config --global user.name "rez s"
git config --global user.email "rezziman@yahoo.com"
git config --global --list

git status

git init

Fresh file
---
git add blue green

git commit -m "v1

git status

Modified file

Edit the the blue
git status
git add blue
git commit -m “v2”

Extra- how to unstage - git restore --staged blue
git
git log or git log --oneline

git show 63f5bb7dbc13037f3dbdbb87ec8750f12c9df4dc

Working github

1) Signup on github
2) Setup repo on githhub
3) git remote add origin https://github.com/devopstrainers1/pgdevops.git
4) git remote -v
5) git push -u origin master
6) Invite collaborators (co-workers), access
7) Then push , pull the changes

git clone <URL>
git push -u origin master

Conflicts between local and remote

User 1 - we push changes on github (change -> commit)

User 2 - we should directly update same (which we have done from system-1) and try commit on local and push on github

Branching

1) git checkout -b features
2) We have update the changes in features
3) We have pushed the changes on features branch on github
4) We want no one should update the master branch in github directly (without PR)
5) Protect the master branch
6) Raise PR
7) Once PR approved, we can merge the changes.

Hands-on

1) Pull of master branch (on local)
2) Take a cut of master on new branch (on local)
3) Make changes on new branch (on local)
4) Push new branch (from local)
5) Raise PR to merge new branch with master (web)

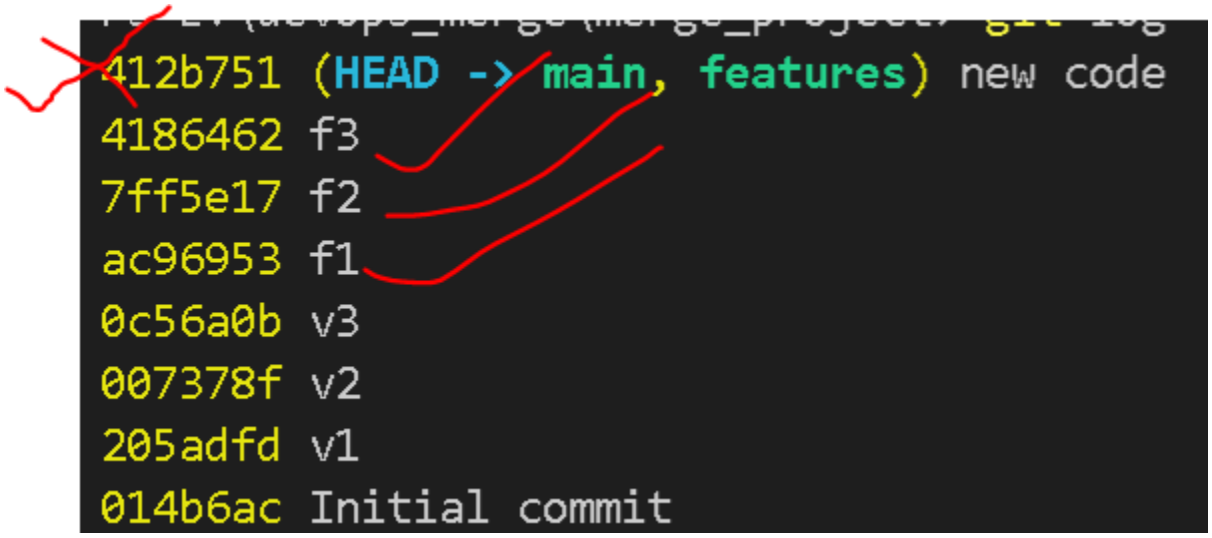
Rebase

git rebase main (we are on features branch)

```
96815a7 (HEAD -> features) f2
b6ef04c f1
7354b3e v3
a3e1889 v1
14b94ee v1
3973e1c Initial commit
PS E:\rebase_devops\rebase_devops>
PS E:\rebase_devops\rebase_devops> git rebase main
Successfully rebased and updated refs/heads/features.
PS E:\rebase_devops\rebase_devops>
PS E:\rebase_devops\rebase_devops> git log --oneline
3bafe40 (HEAD -> features) f2
1cd526b f1
c0e23c6 (main) all
7354b3e v3
a3e1889 v1
14b94ee v1
3973e1c Initial commit
```

Merge

git merge features (we are on master) , using PR



git rebase -i HEAD~5

git tag work_completed a801c05

#Git stash without message
Git stash ----------(stash work from staging area to be used later)

Git stash -m"message"

Git stash list

Git stash pop ---------- (pop last stash only one)

Git stash apply ----------(applied and need to manually drop git stash list)

Git stash drop ----------(will drop git stash list one at time)

Git stash clear ----------(remove all stash work)

Note :-
stash@{0} is latest stash
#pop and apply to staging area
git stash pop stash@{2}

#will revert commit and create new commit on top (nothing will be deleted it can be treat as undo)
Git revert <commithash> ----------(use hash above your hash that need to be reverted)

#will remove work from database area only ,so we have chance to modify and recommit
Git reset --soft HEAD~<number from top to down commits shown in git log>

#will remove work from workspace,staging,database area all (risky)
Git reset --hard HEAD~<number from top to down commits shown in git log>

#git ignore
Easy way to generate your own .gitignore file for any language:
http://gitignore.io

This for next class

Maven
Java
Gradle
Shell scripting
Power shell

==========

How to install Java (JDK) on Ubuntu

Step 1 – Search OpenJDK Packages
$ sudo apt-cache search openjdk

Step 2 – Install JAVA (OpenJDK)
$ sudo apt-get install openjdk-8-jre openjdk-8-jdk

Step 3 – Configure Default Java Version
$sudo java -version

Step 4 – Set JAVA_HOME
$echo "JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64/" >> /etc/environment
if you get permission error use -> sudo vi /etc/environment to write the JAVA_HOME variable
$source /etc/environment
$echo $JAVA_HOME

How to install Jenkins on Ubuntu

Official doc - https://www.jenkins.io/doc/book/installing/linux/

Long Term Support release:

1. wget -q -O - https://pkg.jenkins.io/debian-stable/jenkins.io.key | sudo apt-key add -
2. sudo sh -c 'echo deb https://pkg.jenkins.io/debian-stable binary/ > /etc/apt/sources.list.d/jenkins.list'
3.
4. sudo apt-get update
5. sudo apt-get install jenkins
6. sudo systemctl restart jenkins
7. sudo systemctl status jenkins
8. sudo netstat -tnlup | grep java
9.

http://localhost:8080

Run -> $ sudo cat /var/lib/jenkins/secrets/initialAdminPassword

All Jenkins Plugins available link:
https://plugins.jenkins.io/

Install Plugins

- Git
- Maven (Maven Integration)

How to Install Maven

1) sudo apt install maven
2) mvn -version

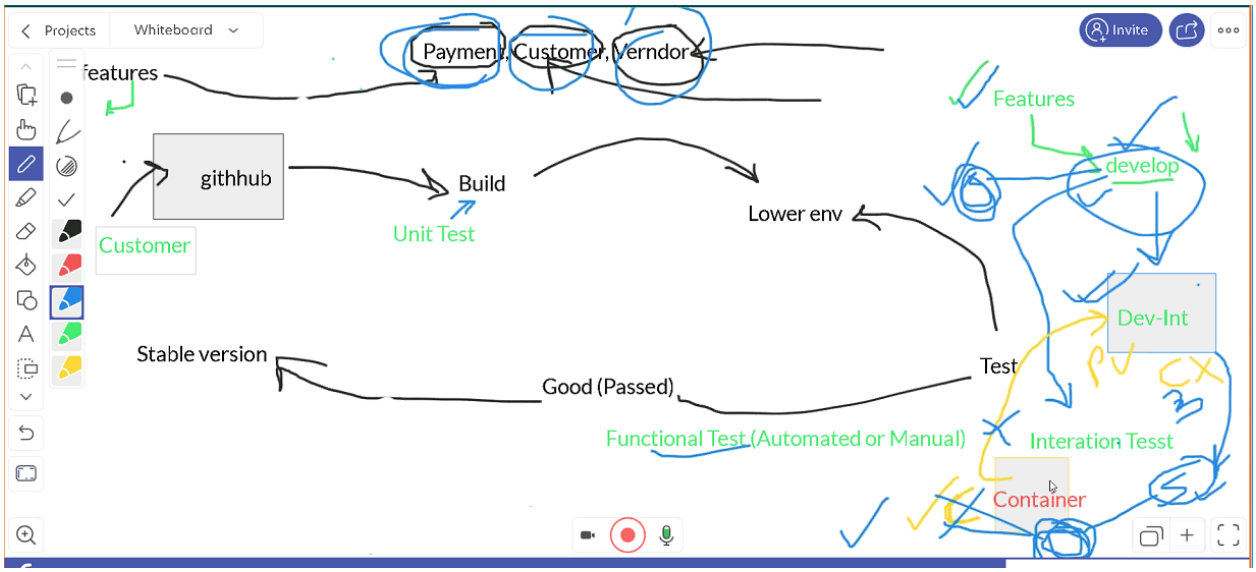
Or

1) wget https://www-us.apache.org/dist/maven/maven-3/3.6.0/binaries/apache-maven-3.6.0-bin.tar.gz -P /tmp

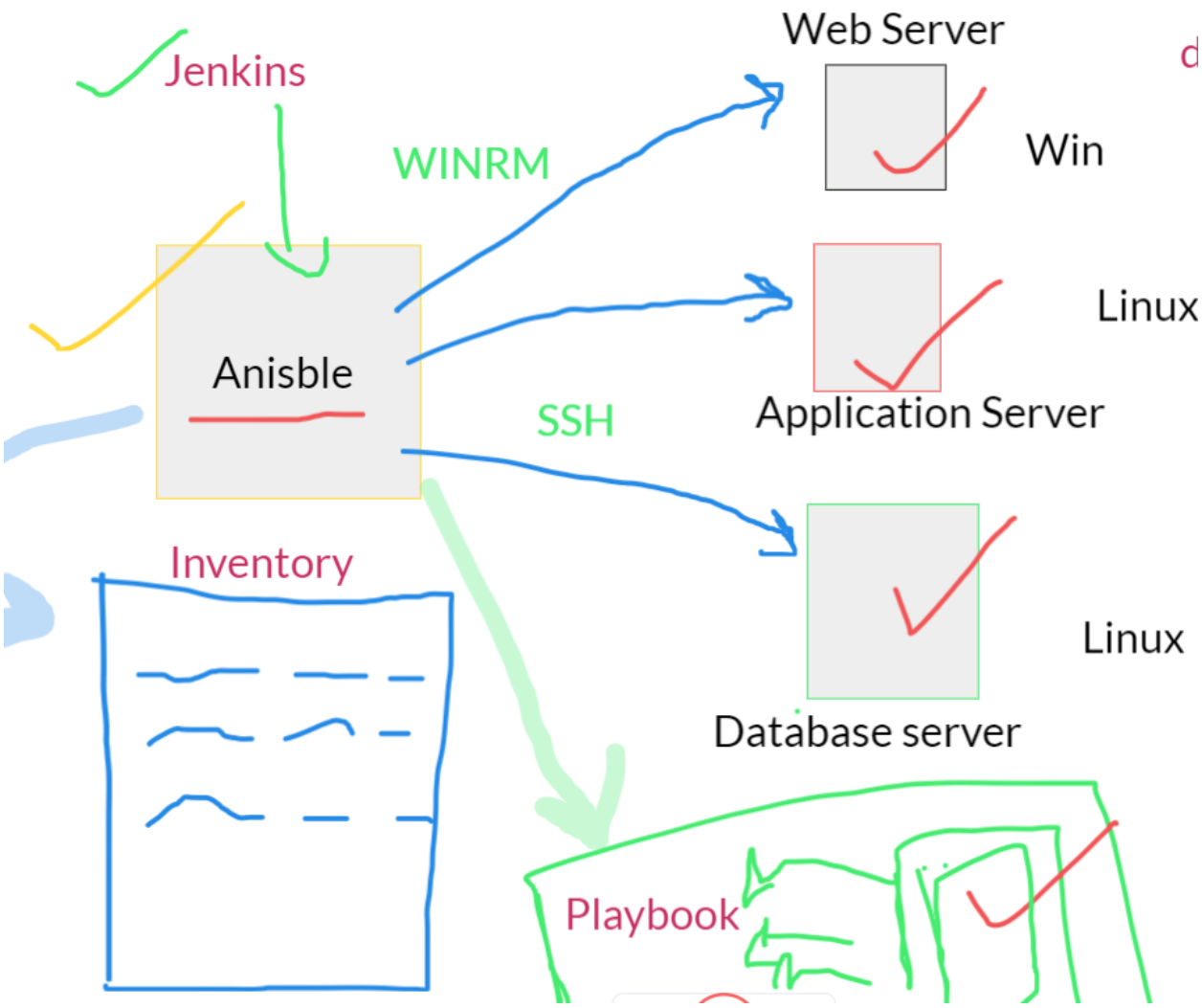
2) sudo tar xf /tmp/apache-maven-*.tar.gz -C /opt
3) sudo ln -s /opt/apache-maven-3.6.0 /opt/maven
4) sudo nano /etc/profile.d/maven.sh

```
export JAVA_HOME=/usr/lib/jvm/default-java
export M2_HOME=/opt/maven
export MAVEN_HOME=/opt/maven
export PATH=${M2_HOME}/bin:${PATH}
```

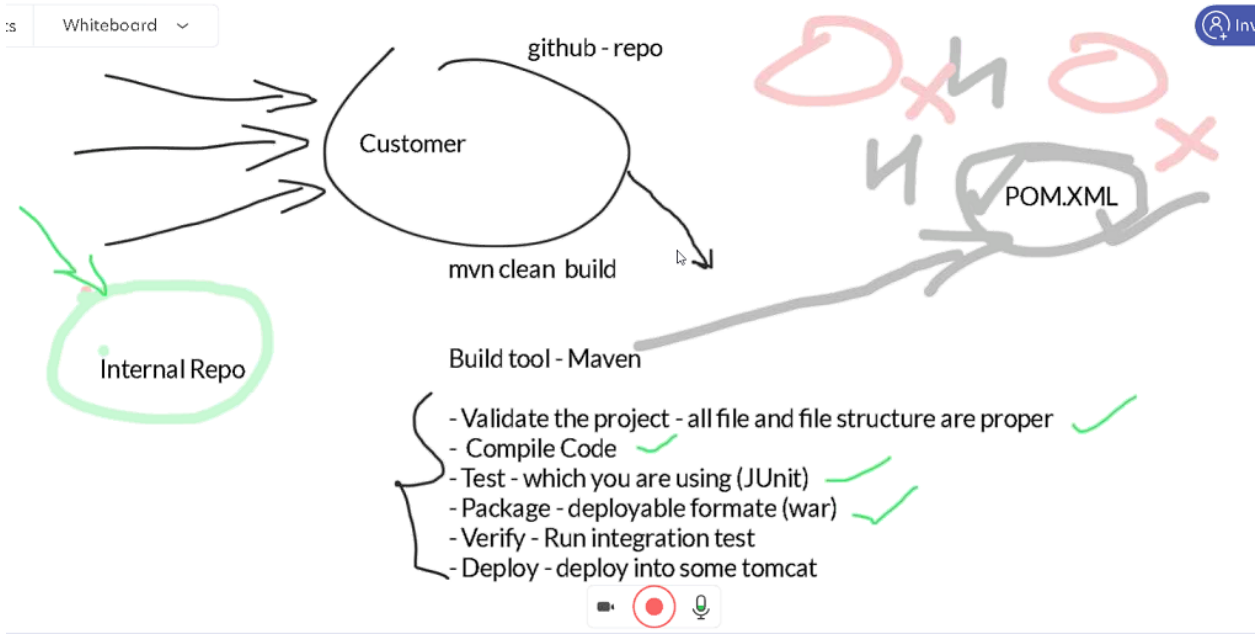
5)
6)
7) source /etc/profile.d/maven.sh
8) mvn -version



Ansible



Installation - https://docs.ansible.com/ansible/latest/installation_guide/intro_installation.html



POM.xml - Project Object Model. It's the xml file that manages, adds, removes dependencies in maven.

Install below to 2 plugin in Jenkins using Manage Plugins

1) Git
2) Maven (Mav

**if anyone is loggedout of jenkins without setting up Admin user/pass:

1. sudo Systemctl stop jenkins
2. sudo vi var/lib/jenkins/config.xml
3. <useSecurity>true</useSecurity>---> <useSecurity>false</useSecurity>
4. sudo systemctl start jenkins
5. Setup admin account.
6. sudo Systemctl stop jenkins
7. <useSecurity>false</useSecurity>--> <useSecurity>true</useSecurity>
8. sudo systemctl start jenkins

Steps to take if you forget password issue:
sudo su
Vi /var/lib/jenkins/config.xml
<useSecurity>false</useSecurity>

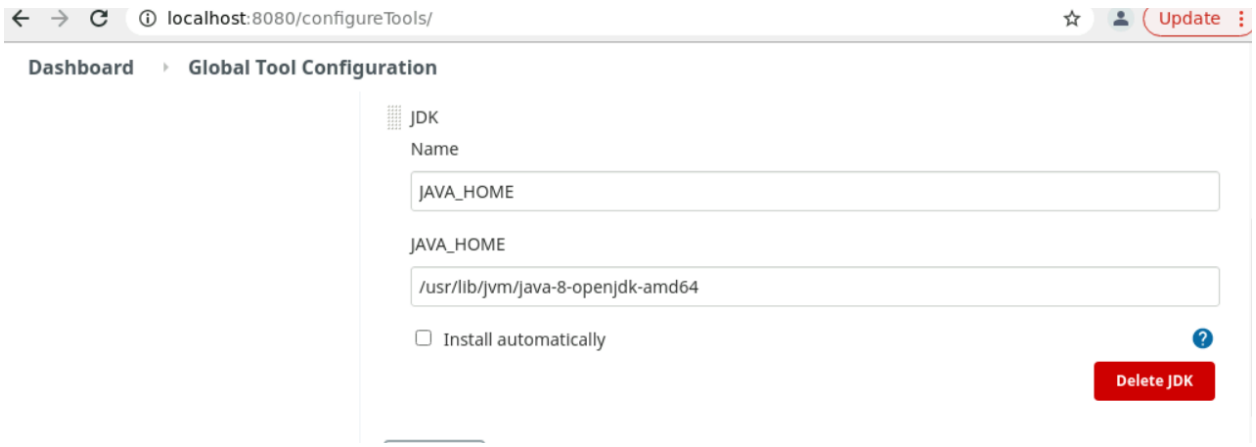
systemctl restart jenkins

Git location for project using Jenkins freestyle maven:
https://github.com/devopstrainers1/addressbook.git

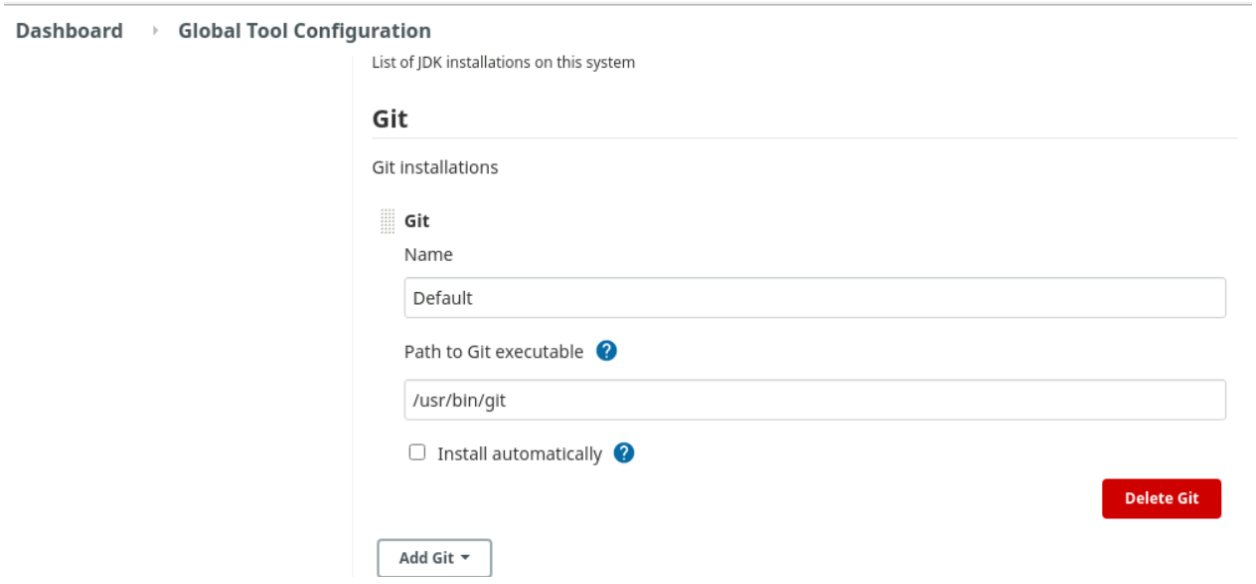
/var/lib/jenkins/workspace will create a folder for the above project and/or every project created in Jenkins.

/var/lib/jenkins/tools will create a folder for the softwares Jenkins install

JAVA HOME: /usr/lib/jvm/java-8-openjdk-amd64/



Git :



Maven: Run => *maven -v* command to check maven_home path.

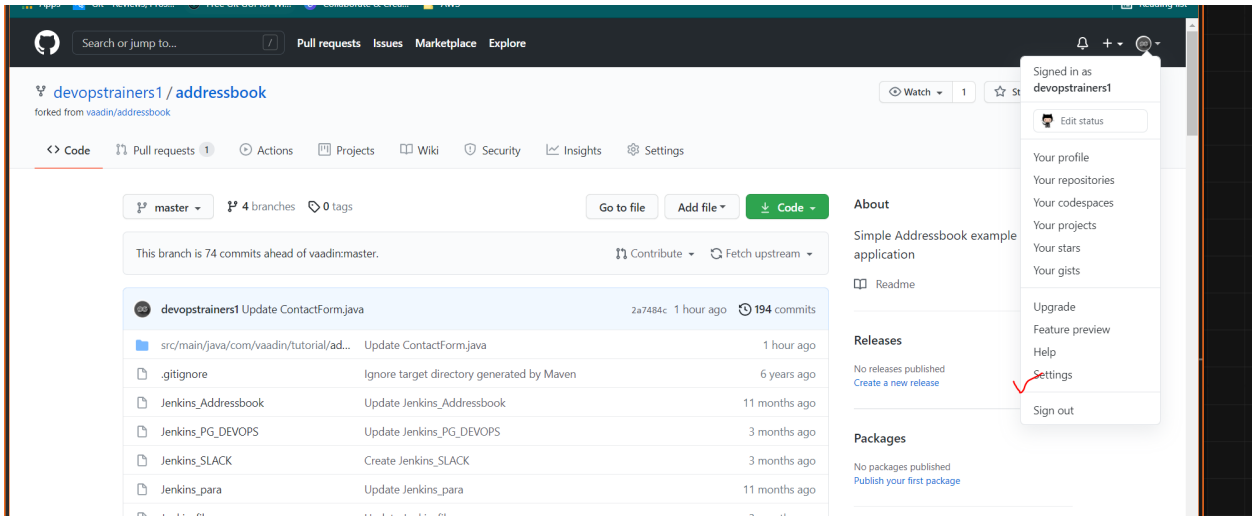
Different goals of Maven:validate, Compile, test, package, verify
** Add ‘Build Pipeline’ Plugin, Pipeline: Job.

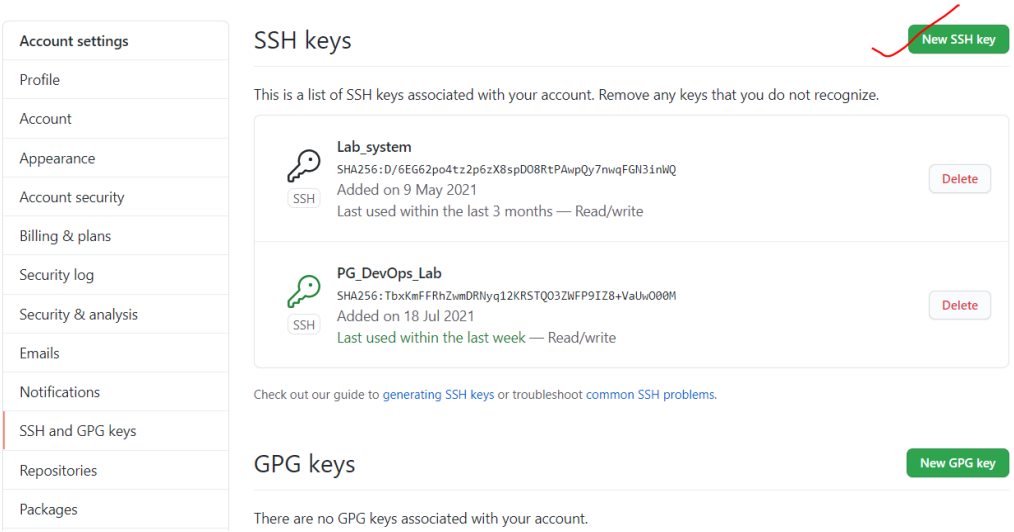
Upstream: before the job
Downstream: After the job

Unstable Build: one success build just after one failed build.
Stable build: 2nd success build after one build failed.

Build Sequence:
No upstream->validate->compile->test->package->verify->no downstream

ssh-keygen





We have to copy the id_rsa.pub in to new key

We need to add the remote repo using ssh instead https
git remote add origin ssh@path
git push -u origin master

Adding Active Directory:

1) Install Active Directory Plugin.

We can install windows server ISO file and then setup AD server.

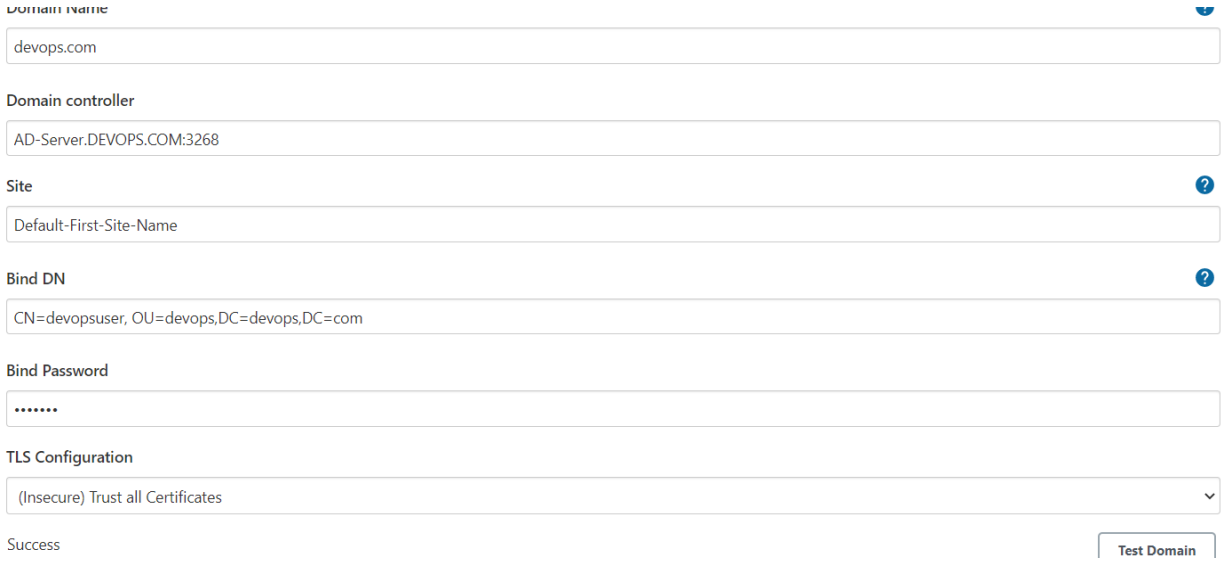
https://www.microsoft.com/en-US/evalcenter/evaluate-windows-server-2019?filetype=ISO

https://www.dummies.com/programming/networking/network-administration-active-directory-organization-units/

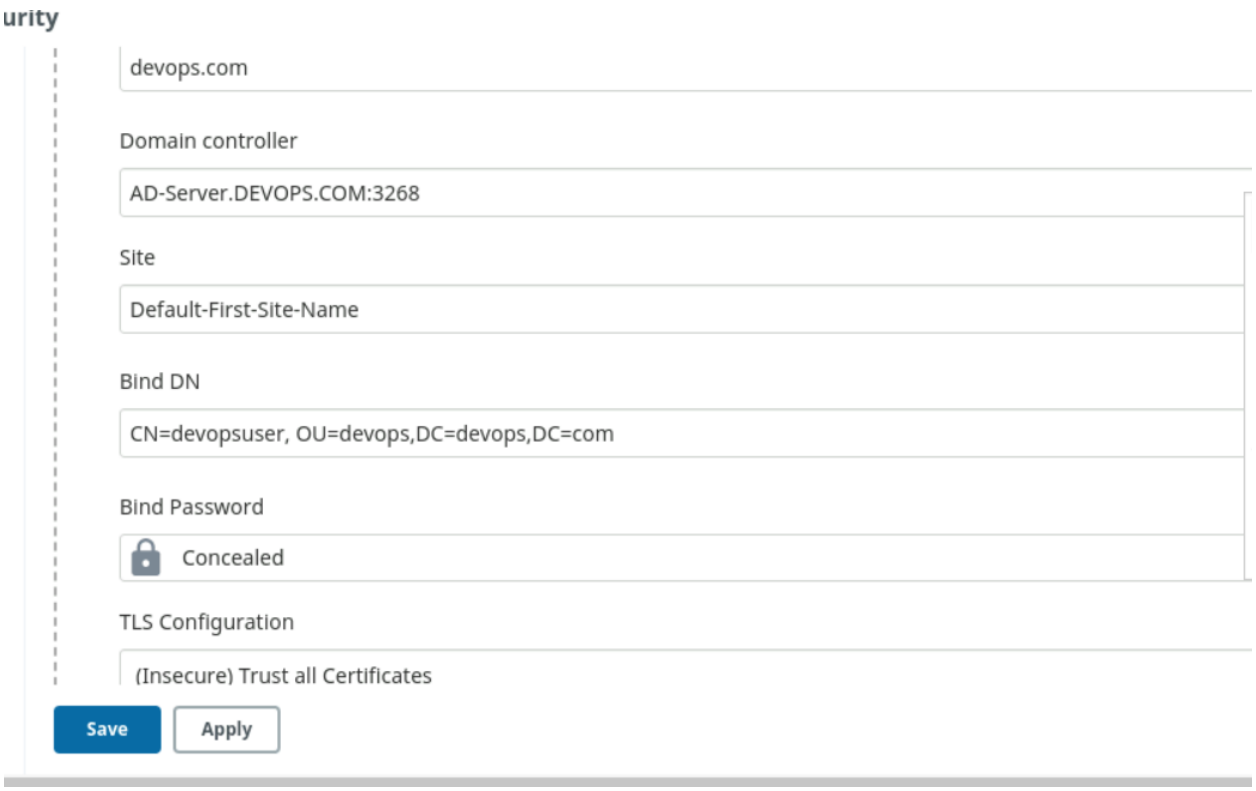
https://www.itingredients.com/create-ou-in-active-directory/

**Jenkions AD user in demo was :1) devopsuser, password: dev!@12

2) testuser and password -test!@12



Domain Name: devops.com
Domain controller: AD-Server.DEVOPS.COM:3268
or 20.55.9.109:3268
Site: Default-First-Site-Name
Bind DN: CN=devopsuser, OU=devops,DC=devops,DC=com
Bind Password: dev!@12

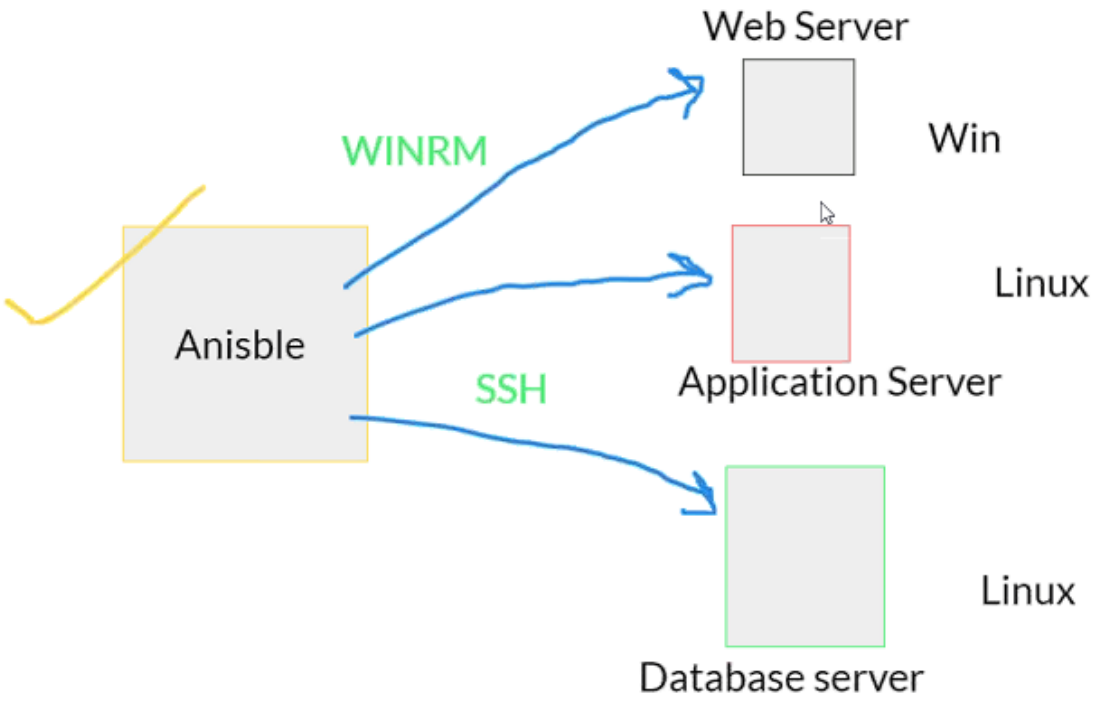


Role-based Authorization Strategy

Example of users in AD
username:-testuser
Password:-test!@12
username:-testuser1
Password:-test!@12

Check on which port ssh service is running.
cat /etc/ssh/sshd_config | grep Port
sudo netstat -tnlup | grep ssh

Step by step doc
https://docs.google.com/document/d/1U2e-ha1vnkWx622aLjG9oC_umcFYIXIb5HmOURv-wms/edit?usp=sharing



SSH to your machine:
ssh -p 42006 <user>@yogeshyogeshsansl0018.simplilearnlabs.com

Ansible Inventory example
1) create inventory file "PG_DEVOPs" in ansible directory (/etc/ansible/PG_DEVOps)

root@ip-172-31-79-36:~# cat PG_DEVOPS
[frontend]
Anshwini ansible_host=172.31.69.134 ansible_ssh_user=ma*** ansible_ssh_pass='XXXXX''
ansible_connection=ssh ansible_port=42006 ansible_ssh_common_args='-o
StrictHostKeyChecking=no'

Ajay ansible_host=172.31.79.36 ansible_ssh_user=XXXX ansible_ssh_pass='XXXXX'
ansible_connection=ssh ansible_port=42006 ansible_ssh_common_args='-o
StrictHostKeyChecking=no'
#server2
#server3

2) add details above to PG_DEVOPS with vi or vim:
3) Run :

ansible -i PG_DEVOPS frontend -m ping

ansible -i PG_DEVOPS frontend -m ping

host_key_checking = False

[all:vars]
ansible_connection=ssh
ansible_port=42006
ansible_ssh_common_args='-o StrictHostKeyChecking=no'

[frontend:vars]
ansible_connection=ssh
ansible_port=42006
ansible_ssh_common_args='-o StrictHostKeyChecking=no'

[frontend]
Anshwini ansible_host=172.31.69.134 ansible_ssh_user=XXXX ansible_ssh_pass='XXX’
Ajay ansible_host=172.31.79.36 ansible_ssh_user=XXXX ansible_ssh_pass=’XXXX'
#server2
#server3

pip install winrm

https://stackoverflow.com/questions/43267157/python-attributeerror-module-object-has-no-attribute-ssl-st-init

https://github.com/ansible/ansible/blob/devel/examples/scripts/ConfigureRemotingForAnsible.ps

Ajay's Finding

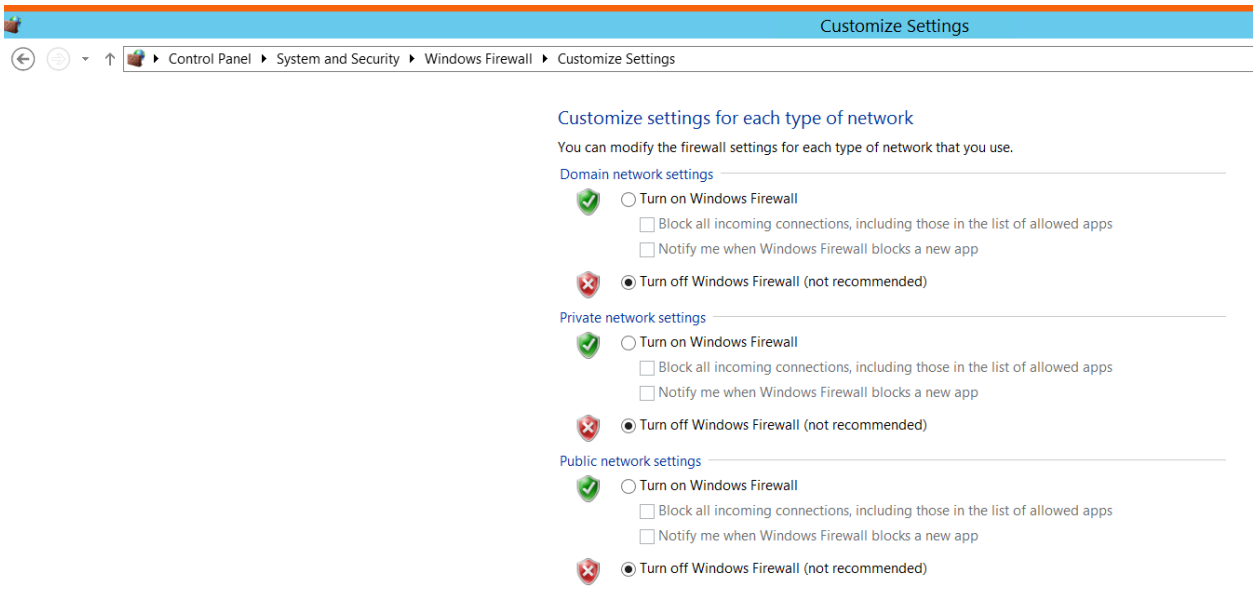
######################THIS IS FIXED ######################

I have left the Azure Window server power on for you, the inventory for window server file is place down in this doc

We don't need to run the ps1 (powershell) script since is already done,

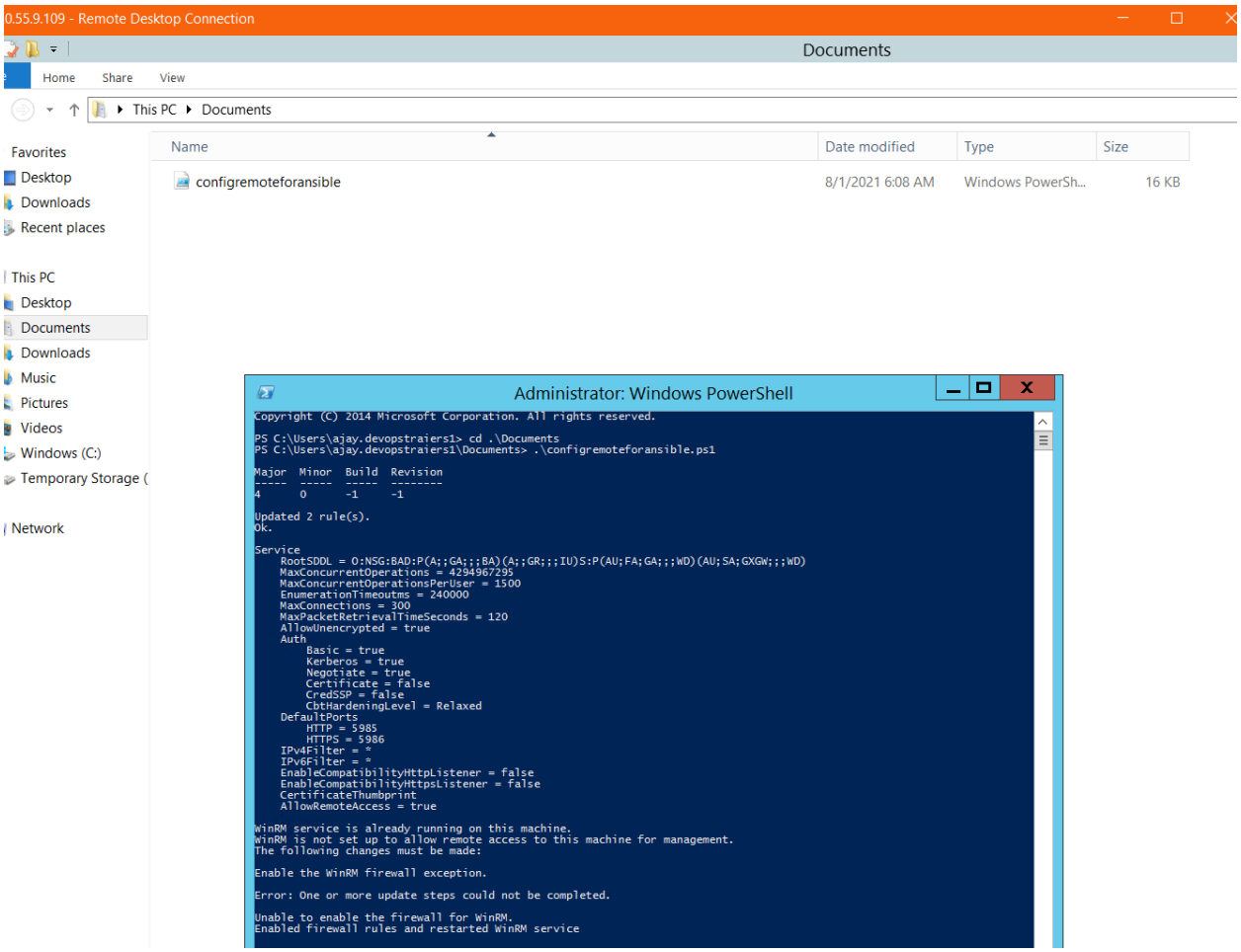
The working ps1 script we can download from the github link given below.

I have disabled the window firewall as a quick fix, but we can allow the right port for winrm in firewall.

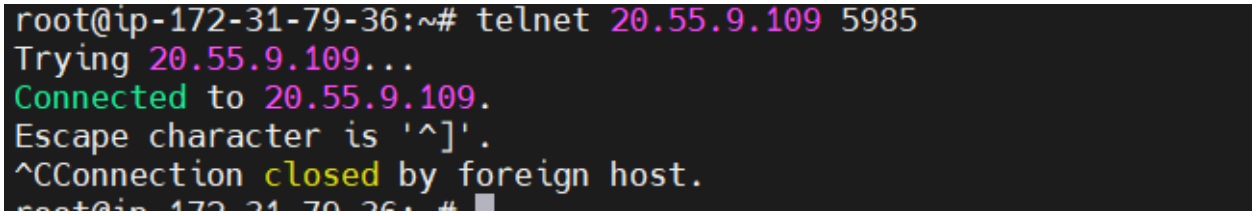


The winrm script place here
https://github.com/devopstrainers1/pgdevops/blob/devopstrainers1-patch-1/configremoteforansible.ps1

You can download the script on the window server and run the script as administrator from powershell terminal. (pls don't run on Azure VM since I already ran it)



We can check the winrm port is opened or not using below command, this we need to run from lab system (option step)



Step need to perform on lab system

1) pip list | grep ansible
2) apt list installed | grep ansible
3) pip uninstall ansible
4) apt remove ansible
5) pip remove ansible
6) `sudo apt update`
7) `sudo apt install software-properties-common`
8) `sudo add-apt-repository --yes --update ppa:ansible/ansible`
9) `sudo apt install ansible`
10) `ansible-galaxy collection install ansible.windows`

```
root@ip-172-31-79-36:~# ansible --version
[DEPRECATION WARNING]: Ansible will require Python 3.8 or newer on the controller starting with Ansible
11:38:31) [GCC 5.4.0 20160609]. This feature will be removed from ansible-core in version 2.12. Deprecat
deprecation_warnings=False in ansible.cfg.
/usr/local/lib/python2.7/dist-packages/ansible/parsing/vault/__init__.py:44: CryptographyDeprecationWarn
e team. Support for it is now deprecated in cryptography, and will be removed in the next release.
  from cryptography.exceptions import InvalidSignature
ansible [core 2.11.3]
  config file = /etc/ansible/ansible.cfg
  configured module search path = [u'/root/.ansible/plugins/modules', u'/usr/share/ansible/plugins/modul
  ansible python module location = /usr/local/lib/python2.7/dist-packages/ansible
  ansible collection location = /root/.ansible/collections:/usr/share/ansible/collections
  executable location = /usr/local/bin/ansible
  python version = 2.7.12 (default, Mar  1 2021, 11:38:31) [GCC 5.4.0 20160609]
  jinja version = 2.8
  libyaml = True
root@ip-172-31-79-36:~#
```

11) pip list | grep winrm

12) pip install winrm

13) ansible frontend -i PG_DEVOPS -m win_ping

```
root@ip-172-31-79-36:~# ansible frontend  -i PG_DEVOPS -m win_ping
[DEPRECATION WARNING]: Ansible will require Python 3.8 or newer on the controller starting with Ansible
11:38:31) [GCC 5.4.0 20160609]. This feature will be removed from ansible-core in version 2.12. Deprec
deprecation_warnings=False in ansible.cfg.
/usr/local/lib/python2.7/dist-packages/ansible/parsing/vault/__init__.py:44: CryptographyDeprecationWar
e team. Support for it is now deprecated in cryptography, and will be removed in the next release.
  from cryptography.exceptions import InvalidSignature
Azure | SUCCESS => {
    "changed": false,
    "ping": "pong"
}
root@ip-172-31-79-36:~#
```

The inventory file for the window system looks like this.

```
[frontend]
Azure ansible_host=20.55.9.109 ansible_user=ajay.devopstraiers1
ansible_password='Hnd3YCTlQBvwHdUEcUEwxER12II1P30='
ansible_connection=winrm ansible_port=5985
ansible_winrm_server_cert_validation=ignore
```

```
apt remove python
apt remove ansible
apt install python
pip install ansible
```

```
ansible frontend_win -i PG_DEVOPS -m win_copy -a "src=devops_pg dest=D:\\"
```

```
ansible frontend -i PG_DEVOPS -m copy -a "src=devops_pg dest=/tmp"

 ansible frontend_win -i PG_DEVOPS -m win_file -a "path=D://ajay  state=directory"

ansible frontend -i PG_DEVOPS -a "df -h"==>file system used
ansible frontend -i PG_DEVOPS -a "free -m"

 ansible frontend -i PG_DEVOPS -b  -m apt -a "name=apache2 state=present"

ansible frontend -i PG_DEVOPS  --become-user "ganesh"  -m apt -a "name=apache2
state=present"
```

```
---
- hosts: frontend
  become: yes
  become_user: root
```

```
  tasks:
    - name: Create file
      file:
        path: /etc/foo.conf
        state: touch

    - name: Change file ownership, group and permissions
      file:
        path: /etc/foo.conf
        mode: '0644'

    - name: Touch a file, using symbolic modes to set the permissions (equivalent to
0644)
      file:
        path: /etc/ajay.conf
        state: touch
        mode: u=rw,g=r,o=r


ansible-playbook -i PG_DEVOPS myplaybook.yml


https://docs.ansible.com/ansible/2.9/modules/file_module.html#file-module




---
- hosts: frontend
  become: yes
  become_user: root
  vars:
    server_port: 8090
    worker_con: 1024

  tasks:
    - name: "Install the package"
      apt:
        name: nginx
        state: present
      ignore_errors: yes
```

```
  - name: "copy content"
    copy:
      src: index.html
      dest: /usr/share/nginx/html/index.html

  - name: "apply config"
    template:
      src: nginx.conf.j2
      dest: /etc/nginx/nginx.conf
      notify: "Restart service if config updated"

  - name: "apply config"
    template:
      src: port.j2
      dest:  /etc/nginx/sites-enabled/default
      notify: "Restart service if config updated"

handlers:
  - name: "Restart service if config updated"
    service:
      name: nginx
      state: restarted
```

Command to Run The nginx
ansible-playbook -i PG_DEVOPS deploy_web.yml

Command to be remove nginx completely:
sudo apt-get purge nginx nginx-common
2222

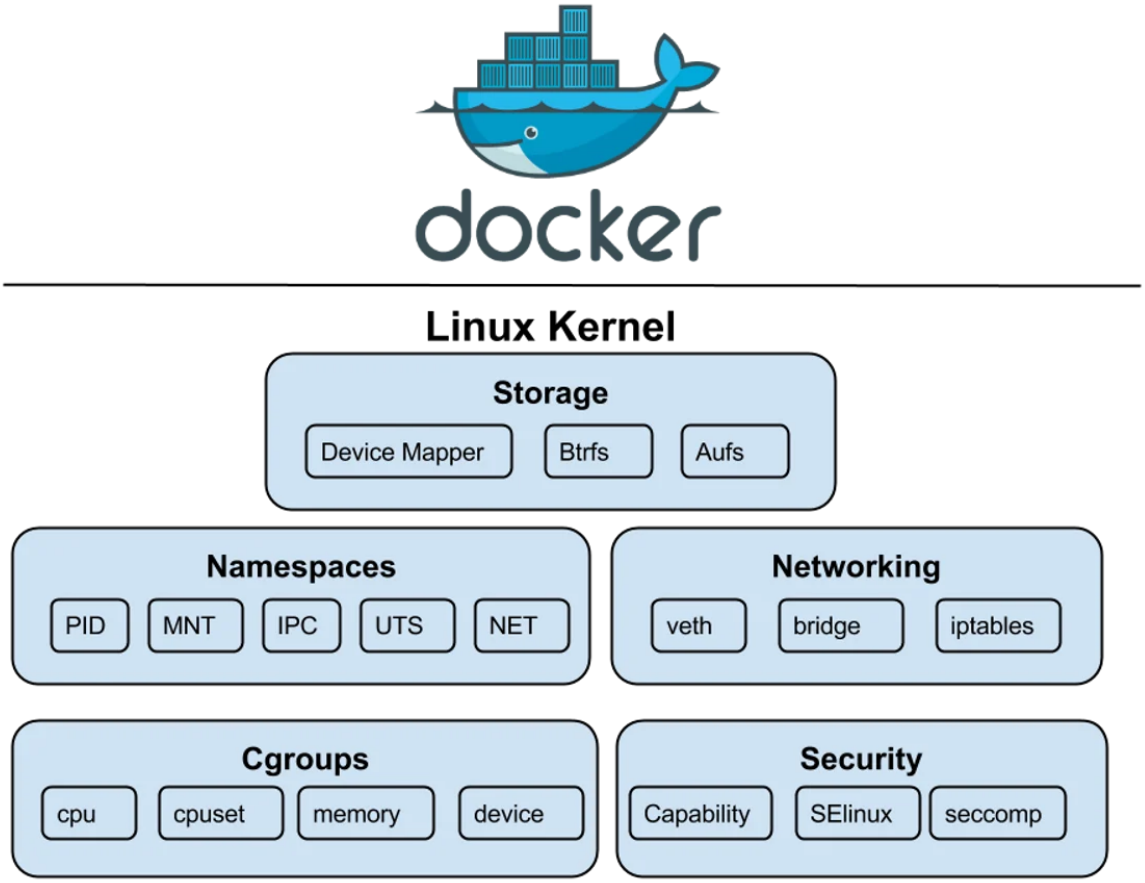
Ansible Vault

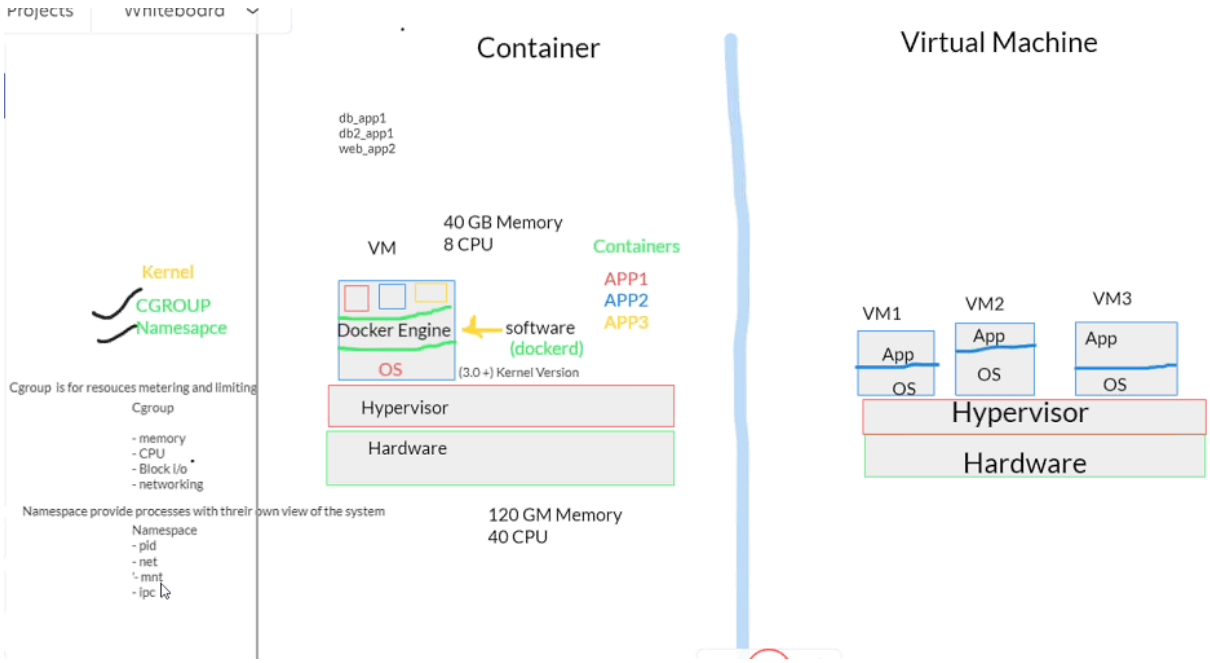
ansible-vault encrypt PG_DEVOPS(inventry file)
ansible-vault view /etc/ansible/hosts
ansible-vault edit /etc/ansible/hosts
ansible-playbook --ask-vault-pass nginx.yml
ansible-vault decrypt /etc/ansible/hosts

ansible-playbook deploy_web.yml -i PG_DEVOPS --ask-vault-pass

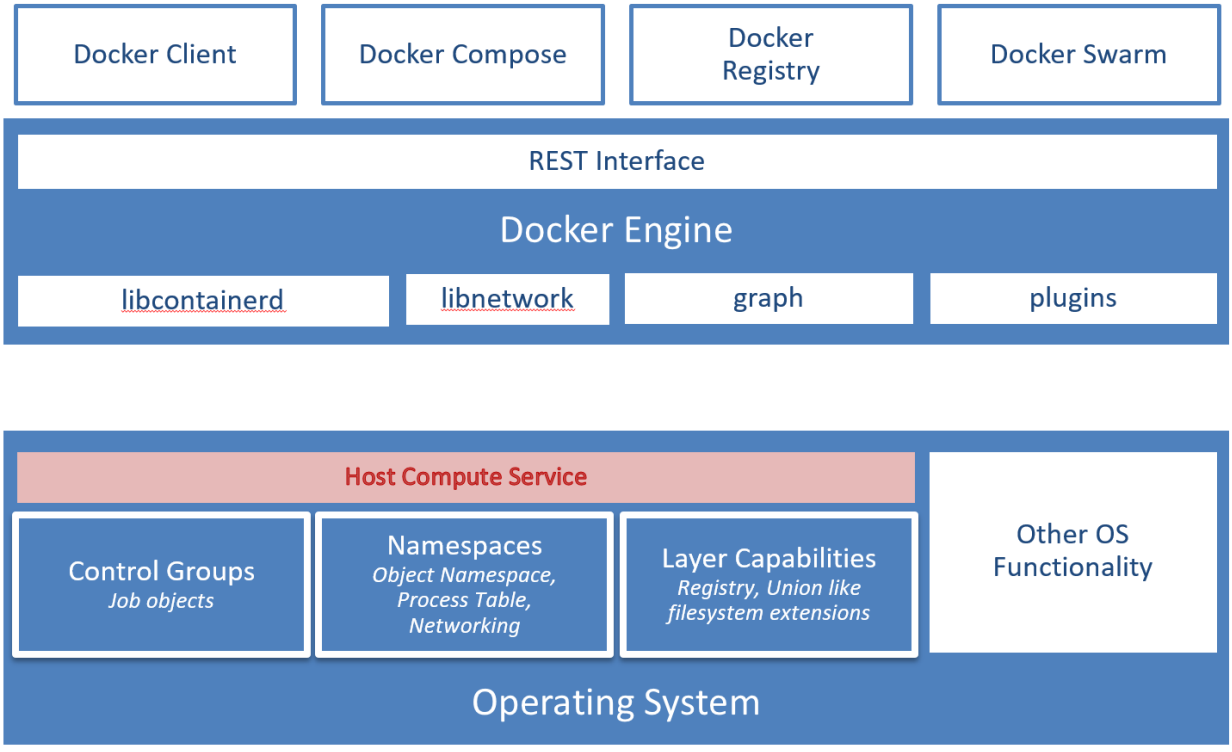
Some things to try to stabilize python environment so ansible-vault works:

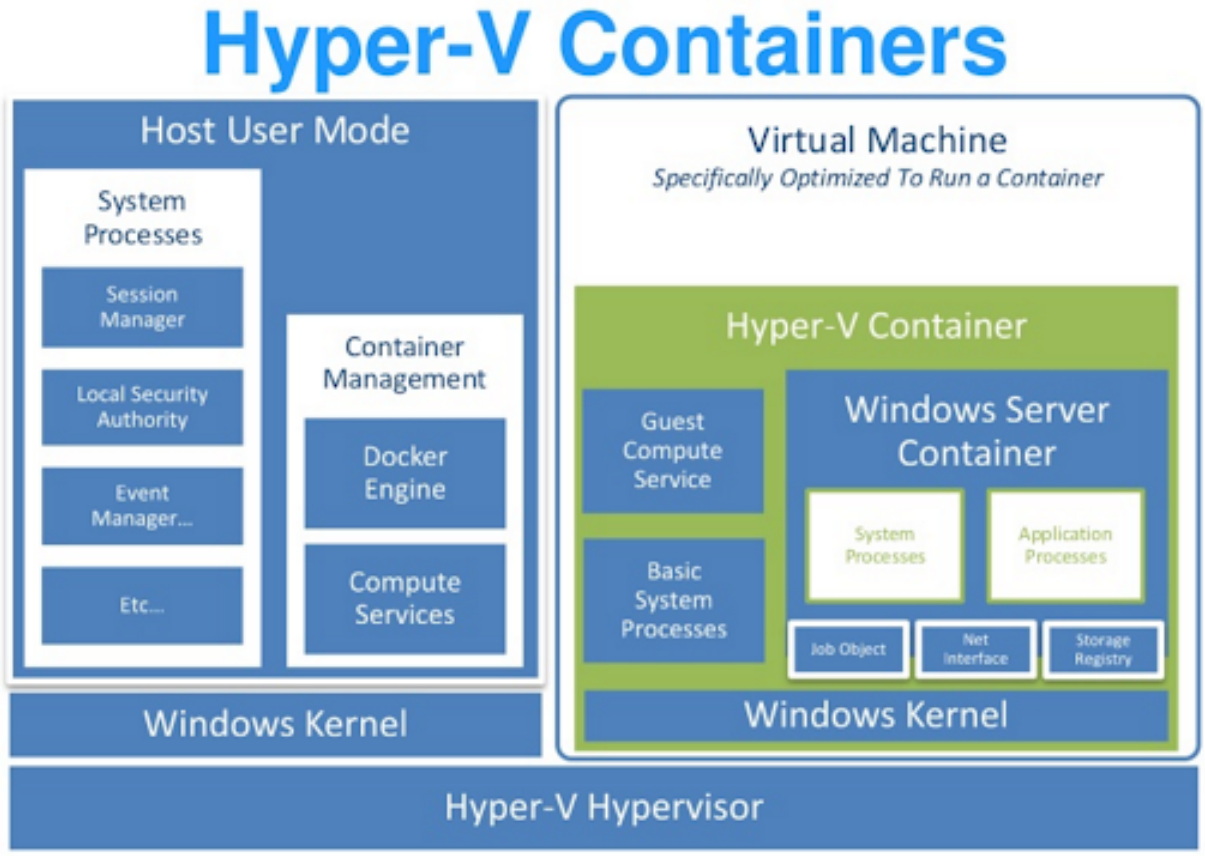
- **Install python3.9**
- **pip3.9 install pycrypto**
- **pip3.9 install cryptography**
- **pip3.9 install ansible**



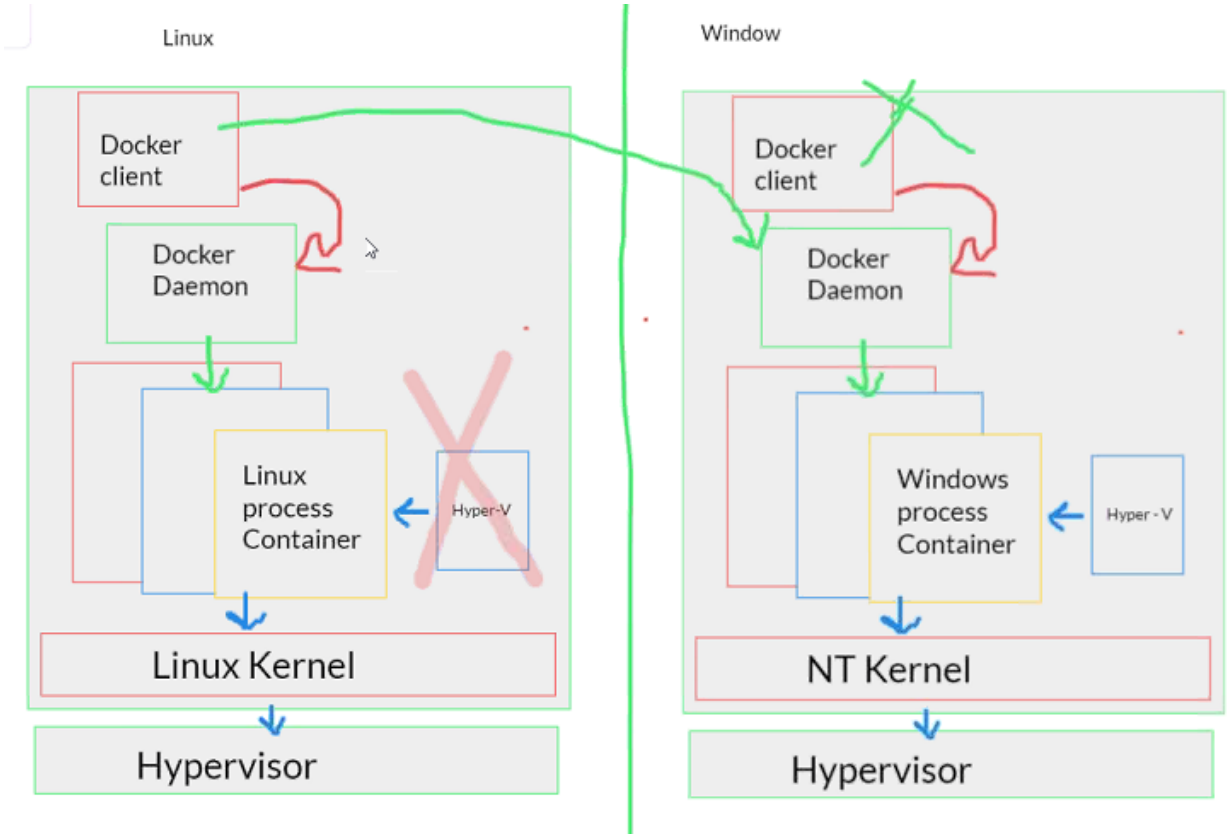


Architecture In Windows





***** Docker client can communicate with windows or linux containers**



LCOW (Linux Container on Windows)

https://www.docker.com/blog/preview-linux-containers-on-windows/

https://docs.docker.com/engine/install/ubuntu/

```
sudo apt-get remove docker docker-engine docker.io containerd runc

sudo apt-get update


$ sudo apt-get install \

    apt-transport-https \

    ca-certificates \

    curl \
```

```
    gnupg \
    lsb-release


curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg
--dearmor -o /usr/share/keyrings/docker-archive-keyring.gpg


echo \
  "deb [arch=amd64
signed-by=/usr/share/keyrings/docker-archive-keyring.gpg]
https://download.docker.com/linux/ubuntu \
  $(lsb_release -cs) stable" | sudo tee
/etc/apt/sources.list.d/docker.list > /dev/null


sudo apt-get update


sudo apt-get install docker-ce docker-ce-cli containerd.io



sudo docker run hello-world


systemctl status docker

https://hub.docker.com/

 docker pull centos:6



docker images
```

https://docs.ansible.com/ansible/latest/installation_guide/intro_installation.html#installing-ansible-on-rhel-centos-or-fedora

```
# the command below can run command and pull images if it doesn't exist
# latest keyword will pull the latest version from the official image
docker run -it centos:latest
Docker run --name ansibleimage -it centos:7

# Install ansible on centos
 yum install epel-release
 yum install ansible

yum install git
git clone https://github.com/devopstrainers1/pgdevops.git
(need to specify the branch above - no longer has all the files (git clone --branch <branchname> <remote-repo-url>
))
git checkout features
rm -rf Ashwini.txt blue demo_ajay.txt filefromjeeshan helloworld justOneMore rainbow rooFile yellow purple  red Purple2 demo.txt  green pgdevops JMConfig.py
ansible frontend -i PG_DEVOPS  -m ping

ansible-vault encrypt PG_DEVOPS
ansible --ask-vault-pass frontend -i PG_DEVOPS -m ping
ansible-playbook deploy_web.yml  -i PG_DEVOPS --ask-vault-pass
ansible --vault-password-file pass.txt frontend -i PG_DEVOPS -m ping
```

git status

```
[root@e4fa357dfc84 pgdevops]# git status
# On branch ansible
# Changes to be committed:
#   (use "git reset HEAD <file>..." to unstage)
#
#       deleted:    Ashwini.txt
#       deleted:    JMConfig.py
#       deleted:    Purple2
#       deleted:    blue
#       deleted:    demo.txt
#       deleted:    demo_ajay.txt
#       deleted:    filefromjeeshan
#       deleted:    green
#       deleted:    helloworld
#       deleted:    justOneMore
#       deleted:    pgdevops
#       deleted:    purple
#       deleted:    rainbow
#       deleted:    red
#       deleted:    rooFile
#       deleted:    yellow
#
```

git add -u .

git commit -m "done"

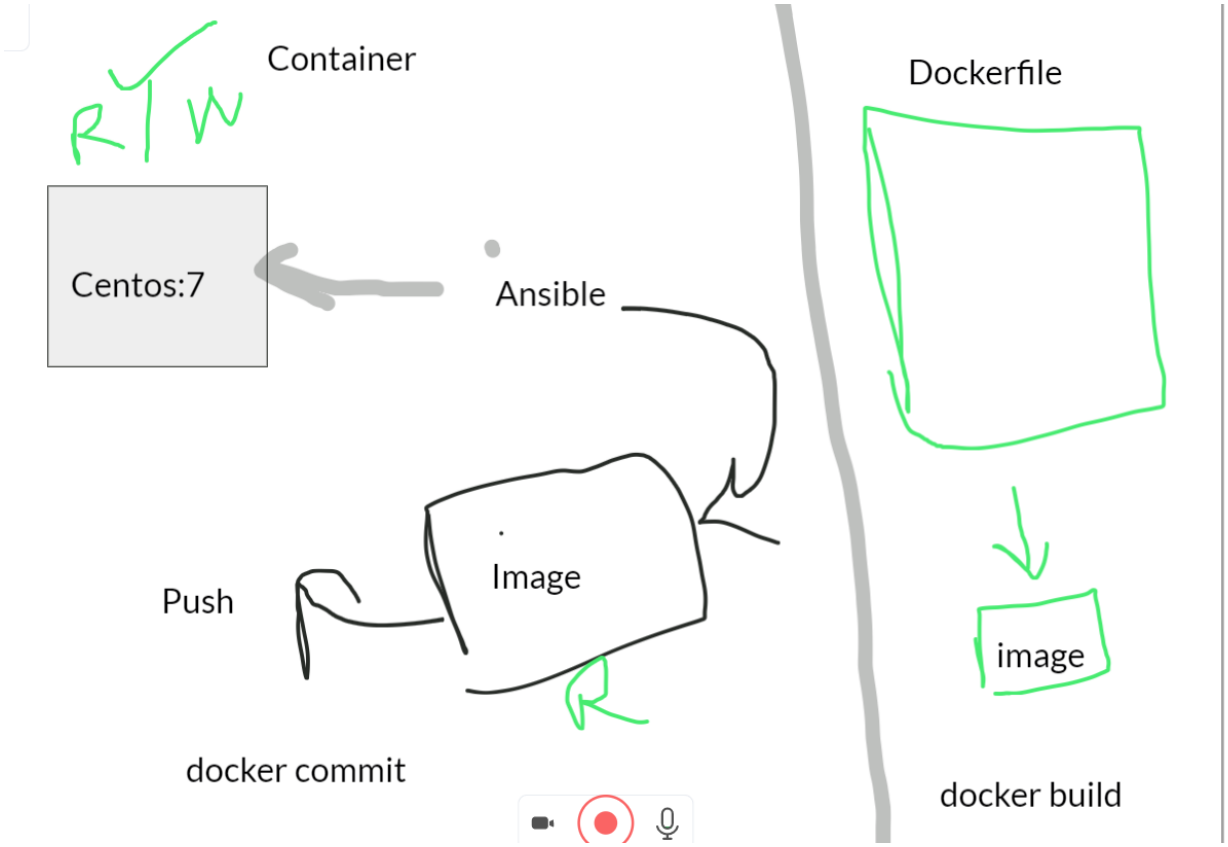
Ansible with Jenkins with vault password

Existing from container to main terminal (your host m/c) -> it's will make sure container still running (not exit)

Ctrl+pq / exit (command)

docker ps (only running container)

docker ps -a (all container)



 # Next 2 steps are key to creating images and pushing it to docker hub

docker commit e4fa357dfc84 ajaycs/ansible:v1

docker commit container_id repo_name/image_name:tag

```
root@ip-172-31-79-36:~# docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
ajaycs/ansible      v1                  e88b84ce8c23        About a minute ago   553MB
centos              7                   8652b9f0cb4c        8 months ago        204MB
hello-world         latest              bf756fb1ae65        19 months ago       13.3kB
centos              6                   d0957ffdf8a2        2 years ago         194MB
root@ip-172-31-79-36:~#
```

docker login

```
root@ip-172-31-79-36:~#
root@ip-172-31-79-36:~# docker login
Login with your Docker ID to push and pull images from Docker Hub. If you don't have a Docker ID, head over to https://hub.docker.com to create one.
Username: ajaycs
Password:
WARNING! Your password will be stored unencrypted in /root/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
root@ip-172-31-79-36:~#
```

docker push ajaycs/ansible:v1

```
root@ip-172-31-79-36:~# docker push ajaycs/ansible:v1
The push refers to repository [docker.io/ajaycs/ansible]
1a8132bbb31d: Pushing [=======================>                           ]  159.9MB/349.2MB
174f56854903: Mounted from library/centos
```

https://github.com/devopstrainers1/pgdevops/tree/ansible

https://docs.docker.com/engine/reference/builder/

docker ps=>To check if any image is running

Docker ps -a==>all containers

mkdir docker_work

cd docker_work

vi Dockerfile

 # Inside Dockerfile copy the commands below

```
#Start of Dockerfile

FROM centos:7

RUN yum install epel-release  -y

RUN yum install ansible git -y
```

```
RUN mkdir ansible_work

RUN cd ansible_work && git clone https://github.com/devopstrainers1/pgdevops.git

# End of Dockerfile
```

```
docker build -t ajaycs/dockerfile_ansible:v2 .

docker run -it ajaycs/dockerfile_ansible:v2
```

```
docker logs <container_id>

docker inspect <image_id>
```

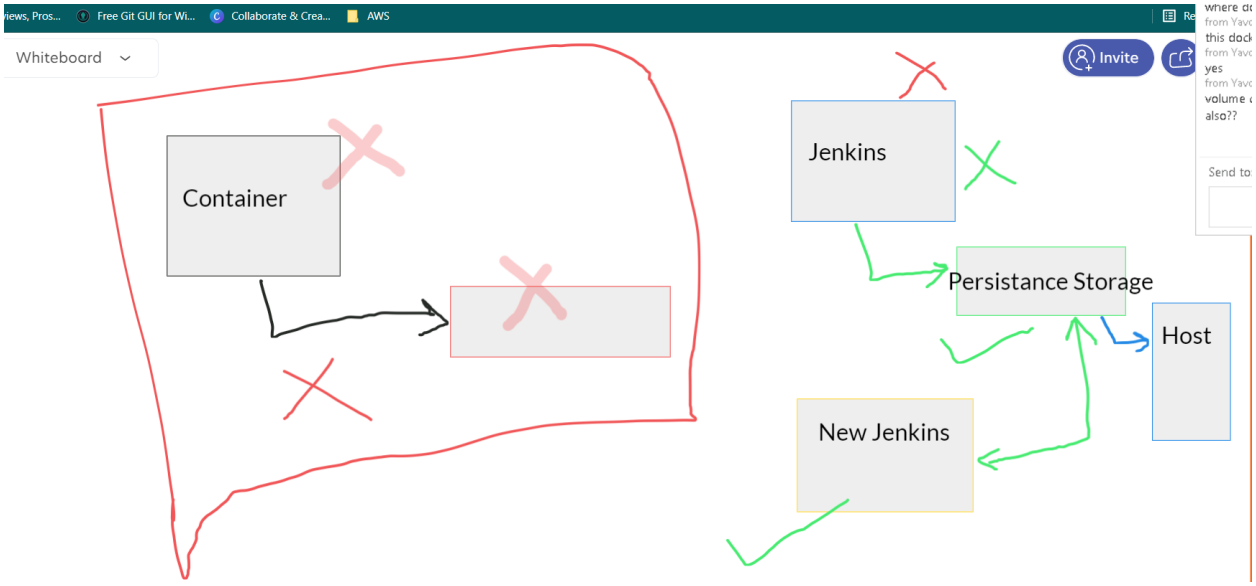
```
docker logs 3c2d26b27643

docker inspect 6c64ff003a1f
```

```
docker inspect --format='{{.Config.Image}}' 6c64ff003a1f

docker inspect --format='{{.NetworkSettings}}' 6c64ff003a1f

docker inspect --format='{{.NetworkSettings.Bridge}}' d8c78b5b79e5
```

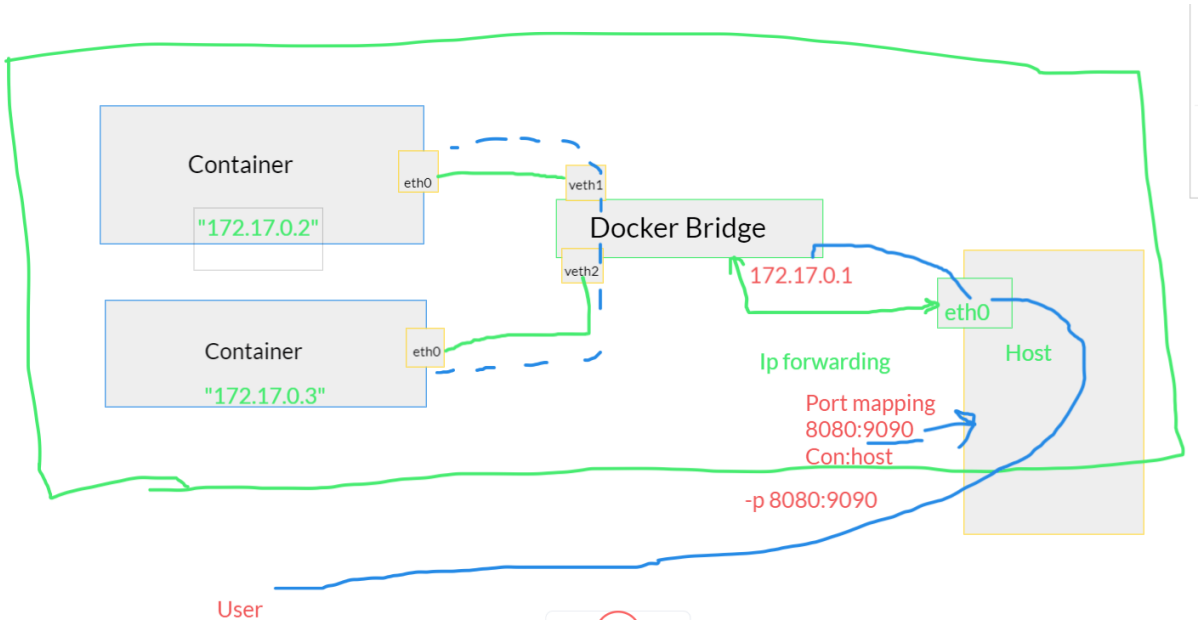


Networking

```
root@ip-172-31-79-36:~# docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
f560eadc93b0        bridge              bridge              local
d4d73178f33d        host                host                local
074228cd7ca0        none                null                local
root@ip-172-31-79-36:~#
```

```
root@ip-172-31-79-36:~# ip r l
default via 172.31.64.1 dev ens5
169.254.0.0/16 dev ens5  scope link  metric 1000
172.17.0.0/16 dev docker0  proto kernel  scope link  src 172.17.0.1
172.31.64.0/20 dev ens5  proto kernel  scope link  src 172.31.79.36
root@ip-172-31-79-36:~#
```

```
root@ip-172-31-79-36:~# #172.31.79.36:9090 ------>  172.17.0.1:8080 ---> 172.17.0.3:8080
root@ip-172-31-79-36:~# #172.31.79.36:9080 ------>  172.17.0.1:8080 ---> 172.17.0.2:8080
```




```
docker exec -it 56fc637bc9df /bin/bash

yum install net-tools===>to enable ip address functionality(ifconfig)
```

https://hub.docker.com/r/sonatype/nexus3/

docker exec -it 6c64ff003a1f /bin/bash

Nexus-uploader - Jenkins plugins

docker rm -f 76d4e75a8712

```
docker volume create --name nexus-data
```

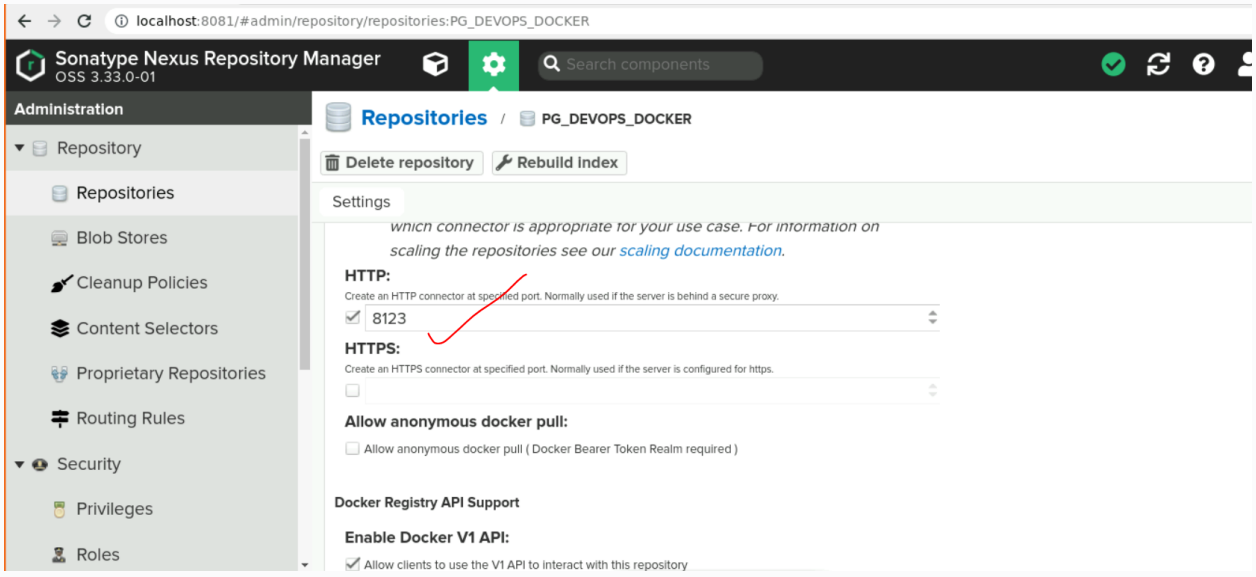
```
docker run -d -p 8081:8081 -p 8123:8123 --name nexus -v nexus-data:/nexus-data
sonatype/nexus3
```

docker run -d -p 8081:8081 -p 8123:8123 --name nexus -v nexus-data:/nexus-data sonatype/nexus3

```
var/lib/docker/volumes/nexus-data/_data/admin.password
```

Just login in Nexus and create hosted docker repo and assigned the required port, here we have used 8123 as private registry port for docker.

Repo type - docker (hosted)



We have added one extra port 8123 for docker private registry

Cat /etc/*release==>to know the ubuntu version

Vi /etc/default/docker==>The below added scripts doesn't work for ubuntu version 16

```
# This is also a handy place to tweak where Docker's temporary files go.
#export DOCKER_TMPDIR="/mnt/bigdrive/docker-tmp"
DOCKER_OPTS=$DOCKER_OPTS --insecure-registry=172.31.79.36:8123
~
~
~
~
~
```

```
root@ip-172-31-79-36:~# docker login -u admin -p ajay 172.31.79.36:8123
WARNING! Using --password via the CLI is insecure. Use --password-stdin.
Error response from daemon: Get https://172.31.79.36:8123/v2/: http: server gave HTTP response to HTTPS client
root@ip-172-31-79-36:~#
```

```
root@ip-172-31-79-36:~# cat /etc/docker/daemon.json
{
    "insecure-registries" : ["172.31.79.36:8123"]

}
root@ip-172-31-79-36:~#
```

```
systemctl restart docker
cd /etc/docker/

cat /etc/docker/daemon.json
{
   "insecure-registries" : ["172.31.79.36:8123"]

}

Netstat -tnlup | grep 8123


docker login -u admin -p ajay 172.31.79.36:8123

docker tag  ajaycs/dockerfile_ansible:v2 172.31.79.36:8123/dockerfile_ansible:v2
docker tag  [SOURCE IMAGE NAME] [TARGET IMAGE NAME]


docker tag ganeshkale/ansible:1.0  localhost:8123/repository/ganeshlocal/ansible:1.0

docker push localhost:8123/repository/ganeshlocal/ansible:1.0



docker start 6c64ff003a1f
docker stop 6c64ff003a1f

docker rm 6c64ff003a1f
docker rm -f 6c64ff003a1f

docker rmi [image id or image name]


ansible-galaxy init DEVOPS_ROLE==>

Yum install tree
```

```
[root@6c64ff003a1f pgdevops]# tree DEVOPS_ROLE/
DEVOPS_ROLE/
|-- README.md
|-- defaults
|   `-- main.yml
|-- files
|-- handlers
|   `-- main.yml
|-- meta
|   `-- main.yml
|-- tasks
|   `-- main.yml
|-- templates
|-- tests
|   |-- inventory
|   `-- test.yml
`-- vars
    `-- main.yml

8 directories, 8 files
```

ansible-playbook -i PG_DEVOPS deploy_web.yml --ask-vault-pass

https://galaxy.ansible.com/

```
[root@6c64ff003a1f pgdevops]# cat  deploy_web.yml
---
- hosts: frontend
  become: yes
  become_user: root
  gather_facts: no
  roles:
    - DEVOPS_ROLE
    - geerlingguy.java
[root@6c64ff003a1f pgdevops]#
```

From Ganesh Kale
@Ajay
Remaining part from Course 2
1:-Docker compose
2:-Terraform

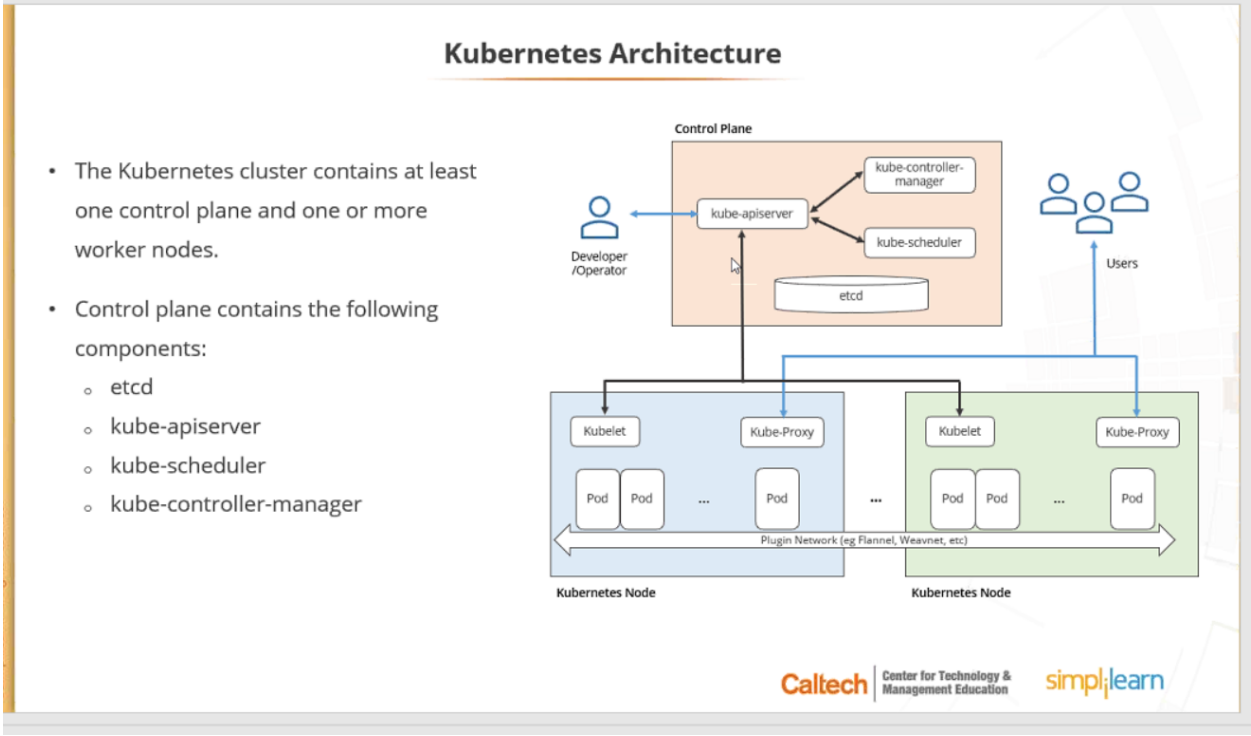
3:-Nagios
4:-Kubernetes

Are we able to do above 4 things in only one session ?
please check live class course syllabus .
Please add one more session on 15 august 2021 for same.

Team what you think ?

@Ajay
Ansible Question :
How to search string in variable having colon in it ?

@Ajay
Ansible Galaxy Question:
Please show us demo of ansible galaxy module creation and pushing to ansible-galaxy website (i am assuming it maybe similar as docker.hub working)



https://kubernetes.io/docs/setup/production-environment/tools/kubeadm/install-kubeadm/

/var/log/messages or /var/log/syslog here to check if kubelet is not working