**IMPLEMETING JINJA2 TEMPLATE:-**



**Create a file named “motd-facts.j2” with below content**

cat motd-facts.j2

This system is based on

{{ ansible\_distribution }} {{ ansible\_distribution\_version }} deployed on {{ ansible\_architecture }} architecture.

**[ansible@node1 project]$ cat jinja2.yaml**

---

- name: jinaj2

hosts: all

become: true

remote\_user: ansible

tasks:

- template:

src: templates/motd-facts.j2

dest: /etc/motd

owner: root

group: root

mode: 0644

force: yes

# ansible-playbook jinja2.yaml

[ansible@node1 templates]$ cat httpd-conf.j2

<VirtualHost {{ansible\_hostname}}:80>

ServerAdmin root@{{ ansible\_fqdn }}

ServerName {{ ansible\_fqdn }}

DocumentRoot /var/www/html

</VirtualHost>

[ansible@node1 project]$ cat httpd-j2.yaml

- hosts: all

remote\_user: ansible

become: yes

tasks:

- name: Package Installation ...

yum: name=httpd state=present

- name: Restart httpd ...

service: name=httpd state=started enabled=yes

- name: using template ...

template:

src: templates/httpd-conf.j2

dest: /etc/httpd/conf.d/"{{ ansible\_hostname }}".conf

force: yes

notify: restart\_httpd

- name: Adding content in index.html file

copy: content="hello from {{ ansible\_hostname }}\n" dest=/var/www/html/index.html

handlers:

- name: restart\_httpd

service: name=httpd state=restarted

**Ansible-Vault:-**

Ansible Vault is a feature of ansible that allows you to keep sensitive data such as passwords or keys in encrypted files, rather than as plaintext in playbooks or roles.

These vault files can then be distributed or placed in source control. Ansible automatically decrypts vault-encrypted content at runtime when the key is provided.

To check the details about vault

# ansible-vault --help

Create:- To create a new ansible vault file in the encrypted format.

**# ansible-vault create <filename>**

View:- To see the data of encrypted file

**# ansible-vault view <filename>**

Edit:- To edit encrypted file.

**# ansible-vault edit <filename>**

Encrypt: To encrypt a file if file is exists.

**# ansible-vault encrypt <filename>**

Decrypt: To decrypt encrypted file

**# ansible-vault decrypt <filename>**

**--ask-vault-pass**: To provide password while running playbook

**--vault-password-file**: To pass a vault password through a file.

# echo “redhat” > pass

[ansible@node1 test]$ cat pass

redhat

[ansible@node1 test]$ ansible-vault view vault\_pass.yml --vault-password-file pass

password: ghp\_sYa9SZuJTZWVjZaloUkUOmqVsJ0wRC0Gyf24

[ansible@node1 test]$ cat vault.yaml

---

- name: vault testing...

hosts: node1.example.com

remote\_user: ansible

become: yes

vars\_files:

- vault\_pass.yml

tasks:

- name: clone the repo

git:

repo: https:// manoj-86:{{ password }}@github.com/ manoj-86/test.git

dest: /home/ansible/git

# **ansible-playbook vault.yaml** --vault-password-file pass

[ansible@node1 test]$ cat vault.yaml

---

- name: vault testing...

hosts: all

remote\_user: ansible

become: yes

vars\_files:

- vault\_pass.yml

tasks:

- name: Install git in all servers...

yum:

name: git

state: present

- name: Create a directory...

file:

path: /home/ansible/git-test

state: directory

mode: 0755

- name: clone the repo

git:

repo: [https://manoj-86](https://manoj-86/):{{ password }}@github.com/manoj-86/test.git

dest: /home/ansible/git-test

force: yes

# ansible-playbook vault.yaml --vault-password-file pass --syntax-check

# ansible-vault view vault\_pass.yml --vault-password-file pass