**Ansible**

---

-hosts: all

tasks

-name: install webservers

apt:

-name : apache2

State: present

(Here – apt is ansible module for installing software

................................................................................................................

---

hosts: all

become: yes

tasks:

-name: i want to install apache2

apt:

name: apache2

state: present

-name: i want to install apache2

service:

name: apache2

state: restart

**Password less authentication**

Master server configuration

Node server configuration

In master server ssh-keygen

Ssh-copy-id <private ip of nodes>

Ssh<private ip of nodes>

**Git**

version control system

git repository

create git account

create token ID

create remote repo

Create folder and git bash over there and execute commands

**GIT HUB**

Login

Globe symbol

Developer settings

Personal access token

Generate token

My token

Never expire

Select all the check boxes

Generate token

Copy token and securely save

Copy all the commands from git hub and git bash those one by one to created folder

And execute each one command one by one.

**Configuration management**

Ssh-copy-id ip of node server

Vi myhosts

Private ip of nodes

:wq!

Ansible all –i myhosts –m ping

Ansbile –Playbook

IAC- YAML, RUBY

Provision: It provides suitable environment for deploying applications on production environment.

Install softwares

Create file and folder

Create user and user group

Deploying application ...etc

**Install and configure ansible master server**

Configure master server

Configure node server

Establish communication between master and node server

Ansible all –i myhosts –m ping

**PLAY BOOK**

Vi mysample.yml

Ansible-playbook -i myhosts mysample.yml

Introduction to ansible

**What ansbile does**

Provisioning

Change management

Automation

Orchestration

Configuration of server

Application deployment

**Why ansible**

It is free and open source management

Agent less

No data base

Highly flexible

It is vey simple and human readable

Custom modules

Self documentation

Configuration roll back in case of error

**Ansible working architecture drawing**

Core component of Ansible

Adhoc commands

Playbooks

Modules

Facts

Inventories

Variables

Configuration files

Templates

Handlers

Roles

Vault

Ansible ad-hoc commands

Ansible all –i myhosts –m ping

Ansbile all –i myhosts –b –m apt –a name =”tree”

**Ansible playbooks**

Playbooks are your instruction manual

A playbook is made up with individual plays

A playbook is a task

Play book is in YAML format

Bash is a scripting languge of - Windows

Shell is for scripting language of - Linux

**YAML**- YET ANOTHER MARKUP LANGUAGE

Key: values

**Ansbile Modules**

Ansible with n’ number of modules

Modules are discrete units of code that can be used from command line or playbook task

Ansible executes each module, usually on th remote target node and collect return values

Each module supports taking argument nearly all modules take key= value management space limited

Same module takes no arguments and the command/shell modules simply take the string of the command you want to run

1. Apt
2. Service
3. Ping
4. Yum
5. get\_url
6. etc

we can write our own module by using java or python language but not required in general

Ansible Facts

Gather\_facts

Red hat server- some commands will be vary

Ansible Inventories

Default inventory

Custom inventory

**Custom inventory**

1.Static inventory

2. Dynamic Inventory

ANSIBLE VARIABLE

Group variable

Host variable

Mydebug.yml

Create files

Static files

Dynamic files

Remote files

Check in google – Files modules in Ansbile

While writing playbooks we can call other playbooks

Ansible – template

This is an ansible module named as template

Ansible handlers

Ansible roles