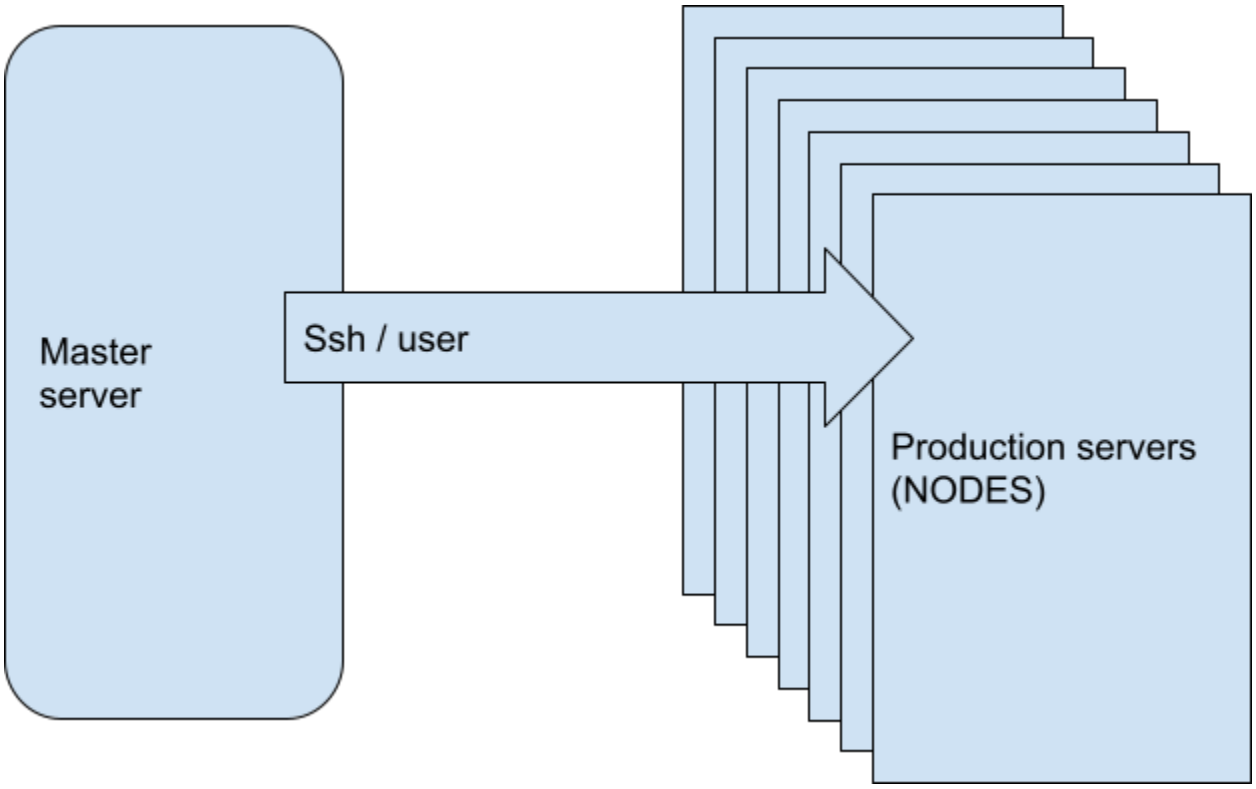
SSH passwordless Authentication :



Stage 01: Configure Master server

1. ***Create ubuntu 16 server with t2.micro***

2. ***Create user (maha)***
   # adduser maha

3. ***Make maha user as sudoer***
   # visudo
   maha   ALL=(ALL)   NOPASSWD: ALL
   Ctr+x, press y, press enter

4. ***Change password authentication***
   # vi /etc/ssh/sshd_config
   Password authentication yes
   :wq!

5. ***Restart ssh service***
   # service ssh restart

Stage 02: configure Production servers (nodes)

6. Create ubuntu 16 server with t2.micro

7. Create user (maha)
 # adduser maha

8. Make maha user as sudoer
 # visudo
maha ALL=(ALL) NOPASSWD: ALL
Ctr+x, press y, press enter

9. Change password authentication
 # vi /etc/ssh/sshd_config
Password authentication yes
:wq!

10. Restart ssh service
 # service ssh restart

Stage 03: Connect from master to nodes without passwd and pem file

ON MASTER As MAHA user

- *generate key pair on maha user*
  $ ssh-keygen

- *Copy public key into node server side*
  $ ssh-copy-id <private ip of nodes>

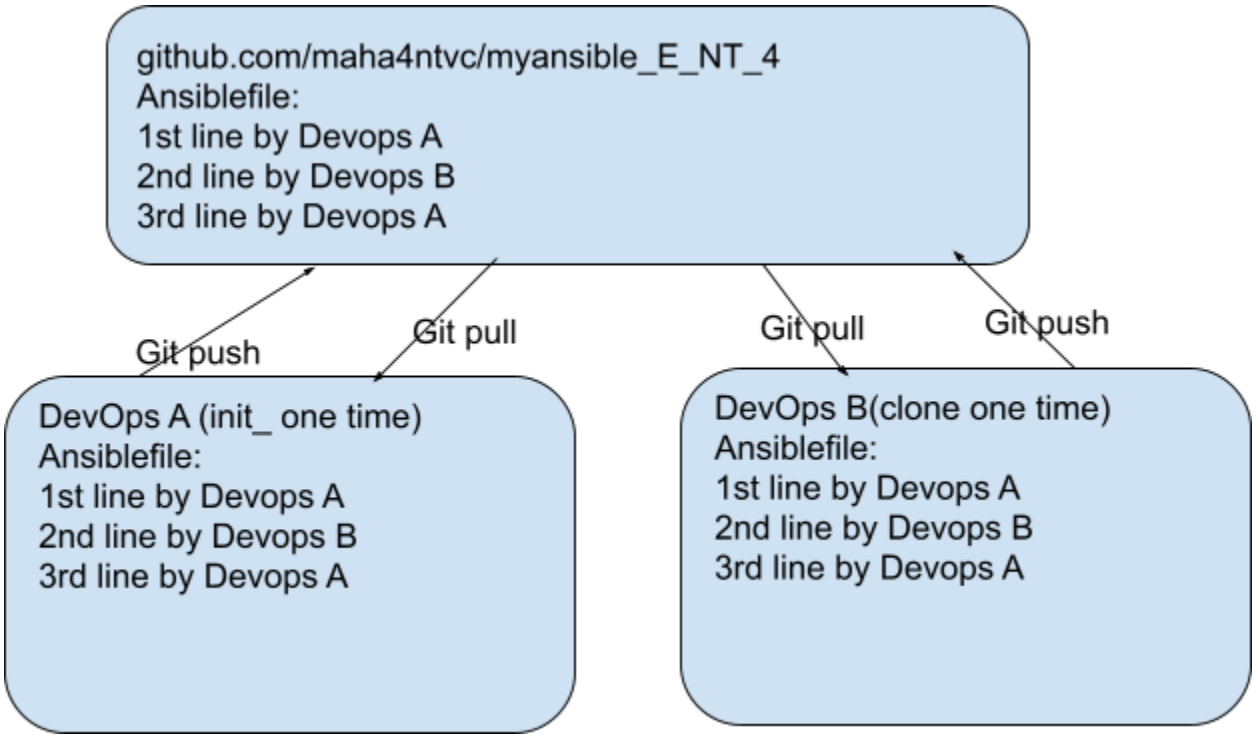
- *We can connect from master to nodes without passwd and with pem file*
  $ ssh <private ip of nodes>

Ansible:

1) *How to use git*
2) *What is CM?*
3) *What is ansible?*
4) *Ansible install configuration*
5) *Ansible ART*

What is Git and use?

Git version control:



Git add -A
Git commit -m " comment"
Git push

vs

Git pull

1) Create github account
2) Sing in github.com
3) Create Repo (mytestrep03dec)
4) Initial repo in ur laptop

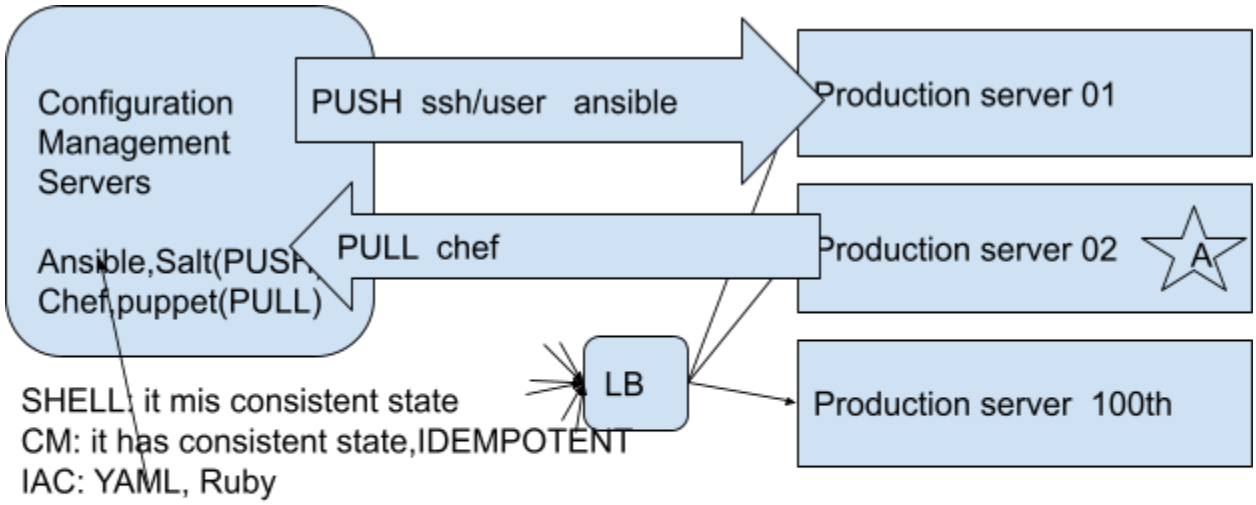
```
echo "# -mytestrep03dec" >> README.md
git init
git add README.md
git commit -m "first commit"
git config --global user.name "maha"
git config --global user.email "maha@gamail.com"
git commit -m "first commit"
git branch -M main
git remote add origin
https://github.com/maha4ntvc/-mytestrep03dec.gi
t
git push -u origin main
```

5) *Create a file*

git add -A

git commit -m "comment"

git push

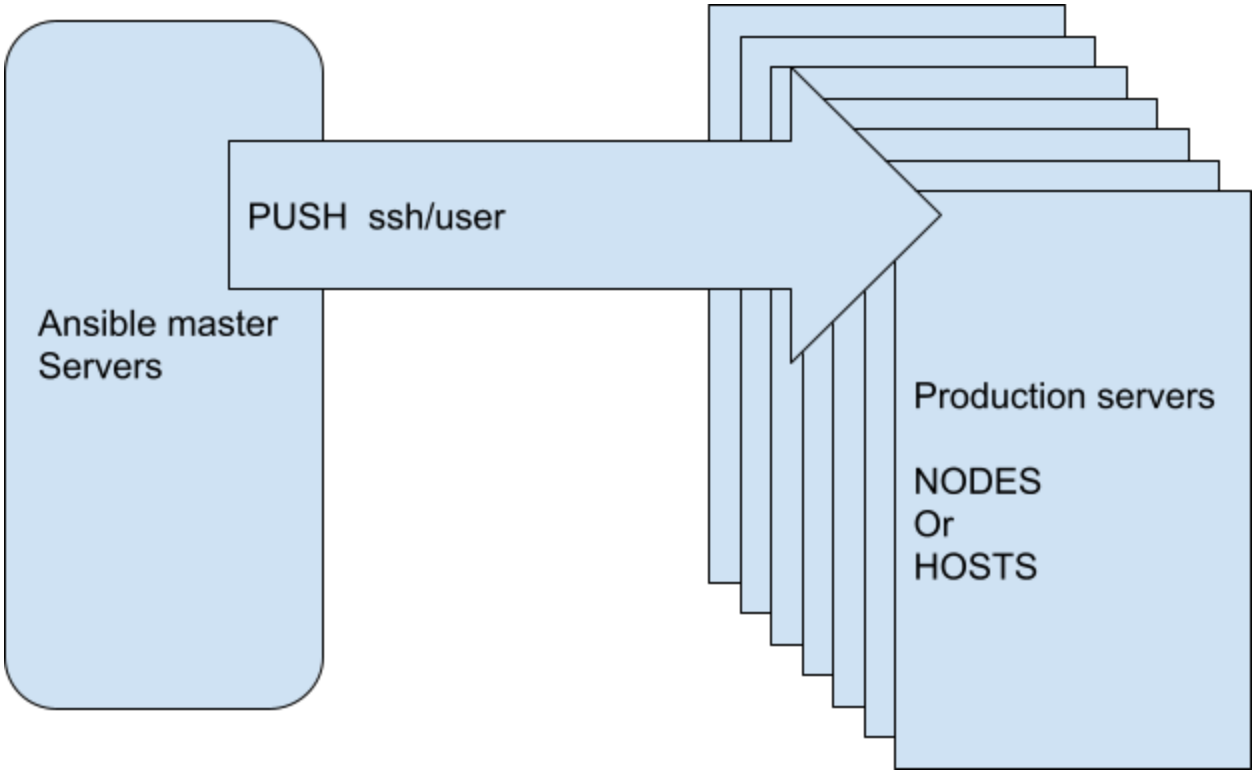


Provision: it provide suitable environment for deploying application

1) Install softwares
2) Create files and folders
3) Create users and groups
4) Change configure files 5) deploy application ……etc

Configuration Management : CM

PUSH : Ansible


PUSH ssh/user
Ansible master
Servers
Production servers
NODES
Or
HOSTS

Ansible master server install and configuration:

1) ***Create a ec2 ubuntu 16 servers***
2) ***Create ansible user as maha with passwd***
   # adduser maha
3) ***Make ansible user as sudoer***
   # visudo
   maha ALL=(ALL)   NOPASSWD: ALL
   Ctr+x, press y, press enter

4) ***Make password authentication yes***
   # vi /etc/ssh/sshd_config
   Password authentication yes
   :wq!

5) ***Restart ssh service***
   # service ssh restart

6) ***Install ansible***
   # apt-add-repository ppa:ansible/ansible
   # apt-get update
   # apt-get install ansible

Ansible NODES configuration:

7) *Create a ec2 ubuntu 16 servers*
8) *Create ansible user as maha with passwd*
   # adduser maha
9) *Make ansible user as sudoer*
   # visudo
   maha ALL=(ALL) NOPASSWD: ALL
   Ctr+x, press y, press enter

10) *Make password authentication yes*
    # vi /etc/ssh/sshd_config
    Password authentication yes
    :wq!

11) *Restart ssh service*
    # service ssh restart

12) *Don't Install ansible*
    # apt-add-repository ppa:ansible/ansible
    # apt-get update
    # apt-get install ansible

Ansible master server communication with ansible nodes: without password and pem file

ON ANSIBLE MASTER AS Ansible user(maha)

13) *$ ssh-keygen*
14) *$ ssh-copy-id <private ip of ansible nodes>*
15) *$ ssh <private ip of ansible nodes>*
16) *We are trying to connect by ansible*
    - *Create hosts file*
      Vi myhosts
      <private ip of node1>
      <private ip of node2>
      <private ip of noden>
      :wq!
    $ ansible all -i myhosts -m ping
    Success with green colour

1) File extends with .yaml or yml
2) File start with ---
3) Key:values
4) Space notation

\

By using ansible playbook

1) Create a file
2) Copy static file
3) Copy dynamic file
4) Download remote files

How to create ansible nodes by executing playbook

1. ***Create IAM Admin user***
2. ***Install python-pip on ansible master as root user***

   # sudo apt-get install python-pip

3. ***Install boto by using pip on ansible master as root user***

   # sudo pip install boto

4. ***Declare IAM user details on .boto file on ansible master as root user***

vi .boto

```
[Credentials]
aws_access_key_id = foo
aws_secret_access_key = bar
```

5. ***Make ansible master as node on Ansible master as ansible user$ maha***

   $ ssh-copy-id localhost

6. ***Create AMI on ansible node***

   ami-076a2532e5946c0f4

7. ***Create and execute playbook on ansible as node***

```
---
- hosts: localhost
  become: yes
  tasks:
  - name: create ansible nodes
    ec2:
      key_name: myOregonOfflinekey
      instance_type: t2.micro
      image: ami-076a2532e5946c0f4
      group: mySG21Dec
      count: 2
      region: "us-west-2"
      wait: yes
      vpc_subnet_id: subnet-00b9d5912be98e6f2
      assign_public_ip: yes
      instance_tags:
        Name: AnsibleNode
```

For Dynamic Host

8. Create IAM Admin user

9. Install python-pip on ansible master as root user

sudo apt-get install python-pip

10. Install boto by using pip on ansible master as root user

sudo pip install boto

11. Declare IAM user details on .boto file on ansible master as MAHA user

$ vi .boto

```
[Credentials]

aws_access_key_id = AKIAQ447TSANUDD6CUNM

aws_secret_access_key =
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
```

12 create ec2.py and ec2.ini file in ansible master as maha user

13) **change 755 file permission of ec2.py and ec2.ini**

$ chmod 755 ec2*

14 set class path of ec2.py and ec2.ini

$ vi /etc/profile

export EC2_INI_PATH=/home/maha/ec2.ini

export ANSIBLE_HOSTS=/home/maha/ec2.py

:wq!

15) ./ec2.py --list