INVENTORY DOCUMENTATION FOR ANSIBLE:

<https://docs.ansible.com/ansible/latest/inventory_guide/intro_inventory.html>

ADHOC COMMANDS FOR ANSIBLE

<https://docs.ansible.com/ansible/latest/command_guide/intro_adhoc.html>

**INSTALLING ANSIBLE:**

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

#sudo apt update

#sudo apt install ansible

**SETTING UP INVENTORY FILE:**

The simplest inventory is a single file with a list of hosts and groups. The default location for this file is **/etc/ansible/hosts**. You can specify a different inventory file at the command line using the

**-i <path>** option or in configuration using inventory.

Inventory can be written in three formats:

* INI
* JSON
* YAML

Generate RSA SSH private key of the machine.

Make Inventory as you want.

Sample Inventory:

**web01 ansible\_host=172.31.23.137**

**web02 ansible\_host=172.31.17.193**

**db01 ansible\_host=172.31.19.79**

**[websrvgrp]**

**web01**

**web02**

**[dbsrvgrp]**

**db01**

**[dc\_ohio:children]**

**websrvgrp**

**dbsrvgrp**

**[dc\_ohio:vars]**

**ansible\_user=centos**

**ansible\_ssh\_private\_key\_file=vprofile-key.pem**

**TURNING OFF HOST KEY CHECKING:**

Execute command:

#cd /etc/ansible/

**NOTE: You can make backup file of ansible.cfg for safe side**

#ansible-config init --disabled -t all > ansible.cfg

#vi ansible.cfg

Then turn:

host\_key\_checking=False

:wq (save and exit the file )

**CHANGE PERMISSION OF SSH KEY:**

#Chmod 400 clientkey.pem

(You have to write the key name you have generated)

**COMMAND FOR PING:**

Make sure to be in the path at which your inventory is.

#ansible all -m ping -i inventory

# ansible web01 -m ping -i inventory

**COMMAND FOR MANAGING USERS:**

#ansible all -m ansible.builtin.user -a "name=foo password=<crypted password here>"

(Making a user)

#ansible all -m ansible.builtin.user -a "name=foo state=absent"

(Deleting a user)

**MANAGING SERVICE:**

ansible -i inventory -m yum -a "name=httpd state=present" web01 –become

ansible -i inventory -m service -a "name=httpd state=started enabled=yes" web01 –become

ansible webservers -m ansible.builtin.service -a "name=httpd state=restarted"

ansible webservers -m ansible.builtin.service -a "name=httpd state=stopped"

**DELETING DIRECTORY OR PATH:**

ansible webservers -m ansible.builtin.file -a "dest=/path/to/c state=absent

**COPY A FILE:**

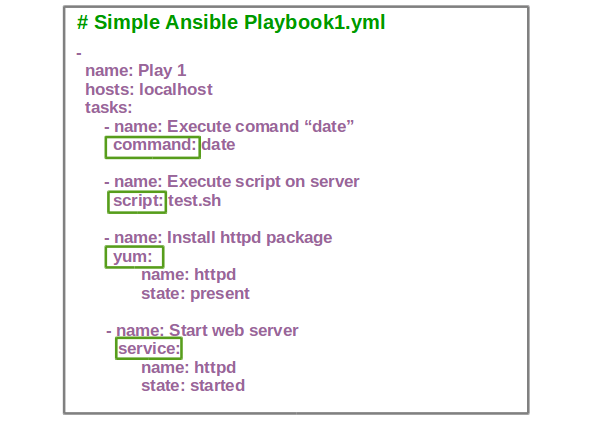
ansible -i inventory -m copy -a "src=index.html dest=/var/www/html/index.html" web01 –become

**PLAYBOOKS:**

Playbook is a list of play.

Playbooks are expressed in YAML format with a minimum of syntax.

A screenshot of a computer

Description automatically generatedSample Playbook:

**SAMPLE PLAYBOOK CODE (YAML):**

---

- name: Setup DBserver

hosts: dbsrvgrp

become: yes

tasks:

- name: Install MySQL server

yum:

name: mariadb-server

state: present

- name: Install Python MySQL

yum:

name: MySQL-python

state: present

- name: Start & Enable maridb service

service:

name: mariadb

state: started

enabled: yes

- name: Create a new database with name 'accounts'

mysql\_db:

name: accounts

state: present

- name: Create database user with name 'admin'

mysql\_user:

name: admin

password: 12345

priv: '\*.\*:ALL'

state: present

**EXECUTE PLAYBOOK:**

(TO execute playbook, you should be in the directory where inventory and playbook resides)

Command to execute:

#ansible-playbook -i inventory web-db.yaml

#ansible-playbook -i inventory (name of playbook)

#ansible-playbook -i inventory (name of playbook) -v

#ansible-playbook -i inventory (name of playbook) –syntax-check

#ansible-playbook -i inventory (name of playbook) –C

**-v flag is used for debugging.**

**-C flag is used for dry run.**

**--syntax-check is used to check syntax of playbook.**