

## Create customized Dockerfile for tomcat

```
[root@docker ~]#  
[root@docker ~]# vim Dockerfile  
[root@docker ~]#  
[root@docker ~]#  
[root@docker ~]# docker build -t demotomcat .  
Sending build context to Docker daemon 15.87kB  
Step 1/2 : FROM tomcat:latest  
--> 5a4ee0c8ccbe  
Step 2/2 : RUN cp -R /usr/local/tomcat/webapps.dist/* /usr/local/tomcat/webapps  
--> Running in 61ba176b3a03  
Removing intermediate container 61ba176b3a03  
--> a60121ae860d  
Successfully built a60121ae860d  
Successfully tagged demotomcat:latest  
[root@docker ~]#  
[root@docker ~]#  
[root@docker ~]# cat Dockerfile  
FROM tomcat:latest  
RUN cp -R /usr/local/tomcat/webapps.dist/* /usr/local/tomcat/webapps  
  
[root@docker ~]#  
[root@docker ~]#
```

```
[root@docker ~]# docker images  
REPOSITORY TAG IMAGE ID CREATED SIZE  
demotomcat latest a60121ae860d 41 seconds ago 460MB  
mytomcat latest bdd25816d1b 20 minutes ago 699MB  
tomcat latest 5a4ee0c8ccbe 2 days ago 455MB  
centos 7 eeb6ee3f44bd 2 years ago 204MB  
[root@docker ~]#  
[root@docker ~]# docker run -dit --name demotomcat-container -p 8085:8080 a60121ae860d  
14ab54fae058560b461914f5470889a6841de113a59df85f11331631b587fafb  
[root@docker ~]#  
[root@docker ~]# docker ps -a  
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES  
14ab54fae058 a60121ae860d "catalina.sh run" 4 seconds ago Up 3 seconds 0.0.0.0:8085->8080/tcp, :::8085->8080/tcp demotomcat-container  
42f5c46c08d4 5a4ee0c8ccbe "/bin/sh -c 'cp -R /..." 3 minutes ago Exited (1) 3 minutes ago nice_chandrasekhar  
[root@docker ~]#  
[root@docker ~]#
```

## Integrate Docker with Jenkins

# Integrate Docker with Jenkins

- Create a dockeradmin user
- Install “Publish Over SSH” plugin
- Add Dockerhost to Jenkins “configure systems”



Jenkins



docker

```
[root@docker ~]# cat /etc/group | grep docker
docker:x:992:
[root@docker ~]#
[root@docker ~]# useradd dockeradmin
[root@docker ~]#
[root@docker ~]# passwd dockeradmin
Changing password for user dockeradmin.
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
passwd: all authentication tokens updated successfully.
[root@docker ~]#
[root@docker ~]# id dockeradmin
uid=1001(dockeradmin) gid=1001(dockeradmin) groups=1001(dockeradmin)
[root@docker ~]#
[root@docker ~]# usermod -aG docker dockeradmin
[root@docker ~]#
[root@docker ~]# id dockeradmin
uid=1001(dockeradmin) gid=1001(dockeradmin) groups=1001(dockeradmin),992(docker)
[root@docker ~]#
[root@docker ~]#
```

But I'm not able to login with dockeradmin user by default instance in disable password based authentication so we need to enable it.

```
# vim /etc/ssh/sshd_config
```

```
# To disable tunneled clear text
PasswordAuthentication yes
#PermitEmptyPasswords no
#PasswordAuthentication no
```

# service sshd restart

Install the plugin publish over ssh it will use for transfer artifacts over the server.

ADD dockerhost to Jenkins configuration.

Dashboard > Manage Jenkins > System >

☐ Test configuration by sending test e-mail

Publish over SSH

Jenkins SSH Key ?

Passphrase ?

Path to key ?

Key ?

Dashboard > Manage Jenkins > System >

SSH Servers

SSH Server

Name ?

Hostname ?

Username ?

Remote Directory ?

☐ Avoid sending files that have not changed ?

☒ Use password authentication, or use a different key ?

Passphrase / Password ?

Path to key ?

Key ?

Now, create Jenkins jobs BuildAndDeployOnContainer

Dashboard > BuildAndDeployOnContainer > Configuration

### Configure

General

Source Code Management

Build Triggers

Build Environment


Pre Steps

Build

Post Steps

Build Settings

#### General

Enabled 

Description

Build code with help of Maven and deploy it on docker

Plain text [Preview](#)

☐ Discard old builds ?

☐ GitHub project

☐ This project is parameterized ?

### Source Code Management

☐ None

☒ Git ?

#### Repositories ?

Repository URL ?

https://github.com/bunty4200/Udemy-hello-world.git

Credentials ?

- none -

+ Add

Advanced

Add Repository

#### Branches to build ?

Branch Specifier (blank for 'any') ?

\*/master

### Build

Root POM ?

pom.xml

Goals and options ?

clean install

Advanced

## Post-build Actions

Send build artifacts over SSH ?

SSH Publishers

SSH Server

Name ?

dockerhost

Advanced

Transfers

Transfer Set

Source files ?

webapp/target/webapp.war

Remove prefix ?

webapp/target

Remote directory ?

Exec command ?

Save and build job