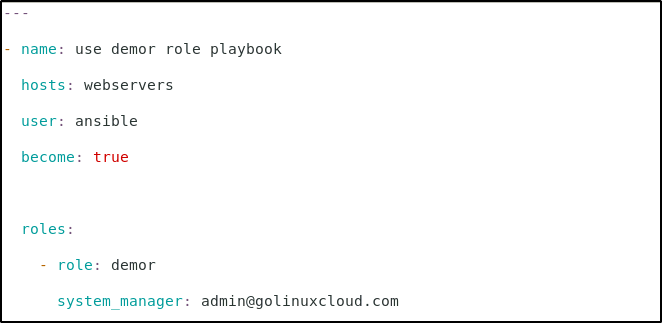
***user: ansible***

***become: true***

***roles:***

***- role: demor***

***system\_manager: admin@golinuxcloud.com***



**Step 8: Deploy Ansible role playbook**

8.1 Execute the below command:

***ansible-playbook demor-role.yml***

**Note:** In case you get an error asking you to install sshpass program as shown below, execute the command: ***sudo apt install sshpass*** and then run the ***ansible-playbook demor-role.yml*** command.

