**Ansible**

configuration management tool to install same data on large number of servers

congiguration tools are ansible, cfengine, chef, puppet,salt etc

any file name extension should be on : .yml or .yaml

the file which we create is called playbook.

Classs 1

1. Connect instance in git
2. Change hostname optional
3. Add ansible user : useradd ansible
4. Create passwd for ansible and rememeber it
5. Give root equal permission for ansible through :visudo
6. After entering of sudo copy paste route line and replace root with ansible and add nopasswd for 2nd all
7. Switch user to ansible
8. To install any application : sudo yum install app name –y
9. Generate key file file instance :” ssh-keygen
10. To see generated key file : ls –la
11. To enter ssh file : c .ssh
12. Tso see keys list in ssh file : ll
13. To see key file : cat public key file
14. To copy public key file fron instance1 to instance2 command is: ssh-copy-id ansible@public ip of instance2
15. If the permission deniened then use this command of: sudo vim /etc/ssh/sshd\_config
16. Then change passwordauthentication to yes
17. Then restrart with : sudo systemctl restart sshd
18. Check serive of sshd : sudo service sshd status
19. Then use ssh-copy-id ansible@public ip of instance2
20. And enter passwd of anasible user
21. After enter passwd to enter instance2: ssh ‘ansible@public ip of instance 2’
22. Create more instances and follow above 1 to 9 steps and 15 to 18 for master connect to extra instances

**Class 2**

1. Connect created instance to git
2. Install ansible
3. Enter into ansible directory : cd /etc/ansible
4. To see details of ansible folder : ll [ansible folder consist of 1) ansible.cfg this file consist of ansible configuration 2) hosts file consist of our ips 3) roles ]
5. To enter public of required instances : vim hosts
6. Add private ips of instances in hosts and mention group name if requires the ips under the group name is consider in that group
7. Enter ansble config with: vim ansible.config
8. Switch to ansible user

**Linux commands for ansible**

1. To check connectivity of added ips in host : ansible –m ping < ip or gn (groupname) or all > as per requirement
2. To enterto s hosts from ansible: sudo vim hosts
3. To see files in other instances from master instance use: ansible ip/gn/all –b –a ‘ls –la’
4. To create files in other instances from master instance: ansible ip/gn/all –b –a ‘touch file name’
5. To delete files in other instances from master instance: ansible ip/gn/all –b –a ‘rm -rf file name’
6. To create directory in other instances from master instance: ansible ip/gn/all –b –a ‘mkdir /path/dir name’
7. To create file directory in other instances from master instance: ansible ip/gn/all –b –a ‘touch /path/dir name/new file name’
8. To install application in other instances from master instance: ansible ip/gn/all –b –a ‘yum install appname –y’
9. To uninstall application in other instances from master instance: ansible ip/gn/all –b –a ‘yum remove appname –y’
10. To add new user in other instances from master instance: ansible ip/gn/all –b –a ‘useradd name of user’
11. To see user data in other instances from master instance: ansible ip/gn/all –a ‘cat /etc/passwd’

**Adhode or ansible module commands**

1. To install application through ansible module command: ansible ip/gn/all –m yum –b –a ‘name=apname state=present’
2. To install application through ansible module command: ansible ip/gn/all –m yum –b –a ‘name=appname state=absent’
3. To start service: ansible ip/gn/all –m service –b –a ‘name=appname state=started’
4. To stop service: ansible ip/gn/all –m service –b –a ‘name=appname state=stopped’
5. To check status : ansible ip/gn/all –a ‘service httpd status’
6. To add user: ansible ip/gn/all –m user –b –a ‘name=required name state=present’
7. To delete user: ansible ip/gn/all –m user –b –a ‘name=required name state=absent’
8. To create a file : ansible ip/gn/all –m file –b –a ‘pat= name of file state=touch’
9. To create a file IN DIRECTORY : ansible ip/gn/all –m file –b –a ‘pat= dir name/name of file state=touch’
10. To create a directory : ansible ip/gn/all –m file –b –a ‘path=/required path name of file state=directory’

**Class-3 playbook commands**

1. This playbook commands will be create in file.yml format: sudo vim filename.yml
2. Command format

---

- host: ip/gn/all

become: yes

tasks:

- name: label of command

yum/service: name=appname state=present or absent/started or stopped

1. To execute created.yml file : ansible-playbook file,yml
2. To create multiple apps in sigle command tpye1
3. ---

- host: ip/gn/all

become: yes

tasks:

- name: label of command

yum: name=appname1 state=present/absent

- name: label of command

yum: name=appname2 state=present/absent

type2

---

- host: ip/gn/all

become: yes

tasks:

- name: label of command

yum: name=appn1,n2,n3.etc state=present/absent

type3

---

- host: ip/gn/all

become: yes

tasks:

- name: label of command

yum: state=latest/absent name={{item}}

with\_items:

* Git
* Httpd
* Wget ….

If you don’t want to install any app remove it or just add # before the name you don’t want to install

\*all commands should be create in only <filename.yml>

Use command to install: ansible-playbook <filename.yml>

To copy file from master to worker instance

---

- host: ip/gn/all

become: yes

tasks:

- name: label of command

copy:

src: /path/filename

dest: /path

to install start stop copy of httpd data from all services from one command

**---**

**- hosts: all/gn/ip**

**become: yes**

**tasks:**

**- name: installing httpd**

**yum: name=httpd state=present**

**- name: stop httpd service**

**service: name=httpd state=stopped**

**tags: stop**

**- name: copy my data**

**copy: src=/opt/index.html dest=/var/www/html**

**tags: copy**

**- name: start httpd service**

**service: name=httpd state=started**

**tags: start**

**- name: checking status httod**

**shell: service httpd status**

**register: status\_httpd**

**- debug:**

**var: status\_httpd**

**Class 4 roles**

1. To change data in only one server command is: **ansible-playbook path/file.yml –limit ip of server**
2. To do only action of httpd start/stop for all servers: **ansible-playbook httpd.yml --tags start/stop**
3. To do only action of httpd start/stop for one server: **ansible-playbook httpd.yml --tags start/stop --limit ip**
4. To create a directory for reference: mkdir filename
5. Enter the created directory
6. To create role file for any app: sudo ansible-galaxy init name of role
7. Create or copy of index,htm dat to files in created role folder
8. Copy of .yml data to main.yml in tasks of in created role
9. Ceate an file of mainplay.yml with data of

**---**

**- hosts: all/gn/ip**

**become: yes**

**roles:**

**{role: file of role}**

**{role: file of role} etc**

**10** to apply mainplay command is: ansible-playbook mainplay.yml

11 add and delete data as per your requiremwnts in mainplay.yml