Ansible-inventory –list #show all inventory conf

Ansible all –m ping #ping all nodes from hosts file

Ansible staging\_servers –m ping #ping all staging\_servers nodes from hosts file

Ansible all –m shell –a “uptime” #-a argument = shell command

Ansible all –m shell –a “cat /etc/hosts”

ansible all -m copy -a "src=/etc/hosts dest=/home mode=777" –b

#copy file, -b = sudo, mode = permission 777

ansible all -m file -a "path=/home/file1 state=absent" –b #delete file with module file

ansible all –m get\_url –a “url=https://web.site/download/file123 dest=/home” –b

#download file from website to remote nodes

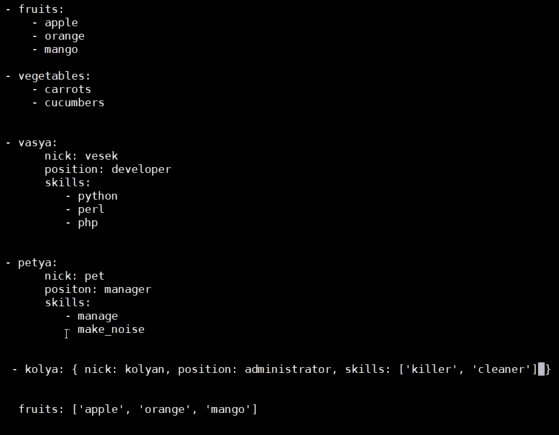
Ansible all –m yum –a “name=stress state=installed” –b

Ansible-doc –l #show all modules available in ansible

Ansible-vault create mysecret.txt (edit, view, rekey)

Ansible-playbook playbook\_vault –ask-vault-pass

Ansible-vault encrypt\_string –stdin-name “mypassword\_var



PLAYBOOKS:

**example playbook1.yml**

- name: Install and run nginx web-server

hosts: all

become: yes

tasks:

- name: install nginx

apt: name=nginx state=latest

- name: start nginx

service: name=nginx state=started enabled=yes

ansible-playbook playbook1.yml

**example playbook2.yml:**

- name: Install, copy page and run nginx web-server

hosts: all

become: yes

vars: #defining variables

source\_file: ./mywebsite/index.html

destin\_file: /usr/share/nginx/html/

tasks:

- name: install nginx

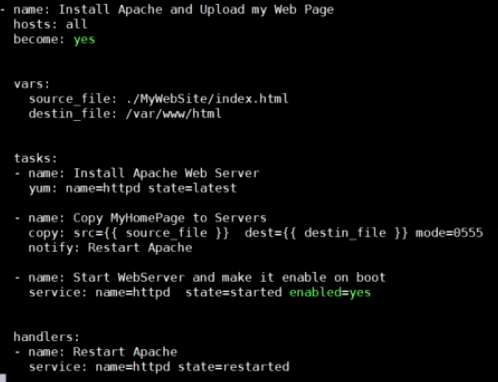
apt: name=nginx state=latest

- name: copy webpage to web-server

copy: src={{ source\_file }} dest={{ destin\_file }} mode=0555 #using variables and jinja

- name: start nginx ad make it enable

service: name=nginx state=started enabled=yes



**example playbook3.yml**

- name: Playbook variables

hosts: all

become: yes

vars:

message1: privet

message2: kak dela?

secret: 123

tasks:

- name: Print secret variable

debug:

var: secret

- debug:

msg: "soobshenie: {{ privet }}"

- debug:

msg: "soobshenie: {{ message2 }}"

- set\_fact: full\_message\_var = "{{ message1 }} {{ message2 }}"

- debug:

var: full\_message\_var

- shell: uptime

register: result

- debug:

var: result

**example playbook4.yml when**

- name: Install, copy page and run nginx web-server

hosts: all

become: yes

vars: #defining variables

source\_file: ./mywebsite/index.html

destin\_file: /var/www/html/

tasks:

- name: install apache for Ubuntu

apt: name=apache2 state=latest

**when**: ansible\_os\_family == "Ubuntu"

- name: install apache for RedHat

yum: name=httpd state=latest

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

- name: copy webpage to web-server

copy: src={{ source\_file }} dest={{ destin\_file }} mode=0555 #using variables and jinja

- name: start apache in Ubuntu and make it enable

service: name=apache2 state=started enabled=yes

**when**: ansible\_os\_family == "Ubuntu"

- name: start apache in RedHat and make it enable

wervice: name=httpd state=started enabled=yes

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

**example playbook5.yml BLOCK**

- name: Install, copy page and run nginx web-server

hosts: all

become: yes

vars: #defining variables

source\_file: ./mywebsite/index.html

destin\_file: /var/www/html/

tasks:

- **block**: #block for redhat

- name: install apache for RedHat

yum: name=httpd state=latest

- name: copy webpage to web-server

copy: src={{ source\_file }} dest={{ destin\_file }} mode=0555 #using variables and jinja

- name: start apache in RedHat and make it enable

service: name=httpd state=started enabled=yes

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

- **block**: #block for ubuntu

- name: install apache for Ubuntu

apt: name=apache2 state=latest

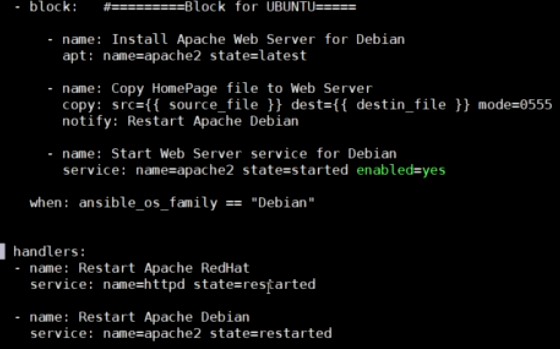
- name: copy webpage to web-server

copy: src={{ source\_file }} dest={{ destin\_file }} mode=0555 #using variables and jinja

- name: start apache in Ubuntu and make it enable

service: name=apache2 state=started enabled=yes

**when**: ansible\_os\_family == "Ubuntu"



- name: create

hosts: all

become: yes

vars:

mytext: "Hello, Timur"

tasks:

- name: Ping test

ping:

- name: create file1

copy:

dest: /home/file1.txt

content: |

This is file.txt

{{ mytext }}

delegate\_to: aws\_ubuntu2

- name: create file2

copy:

dest: /home/file2.txt

content: |

This is file2.txt

{{ mytext }}

- name: Reboot my servers

shell: sleep 3 && reboot now

async: 1

poll: 0

- name: Wait till my servers will come up online

wait\_for:

host: "{{ inventory\_hostname }}"

state: started

delay: 5

timeout: 40