ANSIBLE

(Configuration management tool/python dependency)

Step 1: Launch Master instance and provide required permissions.

Step 2: Instal ansible in EC2 Instance

* Command: amazon-linux-extras install ansible2 -y

Step 3: Instal Python, here python is dependency for Ansible

* Command: yum install python python-pip python-level openssl -y

Step 4: Create one ansible user

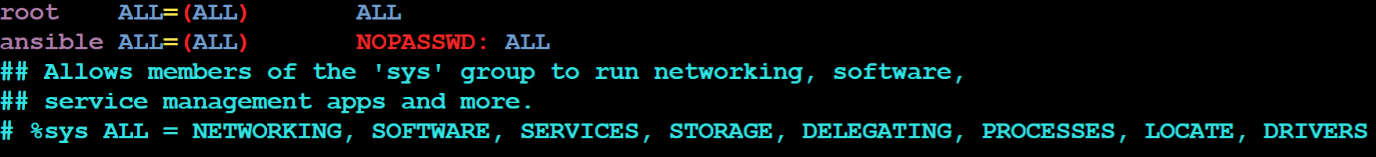
* Command: useradd ansible

Step 5: Provide password:

* Command: passwd ansible

Step 6: Provide root permissions to ansible user

* Command: visudo



Step 6: Provide root permissions to ansible user

* Command: vi /etc/ssh/sshd\_config

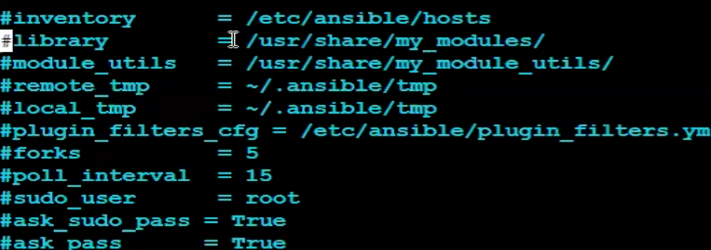
Step 7: Restart the system default SSHD

* Command: systemctl restart sshd

=============================================Do the same configurations to all slaves except ansible installation

Step 8: Create group in master for slaves open =default ansible cgf file

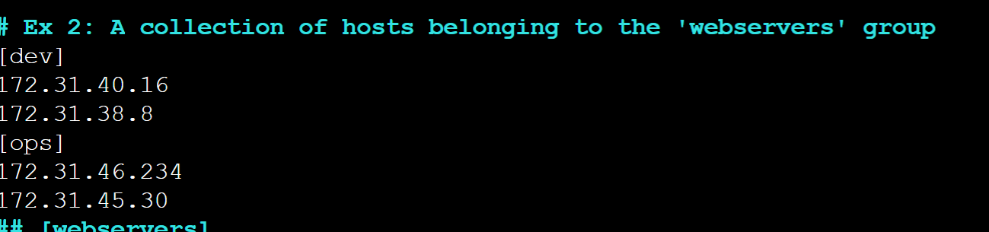
* Command: vi /etc/ansible/ansible.cfg
* Change inventory permission to





Step 9: Add slaves in host file

* Command: vi /etc/ansible/hosts



* To show all available user’s command:

ansible all - -list-host

* To show available users in specific group:

ansible (group name) - -list -host

* ansible all [2] - -list-host

Step 11: CONFIGURE MASTER AND SLAVE

Now login to master ansible and generate a SSH Key

* Login to master ansible: Command: su -ansible
* Generate SSH Key : Command : ssh-keygen
* Connect from master to salve: Command:

ssh-copy-id [bhupesh@172.31.40.16](mailto:bhupesh@172.31.40.16)

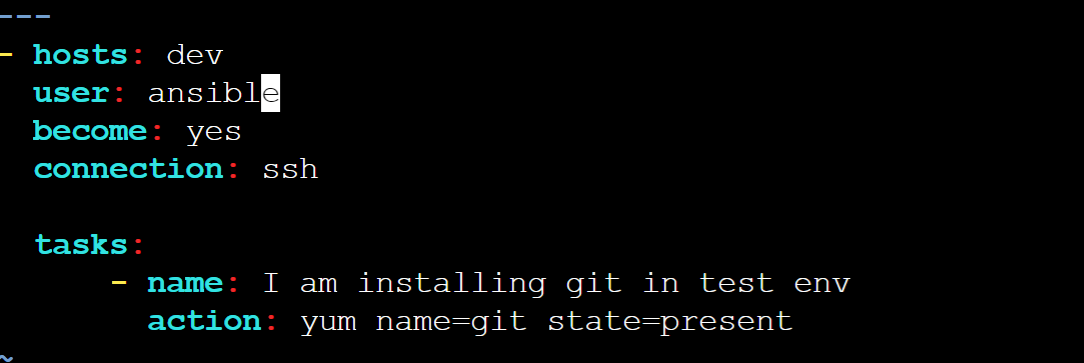
* ssh ansible@172.31.10.67

Step 12: YAML: YET ANOTHER MARKUP LANGUAGE

* yml contains target,variable and task sections
* yml file should contain. yml extension
* yml file starts with - - - & ends with . . .
* playbook states: install-present, uninstall-absent, update-latest,

restart-restarted.

* Command to execute playbooks: ansible-playbook filename.yml
* Yml Script without variable Example:



---

- hosts: dev

user: bhupesh

become: yes

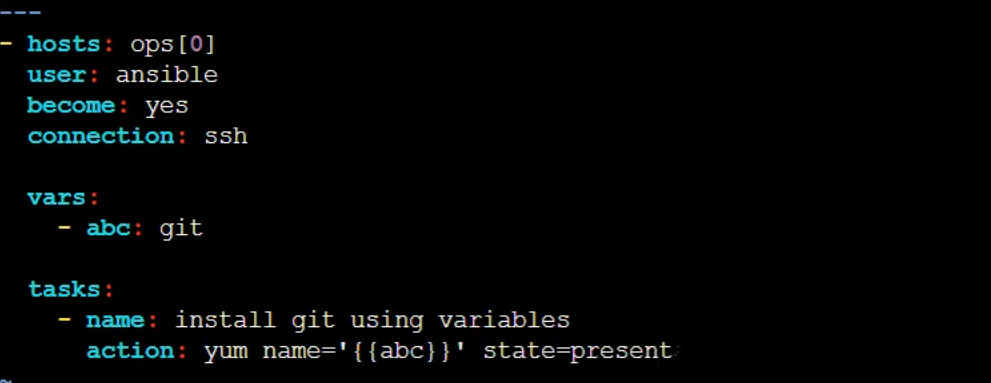
connection: ssh

tasks:

- name: i am installing git

action: yum name=git state=present

* yml script with one variable example:



---

* hosts:ops[0]

user:Mahesh

become:yes

connection:ssh

vars:

- abc: git

tasks:

-name: install git

action: yum name=’{{abc}}’ state=present

* yml script with two variables



* hosts:ops[0]

user:Mahesh

become:yes

connection:ssh

vars:

- abc: git

- bbc:maven

tasks:

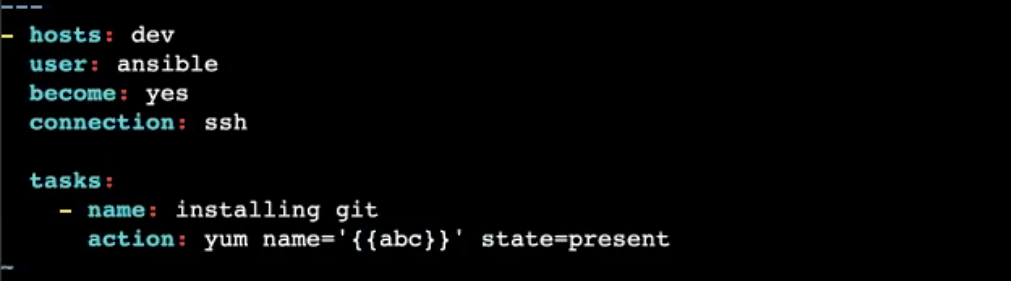
-name: install git

action: yum name=’{{abc}}’ state=present

-name: install maven

action: yum name=’{{bbc}}’ state=present

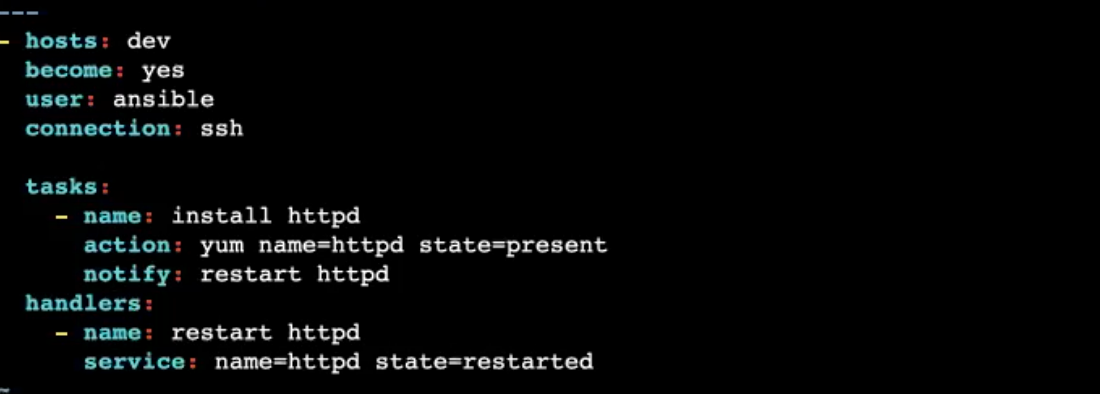
* yml script for passing variable at run time



ansible-playbook filename.yml - -extra-vars “abc=git”

ansible-playbook filename.yml - -extra-vars “abc=git bbc=maven”

* Handler’s concept (httpd server): Handlers is used send a notification between to tasks.



---

-hosts:dev

become:yes

user:bhupesh

connection:ssh

tasks:

-name:I am going to install git

action: yum name=httpd state:present

notify: restart httpd

handlers:

-name:restart httpd

service: name=httpd state=restarted



---

-hosts:dev

become:yes

user:bhupesh

connection:ssh

tasks:

-name:I am going to install git

command: yum install httpd -y

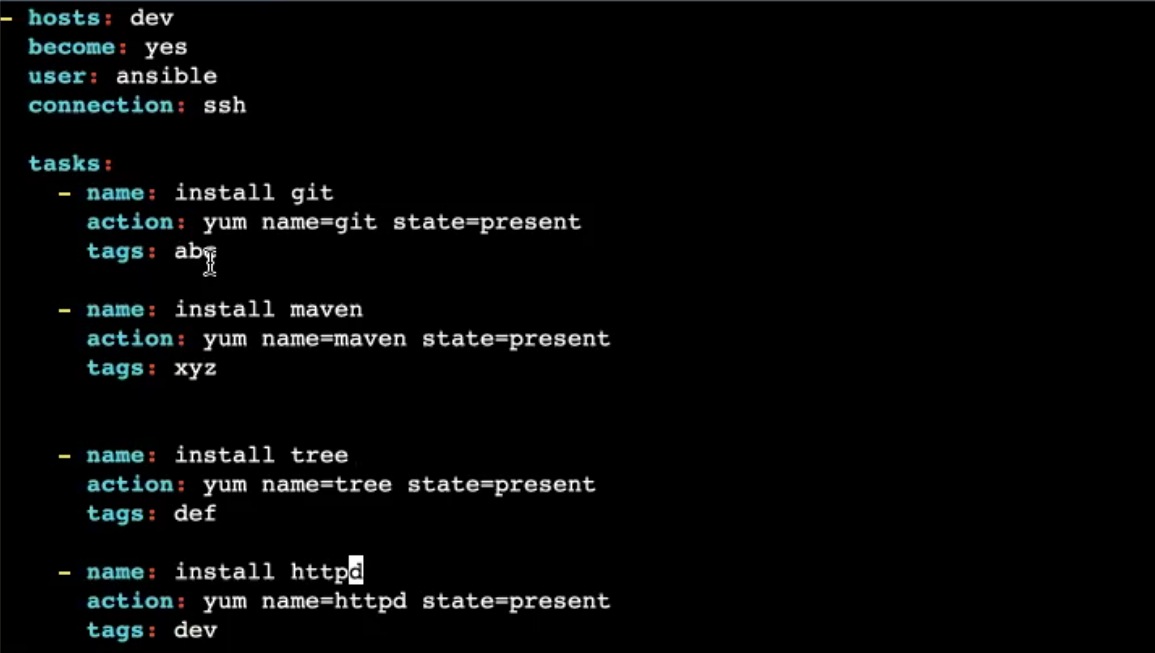
notify: restart httpd

handlers:

-name:restart httpd

Command: systemctl restart httpd

* Tags’ concept: Tags is used to skis particular task.



ansible-playbooks one.yml --skip-tags “def”

---

- hosts: dev

user: ansible

become: yes

connection: ssh

tasks:

- name: installing git

action: yum name=git state=present

tags: abc

- name: installing maven

action: yum name=maven state=present

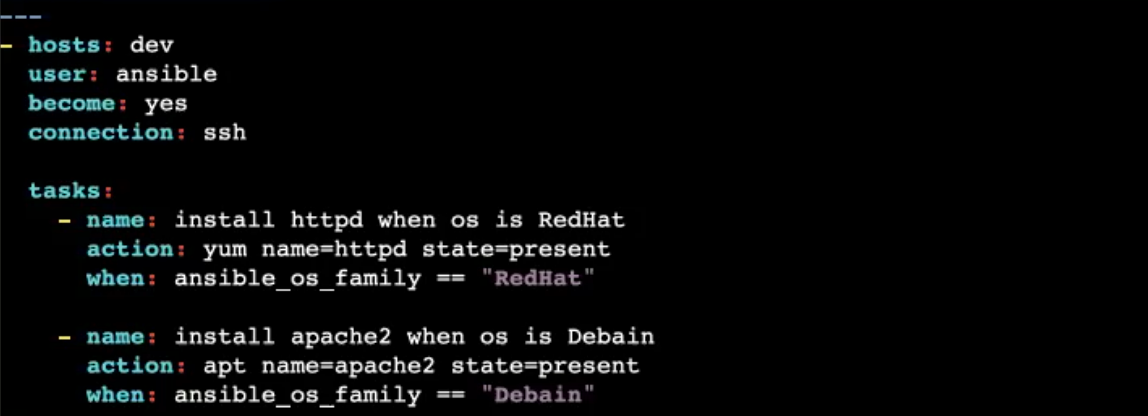
tags: bbc

- name: installing httpd

action: yum name=httpd state=present

tags: dev

* Conditions’ concept: Conditions can be used to execute particular task.



Run Playbook: ansible-playbook two.yml

Command to know which operation system we are using: cat /etc/os-release

---

- hosts: dev

user: ansible

become: yes

connection: ssh

tasks:

- name: installing httpd when os is RedHat

action: yum name=git state=present

when: ansible\_os\_family = ="RedHat"

- name: installing apache when os is Debain

action: apt name=apache2 state=present

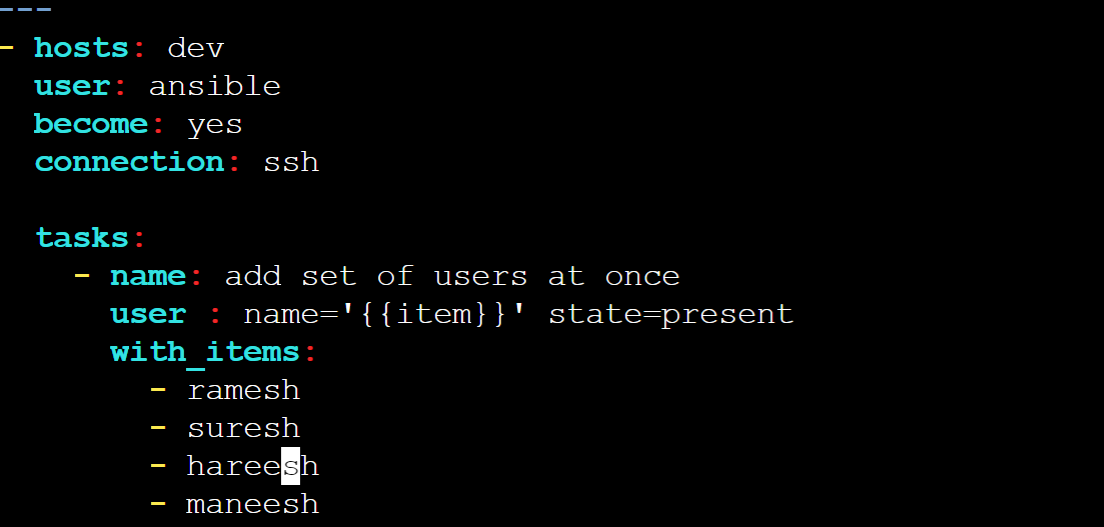
when: ansible\_os\_family == "Debain"

- name: install git when os is ubuntu

command: apt install git -y

when: ansible\_os\_family == " ubuntu"

* Adding and Delete users using loops’ concept:



Command to check users: cat /etc/passwd

Playbook Script: ---

- hosts: dev

user: ansible

become: yes

connection: ssh

tasks:

- name: add set of users at once

user: name='{{item}}' state=present

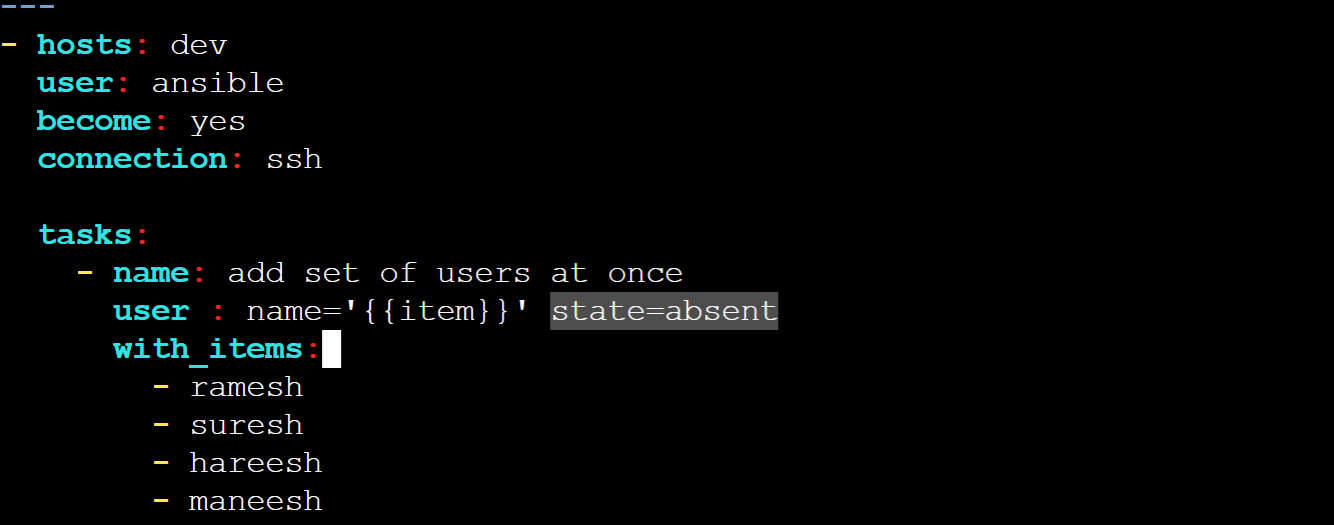
with\_items:

- ramesh

- suresh

- hareesh

- Maneesh



* Adding and Delete GROUPS using loops’ concept:

Command to check users: cat /etc/passwd



* How to get the code from git:

---

- hosts: localhost

tasks:

- name: Install git

yum: name=git state=present

- name: gather code

git:

repo: 'https://github.com/GREATCODERHYD/

GITBRANCH.git'

dest: /home/mahesh/folder2/

* How to CREATE FILES IN LOCALHOST JENKINS:

---

- hosts: dev

user: rakesh

connection: ssh

become: yes

tasks:

- name: create file

file:

path: /home/rakesh/file1.txt

state: touch

* How to CREATE Folder IN LOCALHOST JENKINS:

---

- hosts: dev

user: rakesh

become: yes

connection : ssh

tasks:

- name: file creation

file:

path: "jooush"

state: directory

* How to add data in a file ansible:

---

- hosts: localhost

user: rakesh

tasks:

- name: adding data to a file

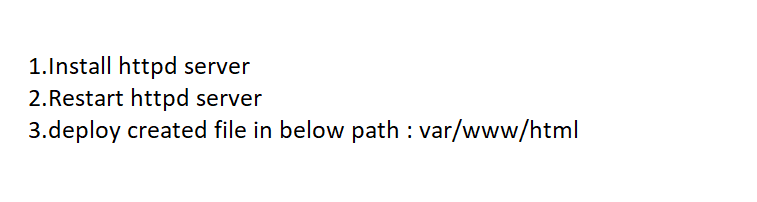
copy:

dest: 'file32.txt'

content: |

this is file1, we are inserting data into a file rakes

* Deployment using ansible:



---

- hosts: localhost

user: rakesh

become: yes

connection: ssh

tasks:

- name: Installing httpd

action: yum name=httpd state=present

- name: create a file

file:

path: "/var/www/html/index.html"

state: touch

- name: Data insertion to a file

copy:

content: |

<h1> Hi Good Morning</h1>

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

- name: restarting httpd

service: name=httpd state=restarted

* Tomcat installation:

---

Become: yes

hosts: all

user: root

connection:ssh

tasks:

- name: Create a Tomcat Directory

file:

path: /tomcat

state:directory

- name: Install tomcat

unarchive:

src:https://dlcdn.apache.org/tomcat/tomcat-10/v10.1.5/bin/apache-tomcat-10.1.5.tar.gz

dest: /tomcat

- name: start tomcat

command: "nohup /tomcat/bin/startup.sh"

* Jenkins setup using ansible:

Step 1: Download Jenkins.io packages from file

Step 2: Import Keys

Step 3: Update imported Keys

Step 4: EPEL & Install Java

Step 5: Install Jenkins

Step 6: import Systemd

Step 7: Restart Jenkins

---

-hosts:localhost

user:mahesh

connection:ssh

become:yes

tasks:

-name: getting Jenkins URL from WEB

get \_url:

url: <https://pkg.jenkins.io./redhat-stable/jenkins.repo>

dest: /etc/yum.repos.d/jenkins.repo

-name: getting key from Jenkins.io

ansible.builtin.rpm\_key:

state:present

key:https://pkg.jenkins.io/redhat-stable/Jenkins.io.key

-name: restart links

action: yum name= “\*” state=present

-name: Install Epel

command: amazon-linux-extras install epel -y

-name: Install Java 11

command: amazon-linux-extras install java-openjdk11 -y

-name: Install Jenkins

command: yum install jenkins -y

-name: Import system to restart server

ansible.builtin.systemd:

daemon\_reload: yes

-name:restart Jenkins

ansible.builtin.systemd:

name: jenkins

state:restarted

* Creating roles using ansible:
* WHAT IS ansible-galaxy:
* What is ansible vault:

Ansible-valut encrypt one.yml

Ansible-valut decrypt one.yml

Ansible-valut rekey one.yml

Ansible-valut create one.yml: before vi edit