**Requirement:**

Ansible Linux Server Host ( Ansible, Python, Openssh)  
Ansible Linux Client host / Ansible Windows Client Host (Openssh, Python)

**Topics :**

1. Architecture & Documentation.

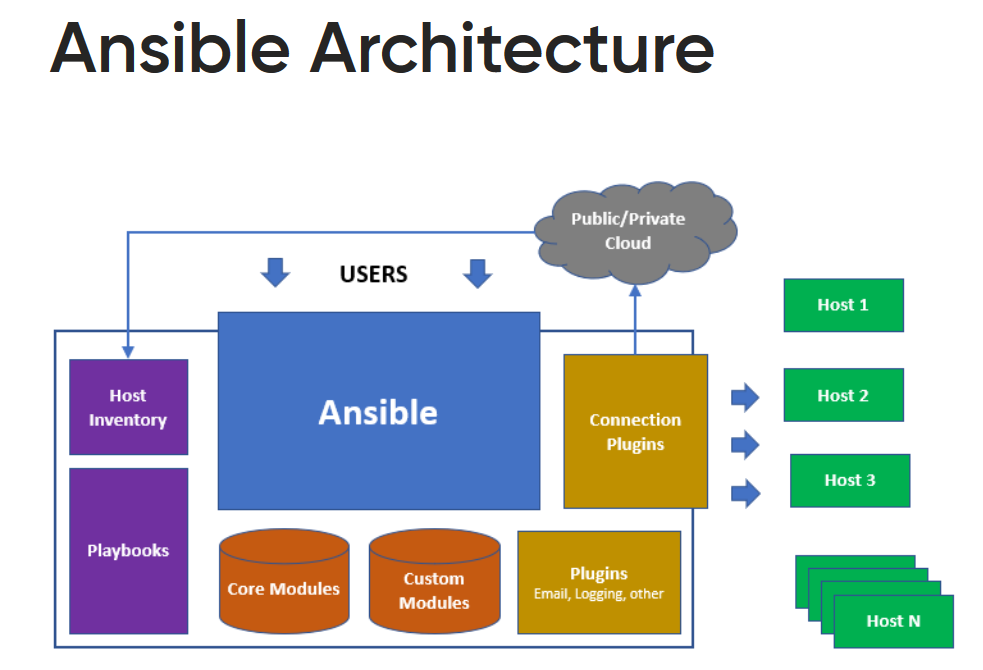
2. Installation

3. SSH Configuration

4. Ansible Adhoc Commands

5. Playbooks

6. Roles

**Architecture & Documentation:**

Ansible Installation Document

<https://docs.ansible.com/ansible/latest/installation_guide/intro_installation.html>

<https://galaxy.ansible.com/>

<https://computingforgeeks.com/install-and-configure-ansible-tower-on-centos-rhel/>

<http://docs.ansible.com/ansible/developing_modules.html>

Ansible document -----> <https://docs.ansible.com/>

**Installation & SSH Configuration:**

Step1:

Install Ansible & python in Server. After installing ansible and python in Server follow below steps

1. Add Client host in Ansible Server host file(/etc/ansible/hosts) at the end

**[webservers]**

**<ip-address>**

1. Add ansible user in both server and in client host.

**useradd user**

1. Modify ssh config(vi /etc/ssh/sshd\_config) in both server and client hosts as mentioned below

**PasswordAuthentication yes**

1. sudoers file(vi /etc/sudoers) in both server and client hosts as mentioned below.

**## Allow root to run any commands anywhere**

**##root ALL=(ALL) ALL**

**root ALL=NOPASSWD: ALL**

**user ALL=NOPASSWD: ALL**

systemctl restart sshd in both server and client.

1. Generate ssh key in server and copy that key in host machine as mentioned below.

**su - user**

**ssh-keygen**

**ssh-copy-id -I user@ipaddress**

**ssh user@ipaddress**

1 yum list ansible

2 yum list git

3 yum list docker

4 yum list Jenkins( Will install Jenkins through docker)

5 yum list yum list httpd

6 yum list tomcat

7 yum list mysql

8 cat /etc/redhat-release

9 yum list sonarqube( Will install sonarqube through docker)

10 yum list java

11 yum list maven

12 yum list ant

13 yum list python

----------------------------------------

cd /etc/ansible/hosts --- add server ip address here at the end of file.

cd /usr/lib/python2.7/site-packages/ansible/modules/ --- modules python list

**Ansible Adhoc Commands:**

ansible-doc -l -----> list of modules

ansible-doc <Module\_Name> ------> details about module with examples

ansible webservers -m user -a "name=prasad password=123 createhome=yes" -b

ansible -m command -a "uptime" webservers

ansible -m ping webservers

ansible -m user -a "name=john password=john" -b webservers

ansible -m yum -a "name=httpd state=present" -b webservers

ansible -m shell -a "ls -lrt /" webservers

ansible -m copy -a "src=/etc/yum.conf dest=/tmp/" -b webservers

ansible webservers -m file -a "path=/opt/oracle/binaries state=directory mode=0755" \

ansible webservers -s -m group -a "name=weblogic state=present" -i ansible\_hosts -b --become-user=weblogic

ansible webservers -m shell -a "cat /etc/passwd|grep -i vagrant" -b -K

ansible webservers -m file -a "path=/opt/oracle/binaries state=directory mode=0755" \

ansible webservers -s -m group -a "name=weblogic state=present" -i ansible\_hosts -b --become-user=weblogic

ansible webservers -m file -a "path=/tmp/testfile state=touch mode=0755"

ansible webservers -m ping -i ansible\_hosts --user=vagrant

ansible webservers -m ping -i ansible\_hosts --user=vagrant --ask-pass

ansible webservers -m shell -a "cat /etc/passwd|grep -i vagrant" -s --ask-sudo-pass

ansible webservers -m shell -a "cat /proc/meminfo|head -2"

ansible webservers -a "free -m" -i ansible\_hosts

ansible webservers -s -m cron -a "name='daily-cron-all-servers' minute=\*/15 job='/path/to/minute-script.sh'"

ansible webservers -s -m cron -a "name='daily-cron-all-servers' hour=4 job='/path/to/hour-script.sh'"

ansible webservers -s -m cron -a "name='daily-cron-all-servers' special\_time=reboot job='/path/to/startup-script.sh'"

ansible webservers -s -m cron -a "name='daily-cron-all-servers' special\_time=daily job='/path/to/daily-script.sh'"

ansible webservers -s -m cron -a "name='daily-cron-all-servers' special\_time=weekly job='/path/to/daily-script.sh'"

ansible replace --sudo /etc/motd "This server is managed by Ansible"

ansible webservers -b -m lineinfile -a 'dest=/etc/motd line="This server is managed by Ansible"'

ansible webservers --become --module-name=lineinfile --args='dest=/etc/motd line="This server is managed by Ansible"'

ansible webservers -b -m copy -a 'dest=/etc/motd content="This server is managed by Ansible"'

ansible -v webservers -m shell -a "mkdir /srv/www" -u <targetuser>

ansible -v webservers -m file -a "path=/srv/www state=directory" -u <targetuser>

ansible -m git -a "repo=https://github.com/devops-trainer/dockerdemo dest=/prasad" -b webservers

ansible webservers -m setup -a "filter=user1" ----- > setup module by default it fetching the gathering facts of hosts. We can able to see the gathering facts while running any playbooks / ad-hoc command.

**Playbooks will keep complete automation.**

Syntax Checking:-

ansible-playbook playbook1.yml --syntax-check

ansible-playbook playbook1.yml --list-hosts

ansible-playbook --check playbook1.yml 🡪 dry run---> means just checking not before running in host file.

Vi file.yml

---

- hosts: webservers

become: true

become\_user: root

tasks:

- name: Check Docker Package Availability

yum: name=docker state=latest

- name: Install Docker Package

yum: name=docker state=installed

- name: start the service

service: name=docker state=started

OUTPUT:-

PLAY [webservers] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

TASK [Gathering Facts] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [10.128.0.5]

TASK [Check Docker Package Availability] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

changed: [10.128.0.5]

TASK [Install Docker Package] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

changed: [10.128.0.5]

TASK [start the service] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

changed: [10.128.0.5]

PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

10.128.0.5 : ok=4 changed=3 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

**Roles:**

**ansible-galaxy install geerlingguy.mysql**

**ansible-galaxy install geerlingguy.java**

**ansible-galaxy install geerlingguy.jenkins**

---

- hosts: webservers

become: true

become\_user: root

vars:

jenkins\_http\_port: 9090

roles:

- geerlingguy.java

- geerlingguy.jenkins

**ansible-playbook playbook.yml --start-at-task="install packages"**