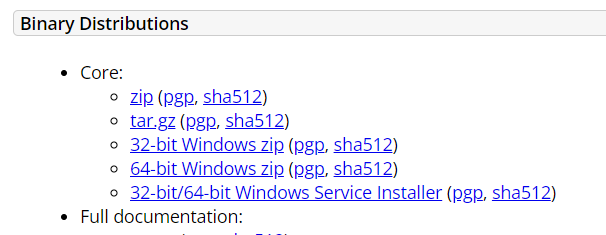
1. Download Tomcat

<https://tomcat.apache.org/download-90.cgi>



1. Conf/tomcat-users.xml

<role rolename="manager-gui"/>

<role rolename="manager-script"/>

<role rolename="manager-jmx"/>

<role rolename="manager-status"/>

<user username="admin" password="admin" roles="manager-gui, manager-script, manager-jmx, manager-status"/>

<user username="deployer" password="deployer" roles="manager-script"/>

<user username="tomcat" password="s3cret" roles="manager-gui"/>

1. Integrate tomcat with jenkins
2. Install “Deploy to container” plugin
3. Configure tomcat server with credentials

How to change the password of tomcat go to server.xml inside conf folder

<Connector port="8080" protocol="HTTP/1.1"

connectionTimeout="20000"

redirectPort="8443"

maxParameterCount="1000"

Docker pull tomcat

Docker run -d –name tomcat-container -p 8082:8080

Mobin@LAPTOP-5HS2QF1F MINGW64 ~

$ docker run -d --name tomcat-container -p 8082:8080 tomcat

b83e7d60803855d0942cbdd70d329c03d3a3b5df43e5580122326eeed0536eff

Mobin@LAPTOP-5HS2QF1F MINGW64 ~

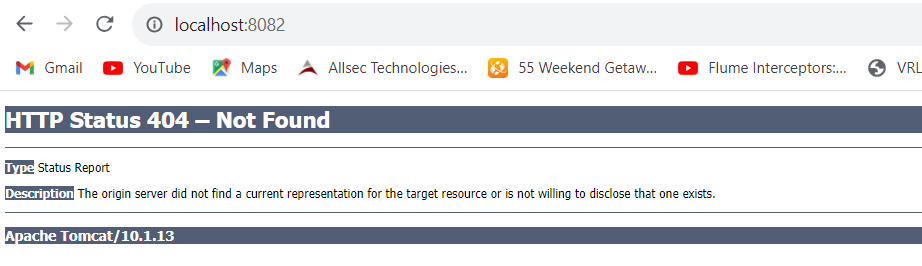
$ docker ps -a

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

b83e7d608038 tomcat "catalina.sh run" 17 seconds ago Up 16 seconds 0.0.0.0:8082->8080/tcp tomcat-container

Go to browser and type:

Localhost:8082



From Git bash it will give below error

Mobin@LAPTOP-5HS2QF1F MINGW64 ~

$ docker exec -it tomcat-container /bin/bash

the input device is not a TTY. If you are using mintty, try prefixing the command with 'winpty'

Go on cmd

To solve this issue you need to copy files from webapps.dist to webapps on :/usr/local/tomcat

C:\Users\Mobin>docker exec -it tomcat-container /bin/bash

root@b83e7d608038:/usr/local/tomcat# ls

bin conf lib logs NOTICE RELEASE-NOTES temp webapps.dist

BUILDING.txt CONTRIBUTING.md LICENSE native-jni-lib README.md RUNNING.txt webapps work

root@b83e7d608038:/usr/local/tomcat# cd webapps

root@b83e7d608038:/usr/local/tomcat/webapps# ls

root@b83e7d608038:/usr/local/tomcat/webapps# cd ..

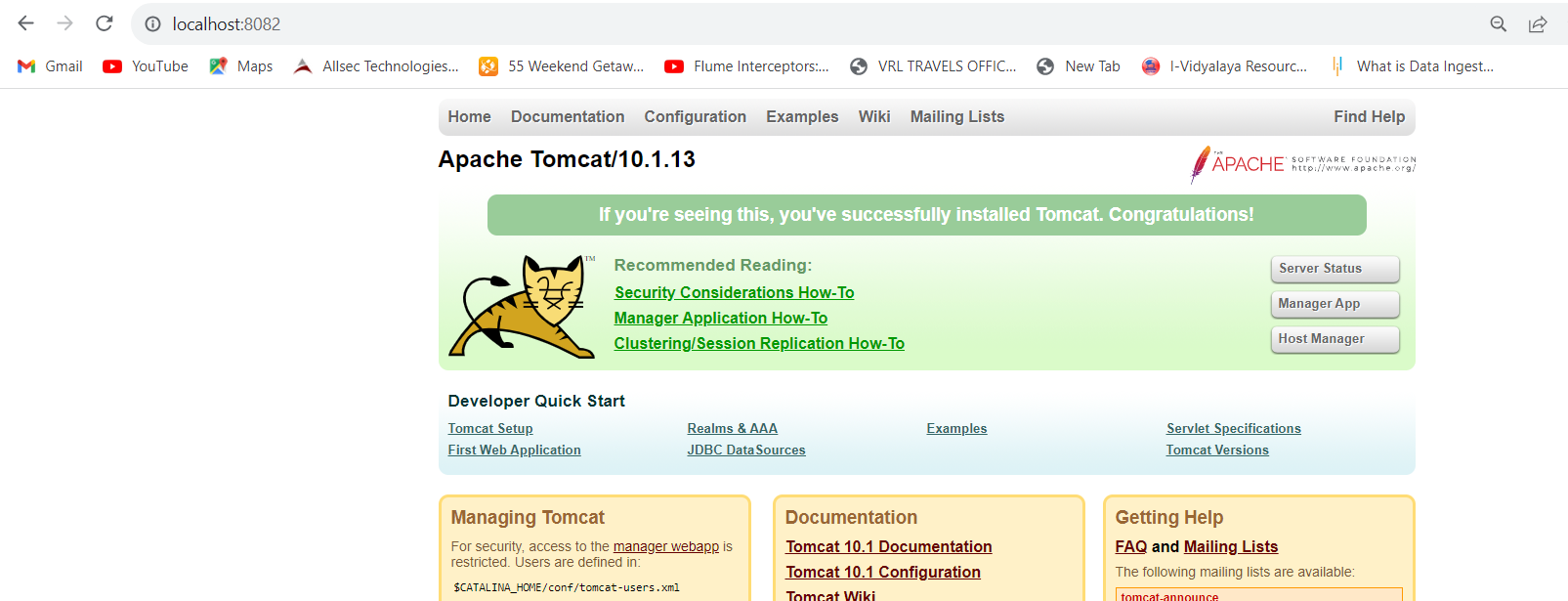
root@b83e7d608038:/usr/local/tomcat# cd webapps.dist/

root@b83e7d608038:/usr/local/tomcat/webapps.dist# ls

docs examples host-manager manager ROOT

root@b83e7d608038:/usr/local/tomcat/webapps.dist# cp -R ./\* ./../webapps/

[root@b83e7d608038:/usr/local/tomcat/webapps.dist#](mailto:root@b83e7d608038:/usr/local/tomcat/webapps.dist)



Now if we create new container we will again have to copy the files webapps to resolve that issue

See below I am killing the docker container and creating new container from same imgae:

C:\Users\Mobin>docker ps

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

b83e7d608038 tomcat "catalina.sh run" 14 minutes ago Up 14 minutes 0.0.0.0:8082->8080/tcp tomcat-container

c7fa57993179 mariadb "docker-entrypoint.s…" 7 weeks ago Up 24 minutes (unhealthy) 3306/tcp microservice-mobeenapp-db-1

C:\Users\Mobin>docker stop b83e7d608038

b83e7d608038

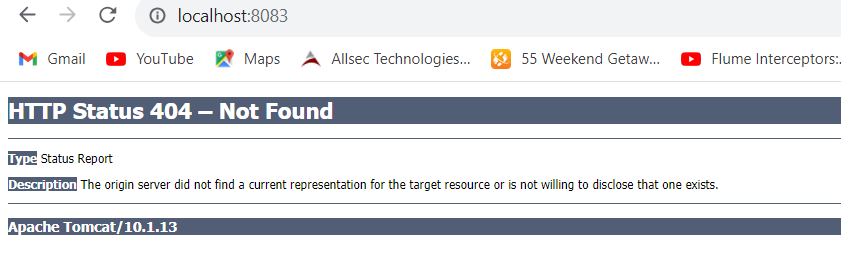
C:\Users\Mobin>docker ps

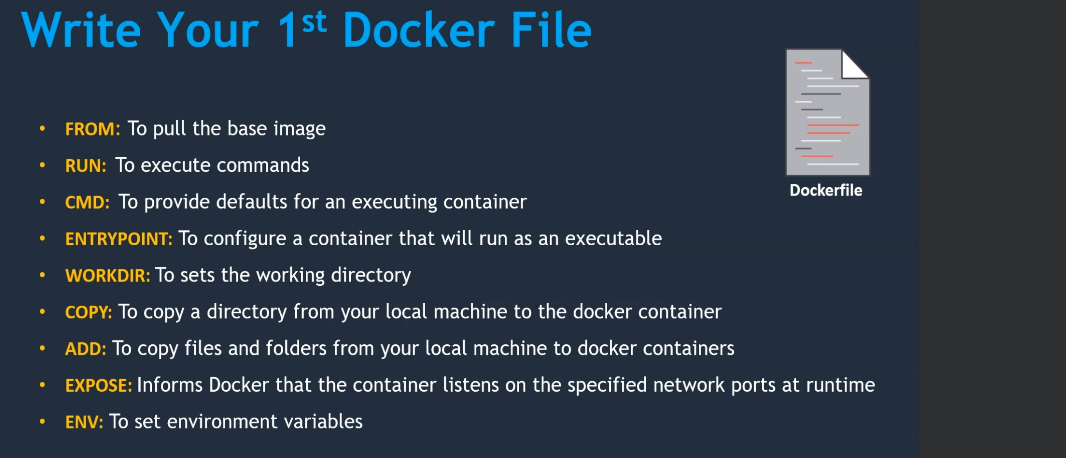
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

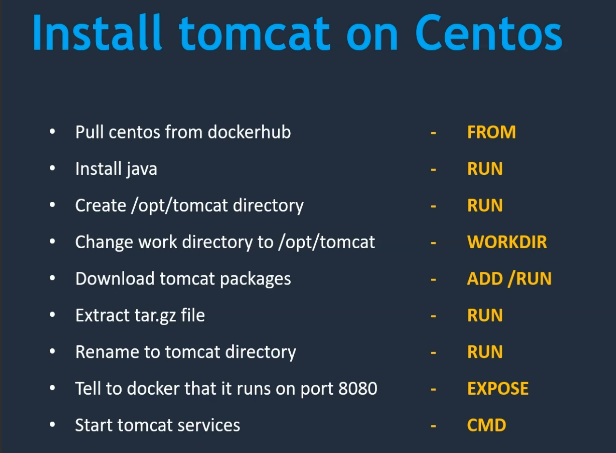
c7fa57993179 mariadb "docker-entrypoint.s…" 7 weeks ago Up 24 minutes (unhealthy) 3306/tcp microservice-mobeenapp-db-1

C:\Users\Mobin>

You can see that new docker container is again showing same error so we need to resolve this issue using docker file creation where we will explicitly specify to copy webapps







FROM centos:centos7.9.2009

RUN yum -y update

RUN yum install java -y

RUN mkdir /opt/tomcat/

WORKDIR /opt/tomcat

ADD https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.80/bin/apache-tomcat-9.0.80.tar.gz /opt/tomcat

RUN tar xvfz apache\*.tar.gz

RUN mv apache-tomcat-9.0.80/\* /opt/tomcat

EXPOSE 8080

CMD ["/opt/tomcat/bin/catalina.sh", "run"]

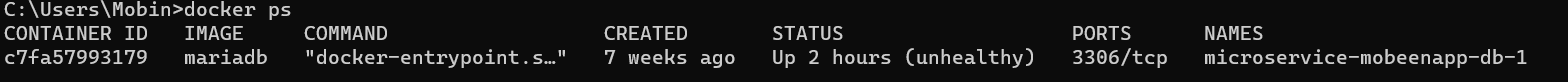


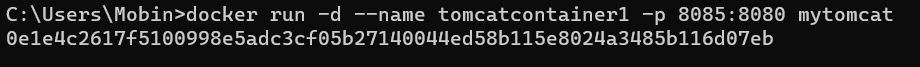


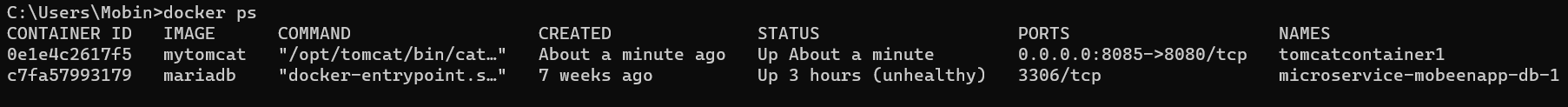
You will get below error if you save dockerfile as txt file

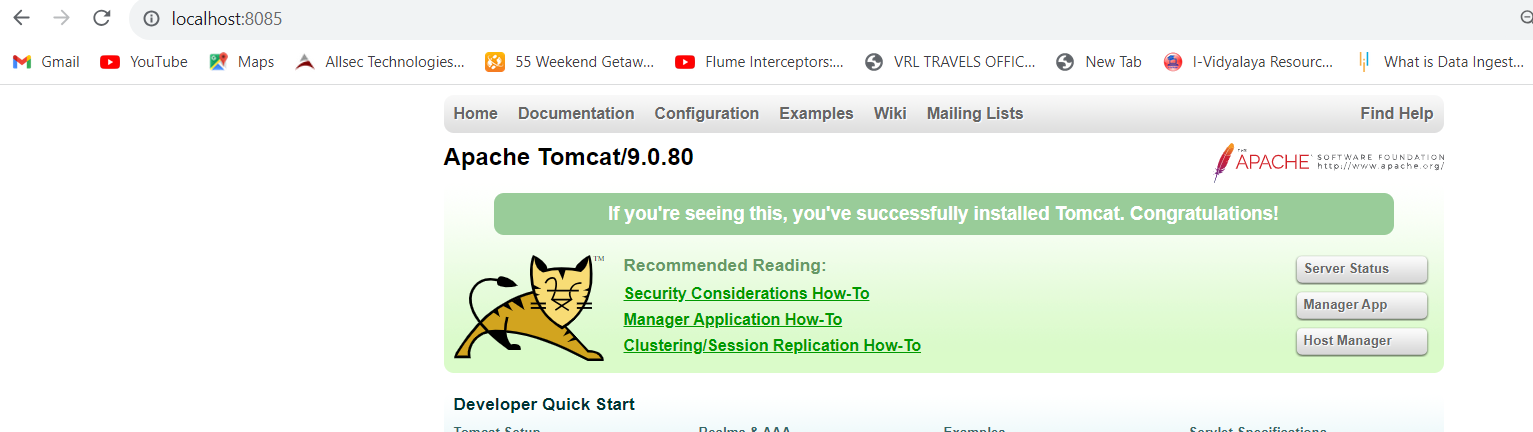
Take the tomcat link from below

https://tomcat.apache.org/download-90.cgi









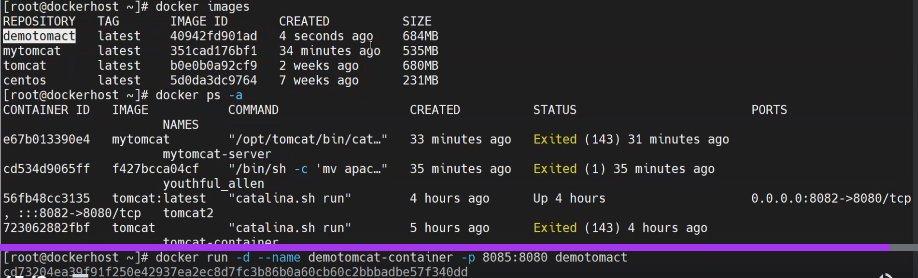
C:\Users\Mobin>docker exec -it 0e1e4c2617f5 /bin/bash

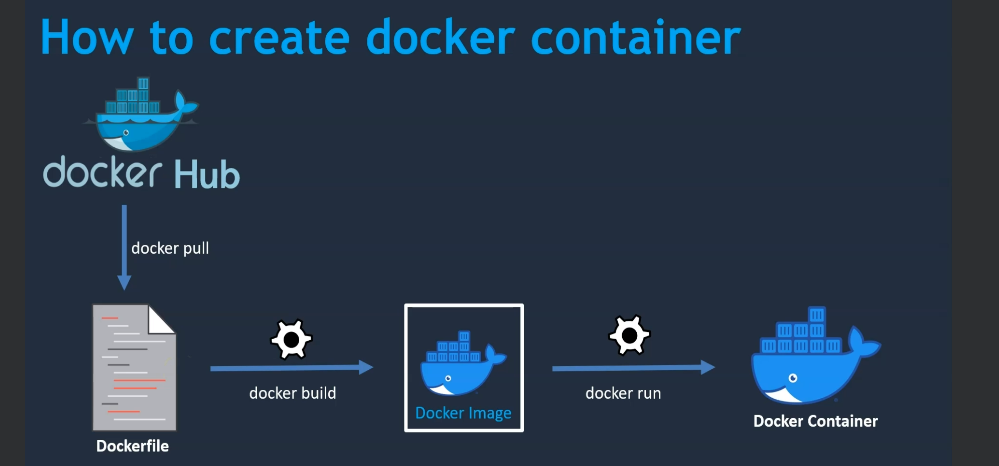
[root@0e1e4c2617f5 tomcat]#

Now one more way to create docker

Vi Dockerfile





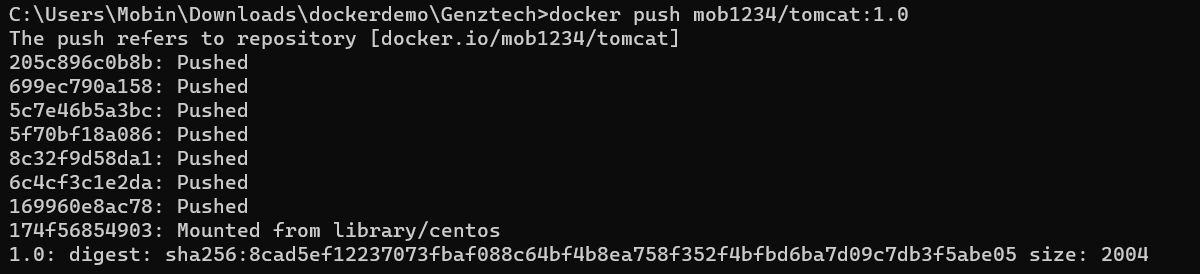


Pushing docker image on hub

docker tag <local\_image\_name> <your\_username>/<repository\_name>:<version>

C:\Users\Mobin\Downloads\dockerdemo\Genztech>docker tag mytomcatdemoclass mob1234/tomcat:1.0





## [Create and manage volumes](https://docs.docker.com/storage/volumes/#create-and-manage-volumes)

**Create a volume**:

$ docker volume create my-vol

**List volumes**:

$ docker volume ls

local my-vol

**Inspect a volume**:

$ docker volume inspect my-vol

[

{

"Driver": "local",

"Labels": {},

"Mountpoint": "/var/lib/docker/volumes/my-vol/\_data",

"Name": "my-vol",

"Options": {},

"Scope": "local"

}

]

**Remove a volume**:

$ docker volume rm my-vol

## [Start a container with a volume](https://docs.docker.com/storage/volumes/" \l "start-a-container-with-a-volume)

docker run -d \

--name devtest \

--mount source=myvol2,target=/app \

nginx:latest

Docker Volumes:

docker volume create mydata

docker volume ls

docker run -d –name tomcat1:v1.0 –mount source=mydata,target=/app tomcat1:v1.0

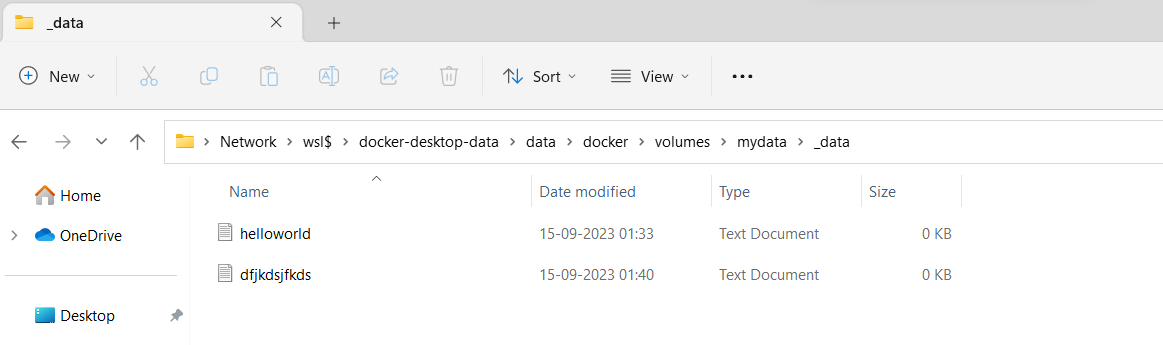
On Windows you can find your mount

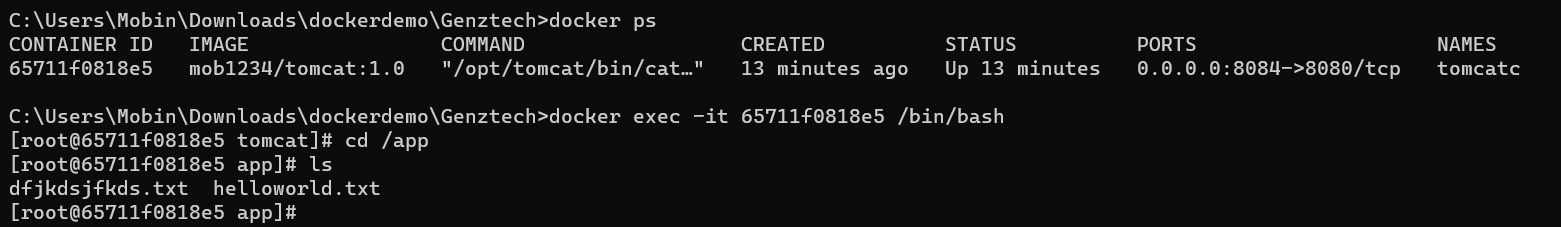
Type in the Windows file explorer :

* For Docker version 20.10.+ : \\wsl$\docker-desktop-data\data\docker\volumes
* For Docker Engine v19.03: \\wsl$\docker-desktop-data\version-pack-data\community\docker\volumes\

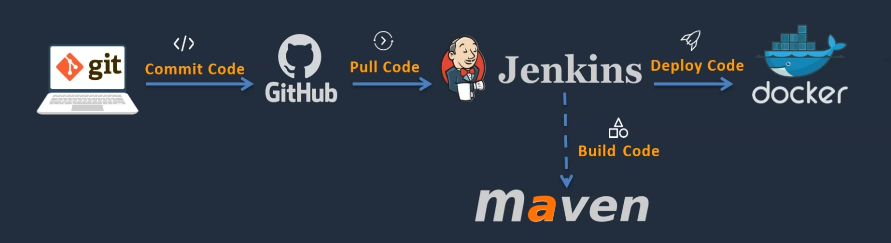
You will have one directory per volume.

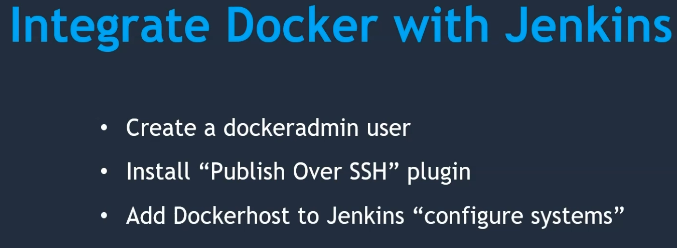






Deploying code on Docker container using Jenkins:

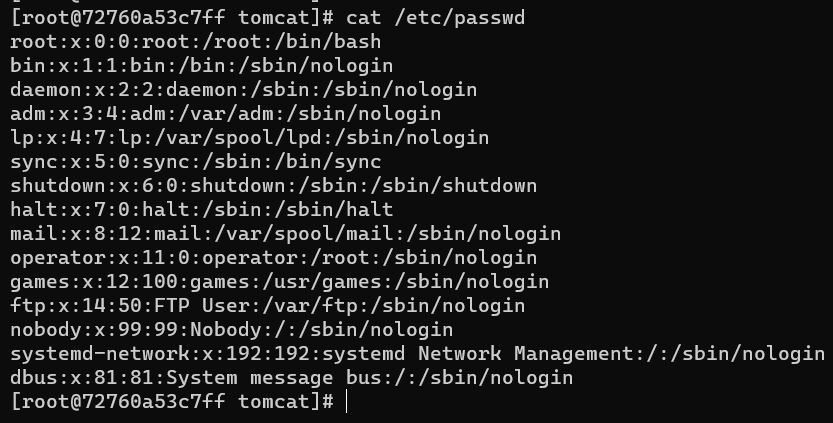




**Steps to Create group docker and user dockeradmin**

Check what all users are available on docker host by running command

Cat /etc/passwd



In similar way we can check what groups we have on this docker host



1. **Create group docker**

[root@72760a53c7ff tomcat]# groupadd docker

Check whether Group got created or not you should see similar line as docker:x:1000  
Cat /etc/groups

1. **Create User dockeradmin**

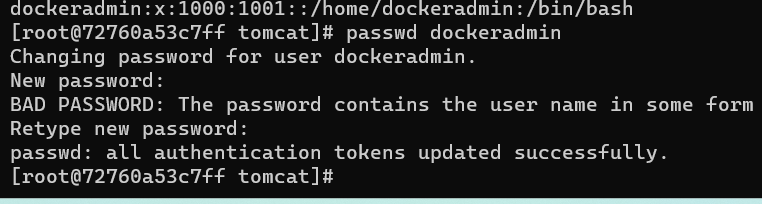
[root@72760a53c7ff tomcat]# sudo useradd -m -G docker dockeradmin

Check whether User got created or not you should see similar line as dockeradmin:x:1000  
Cat /etc/passwd

dockeradmin:x:1000:1001::/home/dockeradmin:/bin/bash

1. **Set password for dockeradmin**

passwd dockeradmin



Check whether dockeradmin got created and docker group is set for it.

[dockeradmin@localhost opt]$ id dockeradmin

uid=1001(dockeradmin) gid=1002(dockeradmin) groups=1002(dockeradmin),1001(docker)

**Now try to login on dockerhost using dockeradmin user**

[root@localhost opt]# su dockeradmin

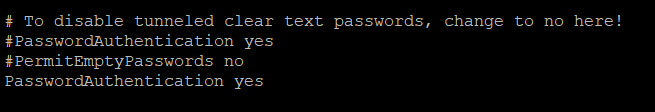
[dockeradmin@localhost opt]$

Open Putty and type ip address and try to login

If you face any password issue, that means you are not getting to login then you have to edit file /etc/ssh/sshd\_config

Vi /etc/ssh/sshd\_config

Uncomment line password authentication yes

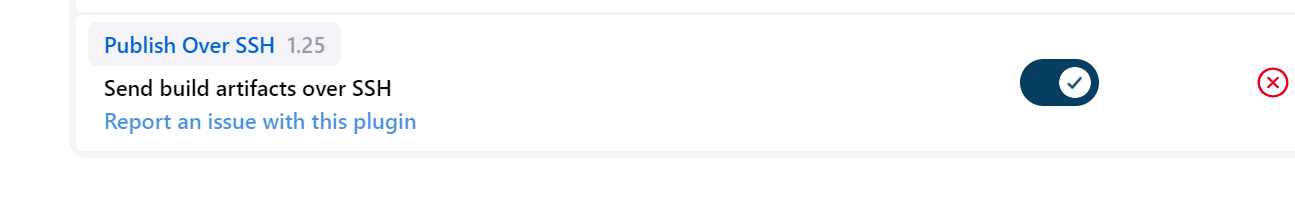


**Restart the sshd service**

Service sshd reload

1. **Integrate Jenkins with dockerhost**

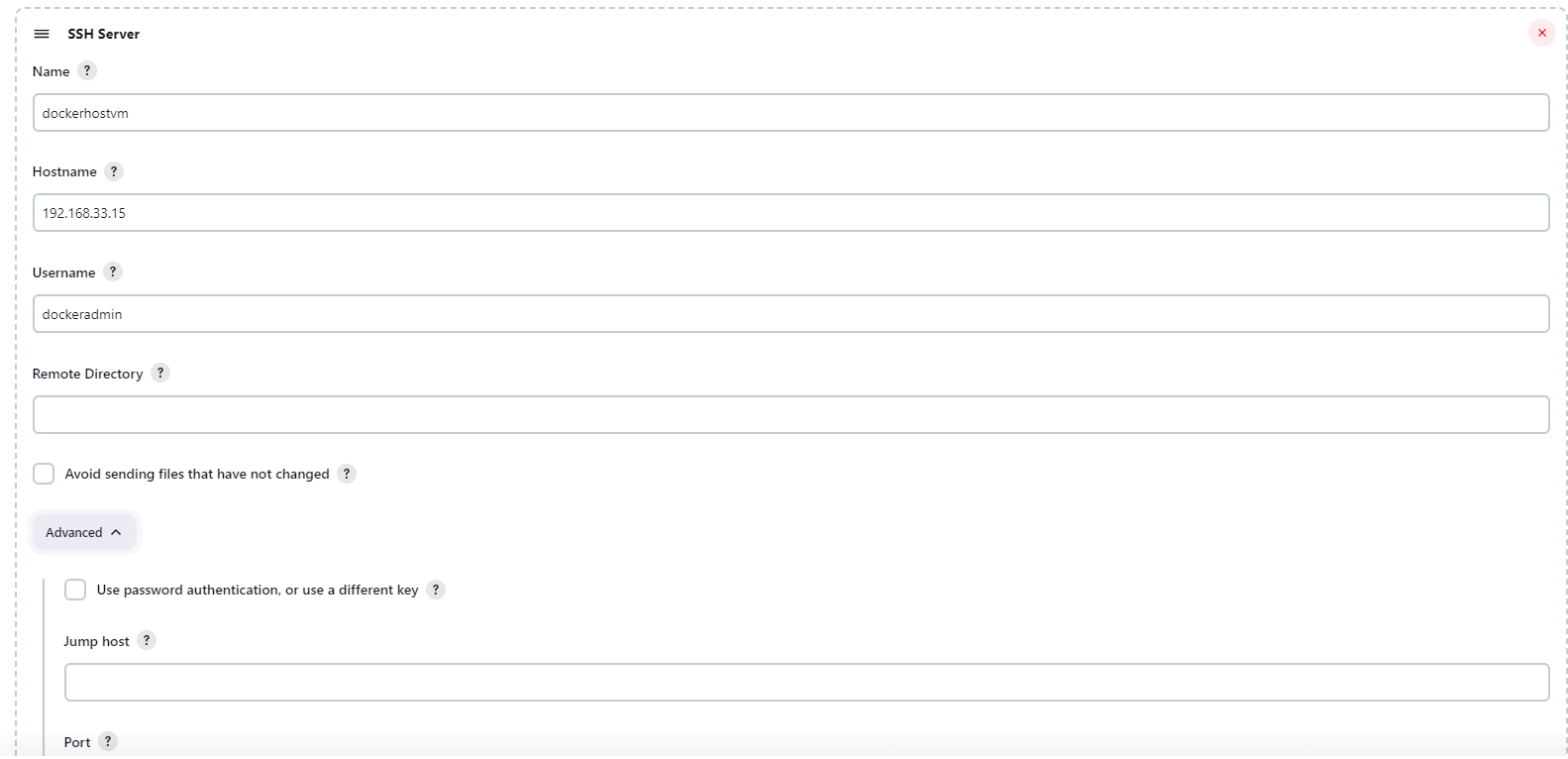
Login to jenkins, install plugin below



1. **Now add the docker host in jenkins which got newly created:**

Go to manage jenkins, system configuration, go till end, add ssh server details as below

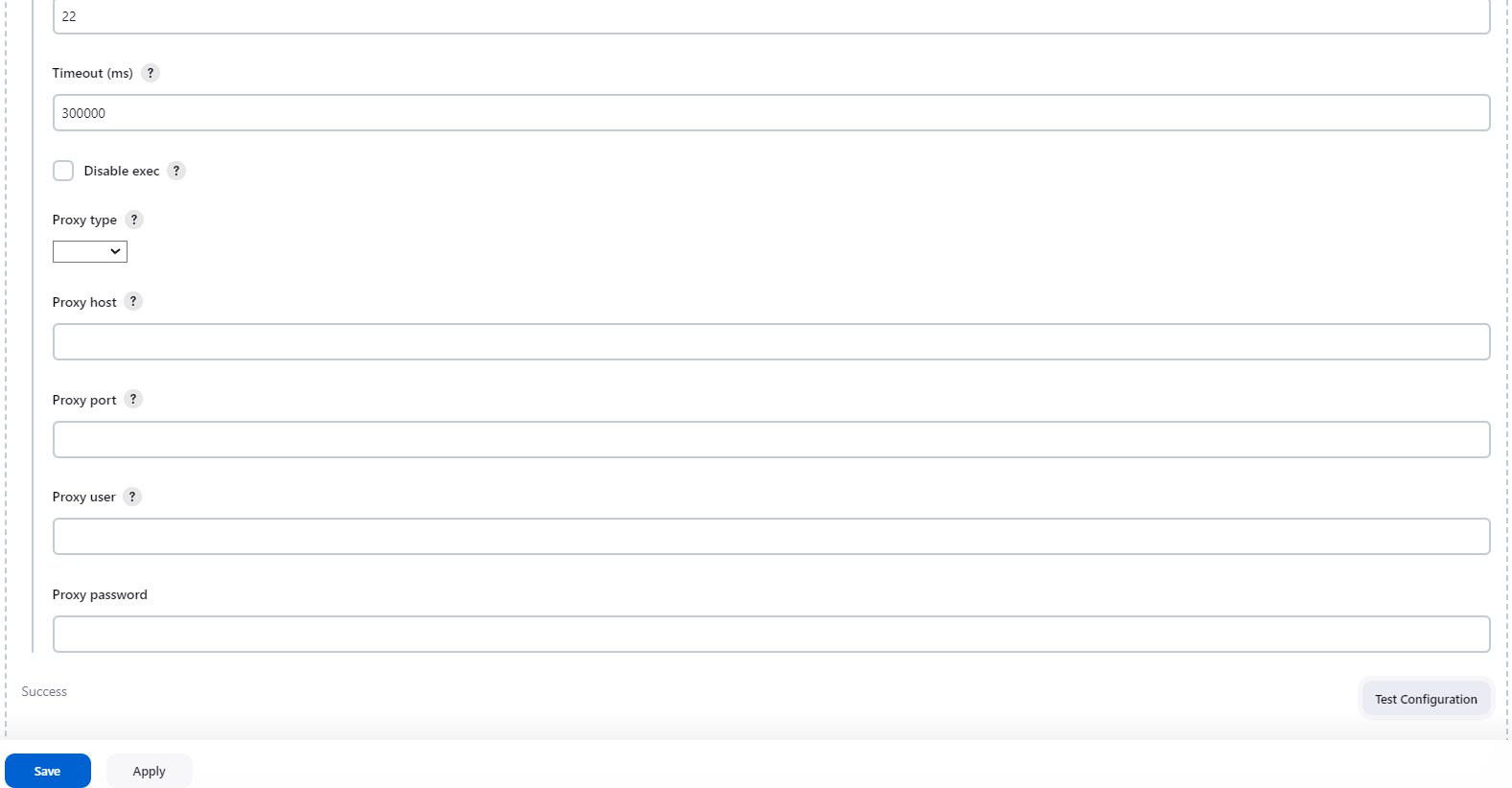
Add, password and test configuration to get success.













EXTRA:

<https://www.redswitches.com/blog/install-docker-in-centos/>

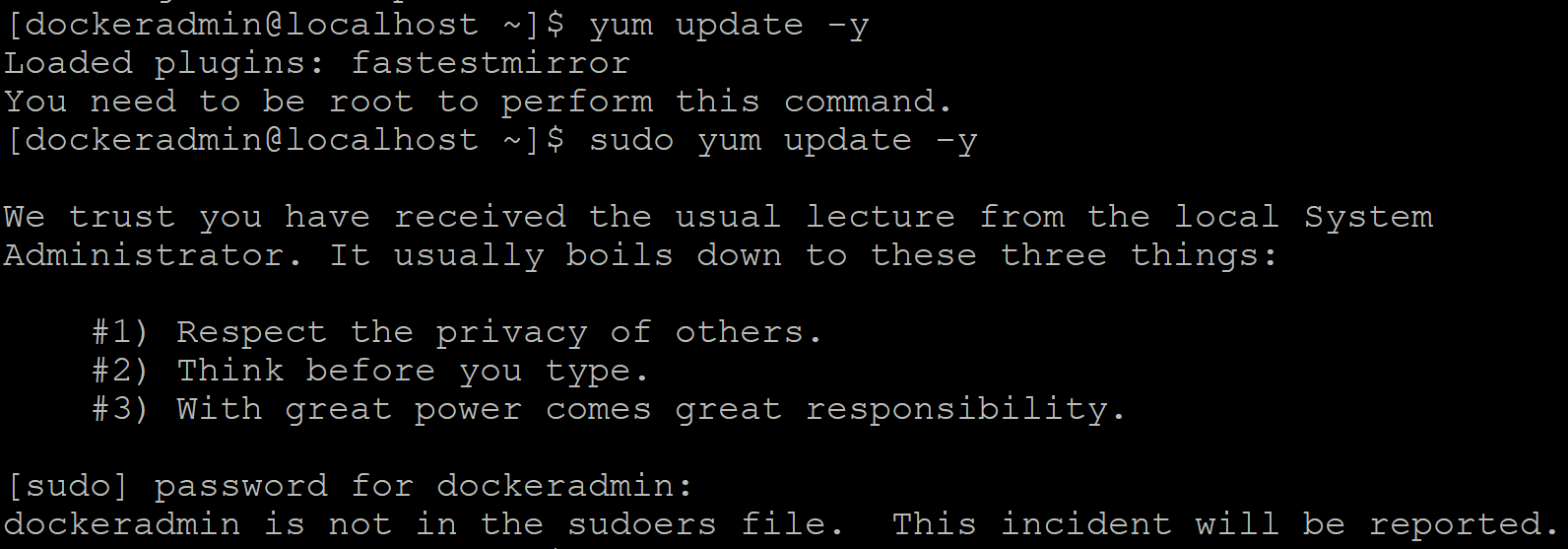
**Install docker on dockerhost**

**Go on newly created server dockerhost using dockeradmin and run below commands**

**Step 1: Update all packages on the server**

Sudo yum update -y

If you get below error then you have to login into dockerhost using root user and need to update the sudoer file as below:

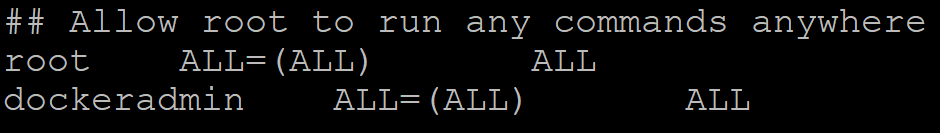


vi /etc/sudoers



Add line below root ALL=(ALL) ALL

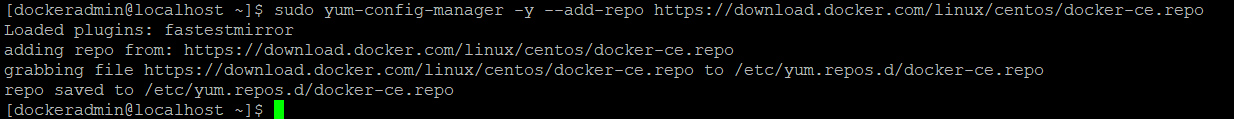
dockeradmin ALL=(ALL) ALL



**Step 2: Install The Docker Repository**

When installing, you first need to add the official Docker repository to your CentOS machine. This will ensure that we have the latest version. Here’s how to install Docker CentOS 7 repository:

# sudo yum-config-manager -y --add-repo https://download.docker.com/linux/centos/docker-ce.repo



Next, install the Docker Community Edition on the machine. For this, use the following command.

#sudo yum install docker-ce -y

Once installed you can verify docker is working or not so run any docker command:



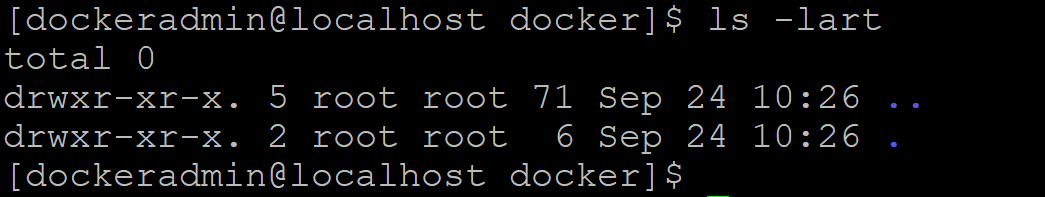
**Step 4: Start the Docker Service**

At this point, you’ve installed Docker and added a non-root user to the Docker group. Next, it is time to start the Docker daemon. For this, use the following command:

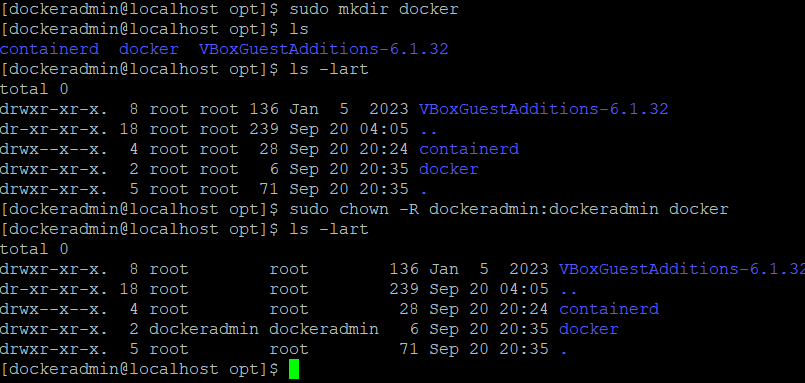
# sudo systemctl start docker

We want to copy everything inside /opt/docker folder on dockerhost so we need to create the docker folder.

Now go on host and create docker folder, see who is user and group if it is root then we have to change it to docker and dockeradmin



sudo chown -R dockeradmin:dockeradmin docker



Go inside docker folder

Cd docker

Create Dockerfile here and copy the content of Dockerfile file github

<https://github.com/mobeenh/project/blob/main/Dockerfile>

create it using vi editor

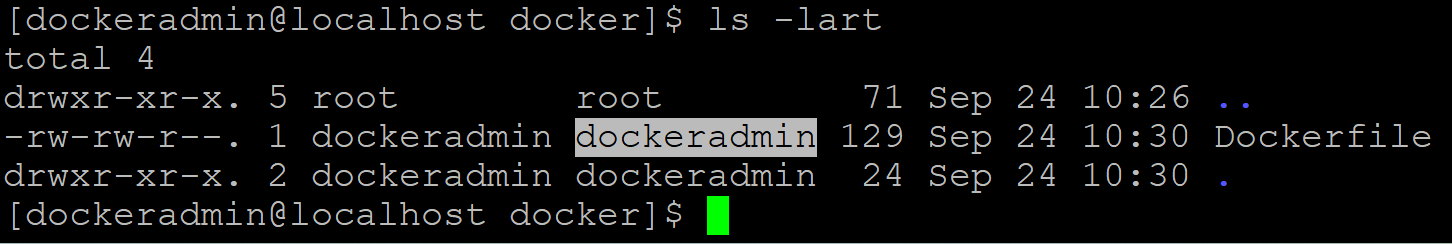
vi Dockerfile

FROM tomcat:latest

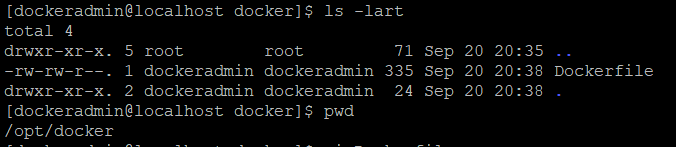
RUN cp -R /usr/local/tomcat/webapps.dist/\* /usr/local/tomcat/webapps

COPY ./\*.war /usr/local/tomcat/webapps

Check the owner of this Dockerfile it should be dockeradmin if not then run the command



sudo chown -R dockeradmin:dockeradmin Dockerfile

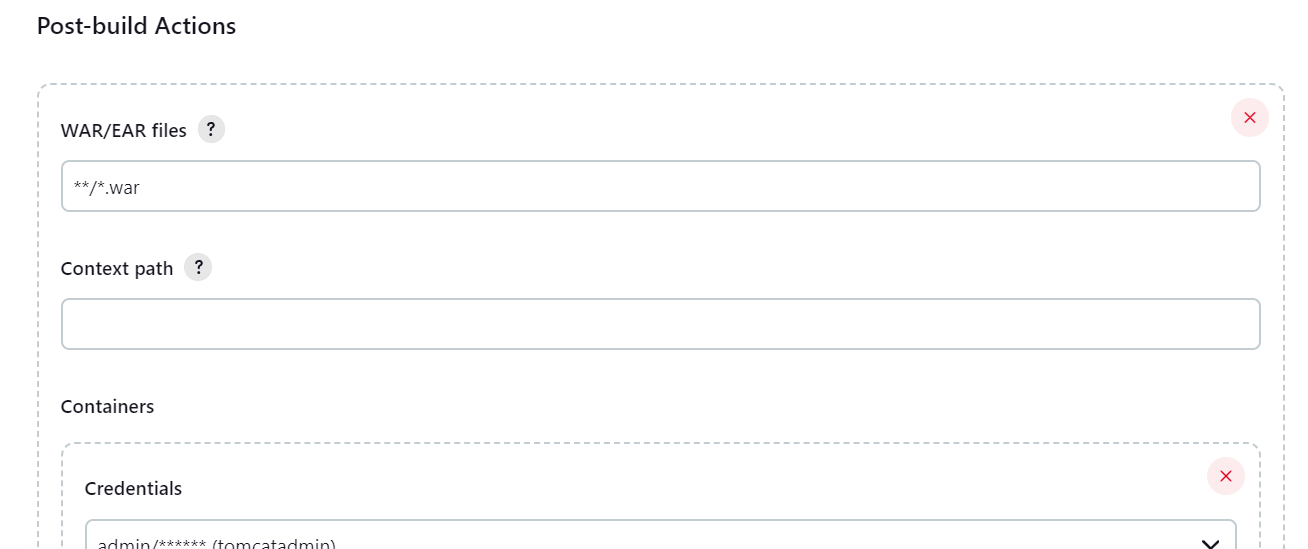


**NOW SETUP IS DONE, WE ARE GOING TO DEPLOY ARTIFACT WAR USING JENKINS ON DOCKERHOST**

**Go to jenkins and create new job by copying existing one**

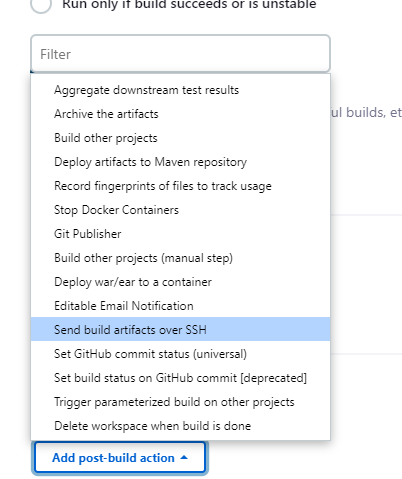
Copy the existing jenkins job and create new jenkins job from maven project(where we have created war file).

Remove the below yellow marked





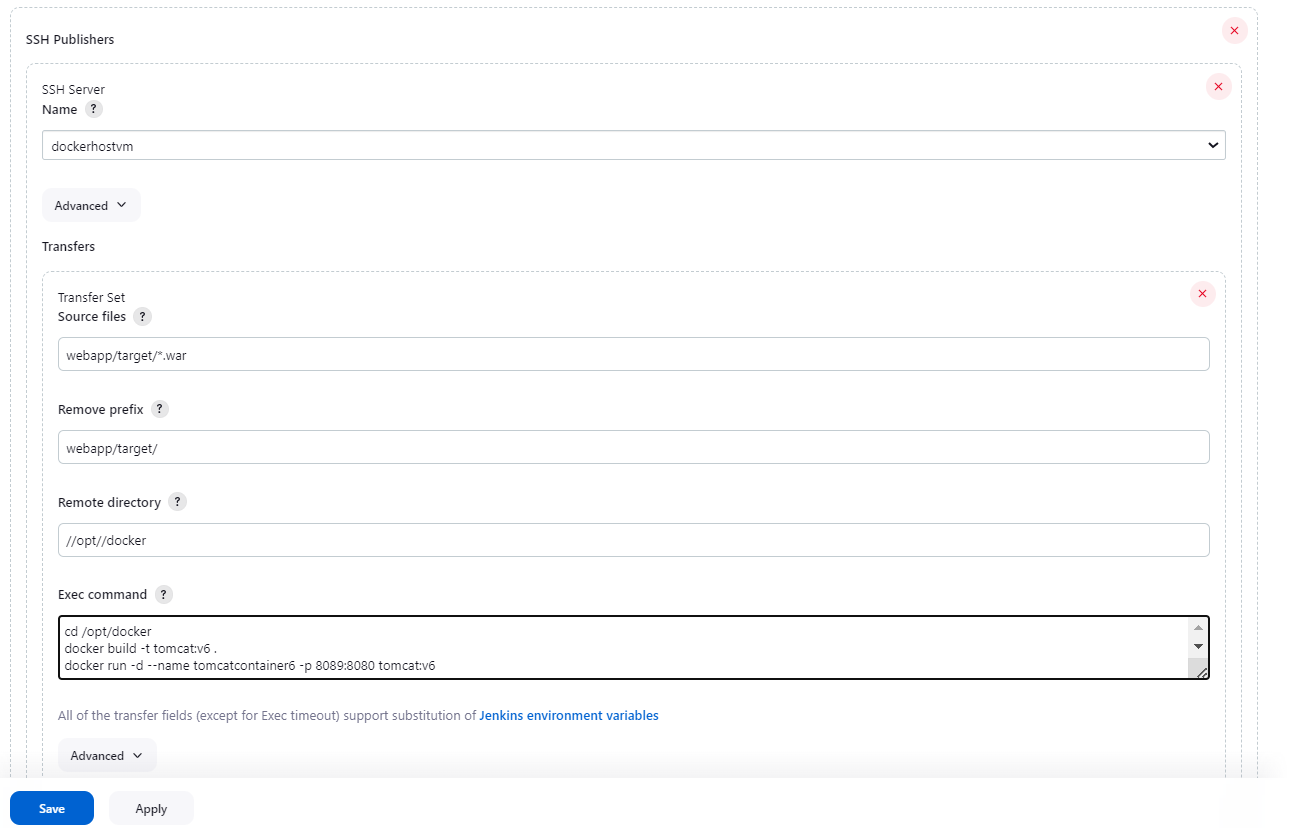
Go to post build steps and add below



Cd /opt/docker

docker build -t tomcat:v5 .

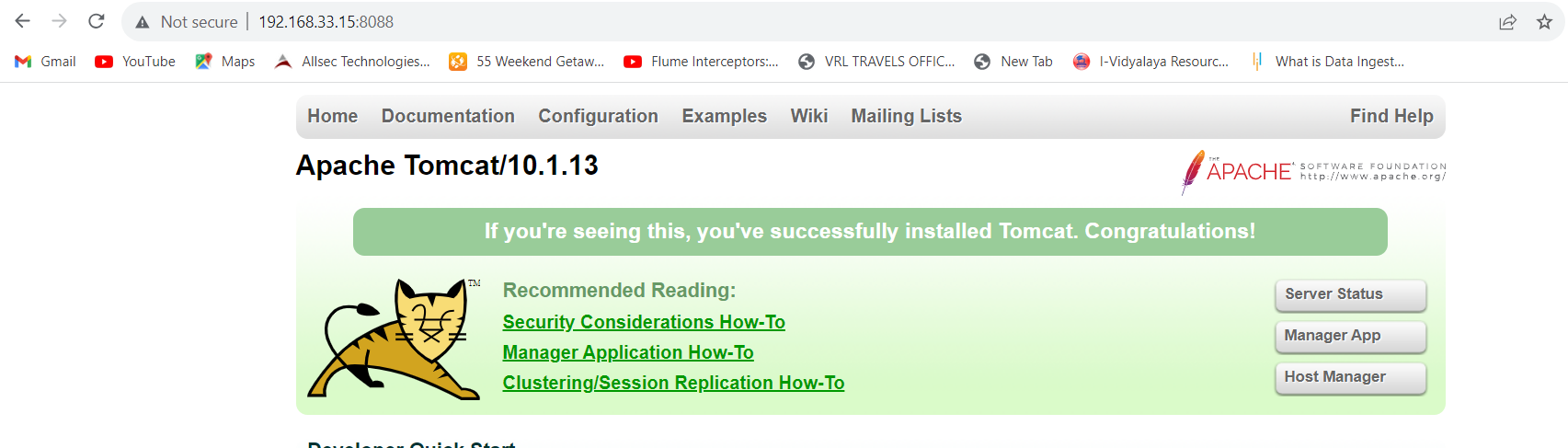
docker run -d --name tomcatcontainer -p 8088:8080 tomcat:v5

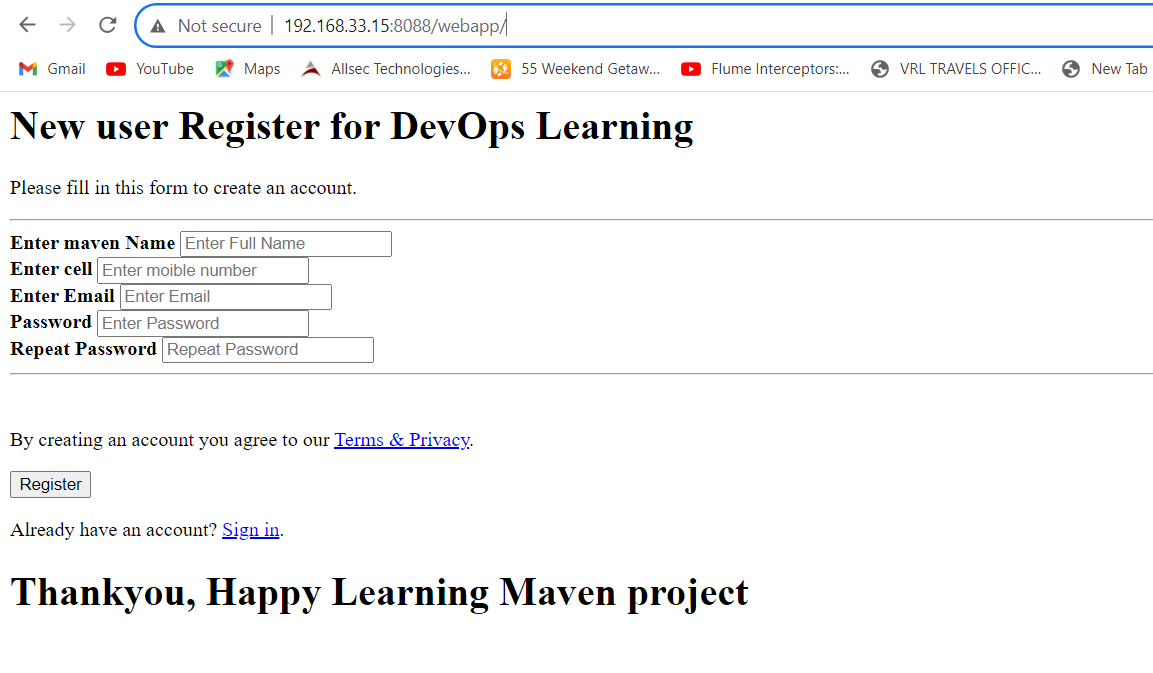


Go on browser and test

<http://192.168.33.15:8088/>

<http://192.168.33.15:8088/webapp>





[dockeradmin@localhost opt]$ ls

VBoxGuestAdditions-6.1.32

[dockeradmin@localhost opt]$ mkdir tomcat

mkdir: cannot create directory ‘test’: Permission denied

[dockeradmin@localhost opt]$ sudo mkdir tomcat

We trust you have received the usual lecture from the local System

Administrator. It usually boils down to these three things:

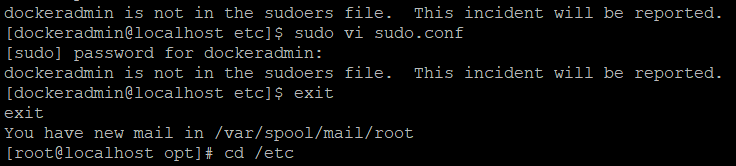
#1) Respect the privacy of others.

#2) Think before you type.

#3) With great power comes great responsibility.

[sudo] password for dockeradmin:

dockeradmin is not in the sudoers file. This incident will be reported.



Exit

[root@localhost etc]# vi /etc/sudoers

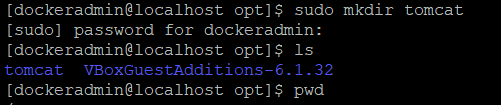
Add dockeradmin ALL=(ALL) ALL

line after root ALL=(ALL) ALL

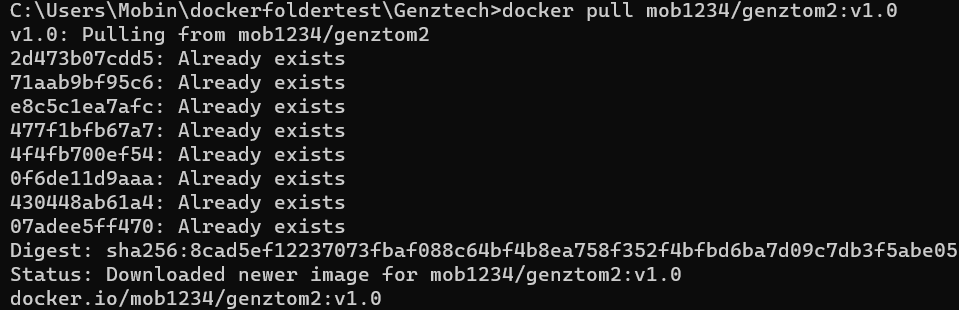
## Allow root to run any commands anywhere

root ALL=(ALL) ALL

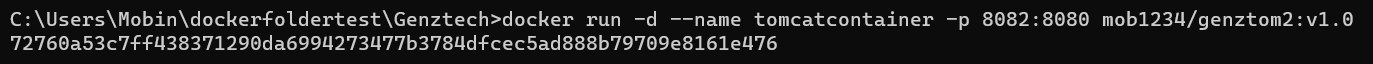
dockeradmin ALL=(ALL) ALL



docker pull mob1234/genztom2:v1:0



Run the container from docker image we just now pulled:



Go inside the container:

Docker exec -it container-id /bin/bash

