**To install Ansible:**

$ sudo apt update

$ sudo apt install software-properties-common

$ sudo add-apt-repository --yes --update ppa:ansible/ansible

$ sudo apt install ansible

sudo apt-get install net-tools (for IP address command)

sudo apt-get install nmap (to check port status)

sudo apt install openssh-server (to install ssh server)

|  |  |
| --- | --- |
| **COMMANDS** | **DESCRIPTION** |
| ifconfig | To get IP-address |
| nmap 192.168.0.110 -p 22 | To check connectivity |
| sudo systemctl status ssh | To check status of ssh protocol |
| sudo gedit /*etc/*ansible/hosts | To confirm installation of ansible (inventory file: hosts) |
| ssh-keygen | To generate public private key |
| ssh-copy-id username@ipaddr **(or)** ssh copy-id ipaddr | To copy the generated key to node machine |
| In node machine:  cd *home/ashwini/.ssh*  *ls* | To verify key copied from master machine  (authorized\_keys) |
| ansible all -m ping (before this add ipaddress of nodes in default inventory file -hosts) | To verify master is connected to all node machines |
|  |  |
| **Adhoc commands:**  Syntax:  ansible host-pattern -m module [-a ‘module arguments’ -i inventory file] |  |
| ansible prod -m copy -a ‘src=/etc/ansible/hosts dest=/tmp’ | Copies file hosts from master to node machine |
| ansible -i myInventory.txt -m setup thbs\_solutions | It shows extra info in node machines listed in host group:thbs\_solutions in myInventory.txt |
| ansible thbs\_solutions -m shell -a ‘df -h’ -i myInventory.txt | It shows disk free space in human readable format |
| ansible thbs\_solutions -m apt -a ‘name=apache2 state=present’ -b | To install apache2 in node machines |
| ansible -m file -a ‘path=/tmp/ashwini.txt state=touch’ | To create file named ashwini.txt in node machines(state=absent removes file) |
| ansible-playbook playbook\_name -K | To run playbook |
| ansible-vault create playbook | To encrypt new file |
| ansible-vault encrypt playbook | To encrypt existing file |
| ansible-vault decrypt playbook | To decrypt existing file |
| ansible-vault rekey playbook | To change encrypted password |
| Ansible-playbook --vault-id @prompt filename -K | To run the encrypted playbook by asking vault password |
| ansible-vault view playbook | this means that the contents are displayed in the terminal. Pass the vault encrypted file to the command: |
| ansible-galaxy init rolename | To create roles file structure |

**Playbook example:**

---

- name: update packages and to install git and apache2, update sample webpage and restart service

hosts: all

become: yes

tasks:

- name: updating all packages to latest versions

apt:

name: '\*'

state: latest

- name: installing git

apt:

name: git

state: present

update\_cache: yes

- name: installing apache2

apt:

name: apache2

state: present

- name: copy index file

copy:

src: 'index.html'

dest: '/var/www/html/index.html'

force: yes

- name: restart

shell: ls -la

notify: restart service

handlers:--------------------->exact level of tasks

- name: restart service

service:

name: apache2

state: restarted

**Ansible Roles:** Primary mechanism to breakdown playbook into multiple files.

**Folders:**

defaults: main.yml file contains default values of roles (vars block to override default variables)

files: static files referenced by roles

handlers: main.yml contains roles handler def

meta: main.yml contains info abt roles including author, license, platforms and optional roles dependencies

tasks: main.yml contains roles task defaults

template: contains jinja2 template engine

**STEPS:**

mkdir test-playbook create directory