Docker installation

<http://paxcel.net/blog/automation-of-warear-deployment-using-jenkins/> (War file deployment)

<https://www.youtube.com/watch?v=4o0WYS32x_E>

Java app(Spring boot) <https://www.youtube.com/watch?v=FlSup_eelYE>

<https://www.stratoscale.com/blog/devops/practical-devops-use-case-github-jenkins-docker/>

(<https://www.youtube.com/watch?v=1KU_IWpAiDE>

<https://www.docker.com/sites/default/files/UseCase/RA_CI%20with%20Docker_08.25.2015.pdf>

<https://forums.docker.com/t/connecting-to-docker-containers-from-other-host/14278/4>

(from the other hosts)

+++++++++

Tomcat + build deployment

<https://stackoverflow.com/questions/46052973/how-to-deploy-war-file-in-tomcat-7-of-docker-container>

+++++++++++

<https://www.youtube.com/watch?v=HUpIoF_conA>

++++

Define volume in tomcat

<https://stackoverflow.com/questions/47941168/docker-compose-run-java-web-application-war-file-on-tomcat-using-volumes>

https://www.linux.com/learn/docker-volumes-and-networks-compose

++++++++

<https://www.youtube.com/watch?v=iN3he0eVUyw> (Jenkins on docker)

docker run --name jenkins4 -p 9900:8080 -p 50000:50000 -v /home/janki/Jenkins\_home:/var/jenkins\_home Jenkins (we can take any path on local and provide chmod 777 to path)

docker run --name jenkins4 -p 9900:8080 -p 50000:50000 -v jenkinsv:/var/jenkins\_home Jenkins (using volume)

+++++++++++++++++++

To save a Docker image after you have pulled, committed or built it you use the docker save command. For example, lets save a local copy of the verse\_gapminder docker image we made:

docker save verse\_gapminder > verse\_gapminder.tar

If we want to load that Docker container from the archived tar file in the future, we can use the docker load command:

docker load --input verse\_gapminder.tar

+++

FROM ubuntu

MAINTAINER your\_name <user@domain.tld>

RUN apt-get -y install apache2

RUN echo “Hello Apache server on Ubuntu Docker” > /var/www/html/index.html

EXPOSE 80

CMD /usr/sbin/apache2ctl -D FOREGROUND

++++++

FROM centos

RUN yum -y install httpd

COPY index.html /var/www/html

EXPOSE 80

CMD apachectl -D FOREGROUND

+++++

FROM node:6

RUN mkdir -p /usr/src/app

WORKDIR /usr/src/app

COPY package.json /usr/src/app

RUN npm cache cleaN

RUN npm install

COPY . /usr/src/app

EXPOSE 4200

CMD ["npm","start"]

+++++++

$ docker exec -it web ip addr

1: lo: <LOOPBACK,UP,LOWER\_UP> mtu 65536 qdisc noqueue state UNKNOWN

link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00

inet 127.0.0.1/8 scope host lo

valid\_lft forever preferred\_lft forever

inet6 ::1/128 scope host

valid\_lft forever preferred\_lft forever

18: eth0: <BROADCAST,UP,LOWER\_UP> mtu 1500 qdisc noqueue state UP

link/ether 02:42:ac:11:00:03 brd ff:ff:ff:ff:ff:ff

inet 172.17.0.3/16 scope global eth0

valid\_lft forever preferred\_lft forever

inet6 fe80::42:acff:fe11:3/64 scope link

valid\_lft forever preferred\_lft forever

I could accomplish the same thing by running:

$ docker exec -it d2d4a89aaee9 ip addr

Similarly, I could start a shell in the container;

$ docker exec -it web sh

/ # echo This is inside the container.

This is inside the container.

/ # exit

$

#docker file contain

FROM centos:7

RUN yum -y install epel-release

RUN yum -y update

RUN yum -y install nginx

RUN echo "hellow shiv" > /usr/share/nginx/html/index.html

EXPOSE 80/tcp

CMD ["nginx", "-g daemon off;"]

+++++++++

user@node2$ docker -H tcp://node1:port exec -i -t container1 bash

++++++++++++

<https://stackoverflow.com/questions/52062042/docker-compose-nginx-and-tomcat-expose-only-nginx-on-internet>

<https://github.com/dmulligan/docker-example-nginx-tomcat-mysql/blob/master/docker-compose.yml>

++++++++

<https://www.youtube.com/watch?v=bU2NNFJ-UXA&t=508s> (Docker swarm raghav pal)

+++++++++++++++++

http://queirozf.com/entries/angular-2-app-running-on-nginx-on-docker-a-simple-example

(Here dist file should be in /tmp directory because its docker root path)

(Docker file)

FROM nginx

COPY dist /usr/share/nginx/html

EXPOSE 80

Command to run: docker run -p 8900:80 -it my-anguler-app

+++++++

Docker file for tomcat

FROM tomcat:8

COPY ["erp.war","/usr/local/tomcat/webapps/"]

CMD ["catalina.sh","run"]

EXPOSE 8080

+++++

FROM tomcat:8

COPY erp.war /usr/local/tomcat/webapps/

CMD ["catalina.sh","run"]

EXPOSE 8080

++++++

<http://www.dev-garden.org/2014/12/27/setting-up-a-docker-container-with-centos6-and-tomcat7/>

<https://www.youtube.com/watch?v=3dOWkiBEjXY> (Docker file by javahome cloud)

FROM ubuntu:16.04

LABEL maintainer='shiv gurjar’

RUN apt-get -y update

RUN apt-get -y install apache2

RUN apt-get -y install wget

RUN apt-get -y install unzip

WORKDIR /tmp

RUN wget https://github.com/shivg769/project1/archive/master.zip

RUN unzip master.zip

RUN mv warproject1-master/\* /var/www/html/ (one time build with cp and one time with mv , project will not work because it’s war file which runs only in tomcat)

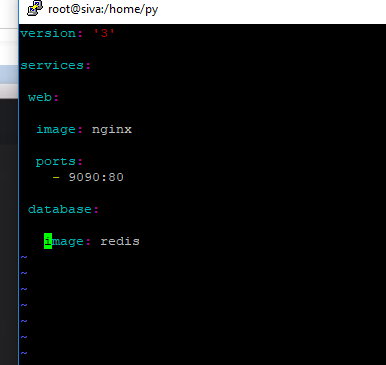
EXPOSE 80

CMD /usr/sbin/apache2ctl -D FOREGROUND

<http://www.dev-garden.org/2014/12/27/setting-up-a-docker-container-with-centos6-and-tomcat7/(Tomcat> with centos dockerfile)

<https://serversforhackers.com/c/dckr-docker-volumes>

Docker – compose



<https://www.penta-code.com/creating-a-lemp-stack-with-docker-compose/>

(Galaxy 1 to 4 project)

<https://www.youtube.com/watch?v=nMLQgXf8tZ0> (Jenkins git docker)

<https://www.youtube.com/watch?v=Z9G5stlXoyg>

<https://www.youtube.com/watch?v=le0WVjLFAv8> (ansible on docker)

<https://docs.docker.com/compose/compose-file/>

<https://tecadmin.net/tutorial/docker/docker-compose-example/>

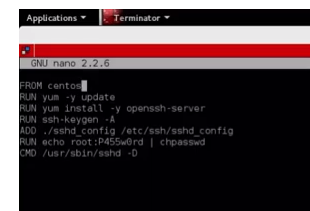
<https://www.linode.com/docs/applications/containers/how-to-use-docker-compose/>

Docker file by Gayatri (Ansible)

**(where is package for ansible and ssh) and I think CMD IS BYDEFAULT**

|  |
| --- |
| FROM williamyeh/ansible:ubuntu14.04-onbuild |
|  | MAINTAINER Sreeprakash Neelakantan <[sree@schogini.com](mailto:sree@schogini.com)> |
|  |  |
|  | COPY ansible/hosts /etc/ansible/hosts |
|  | COPY ansible/playbook.yml /etc/ansible/playbook.yml |
|  | COPY ansible/ansible.cfg /etc/ansible/ansible.cfg |
|  |  |
|  | ENV INVENTORY /etc/ansible/hosts |
|  | ENV ANSIBLE\_HOST\_KEY\_CHECKING False |
|  | ENV PLAYBOOK /etc/ansible/playbook.yml |
|  |  |
|  | COPY ./Dockerfile / |

<https://www.youtube.com/watch?v=ybSKcy2IHGE> (Tomcat modified imagae)

)

<https://hub.docker.com/r/>schogini/docker-ansible-ssh-node-ubuntu/dockerfile/  (Gayatri

<https://github.com/jenkinsci/docker/> (Jenkins image)

Ansible-playbook for centos

- hosts: all

tasks:

- name: Add epel-release repo

yum:

name: epel-release

state: present

- name: install nginx

yum:

name: nginx

state: present

Note: different playbook for Ubuntu node or mentioned condition

For Jenkins issue

**docker run --rm -d -p 9091:8080 -p 5000:5000 -v jenkins\_home:/var/jenkins\_home -v /var/run/docker.sock:/var/run/docker.sock -v /usr/bin/docker:/usr/bin/docker -e JENKINS\_USER=shiv --name my-jenkins jenkins/jenkins:lts**

docker exec -it -u root jenkins bash

sudo mkdir /opt/jenkins\_home

sudo chown -R 1000:1000 /opt/jenkins\_home

(#1 You are trying to map /home/jenkins\_home/ folder on the host machine to the container. You do not have the permissions needed to create such a folder on the host machine since, this is the folder that linux creates when you create a jenkins\_home user.

#2 Confirm if you have /var/run/docker.sock - do an ls and confirm. Please map the docker client too.

#3 Have you created the user shiv on the host machine? The user has to exist. I see you are logged as the root user and not the shiv user.

)

+++++++++

Ansible Dockerfile

**FROM** ubuntu:trusty

**MAINTAINER** Sreeprakash Neelakantan <sree@schogini.com>

**RUN** apt-get update && \

apt-get install -y wget tree curl nano && \

apt-get install -y openssh-server

**RUN** mkdir /var/run/sshd

**RUN** passwd -d root

**RUN** sed -ri 's/^#IgnoreUserKnownHosts\s+.\*/IgnoreUserKnownHosts yes/' /etc/ssh/sshd\_config

**RUN** sed -ri 's/^PermitEmptyPasswords\s+.\*/PermitEmptyPasswords yes/' /etc/ssh/sshd\_config

**RUN** sed -ri 's/^PermitRootLogin\s+.\*/PermitRootLogin yes/' /etc/ssh/sshd\_config

**RUN** sed -ri 's/UsePAM yes/#UsePAM yes/g' /etc/ssh/sshd\_config

**EXPOSE** 22

**CMD** ["/usr/sbin/sshd", "-D"]

+++

+++++++++++++++++++

Jenkins job remove old container and create new

docker ps -aq --filter "ancestor=tomcat" | xarg -r docker rm –f

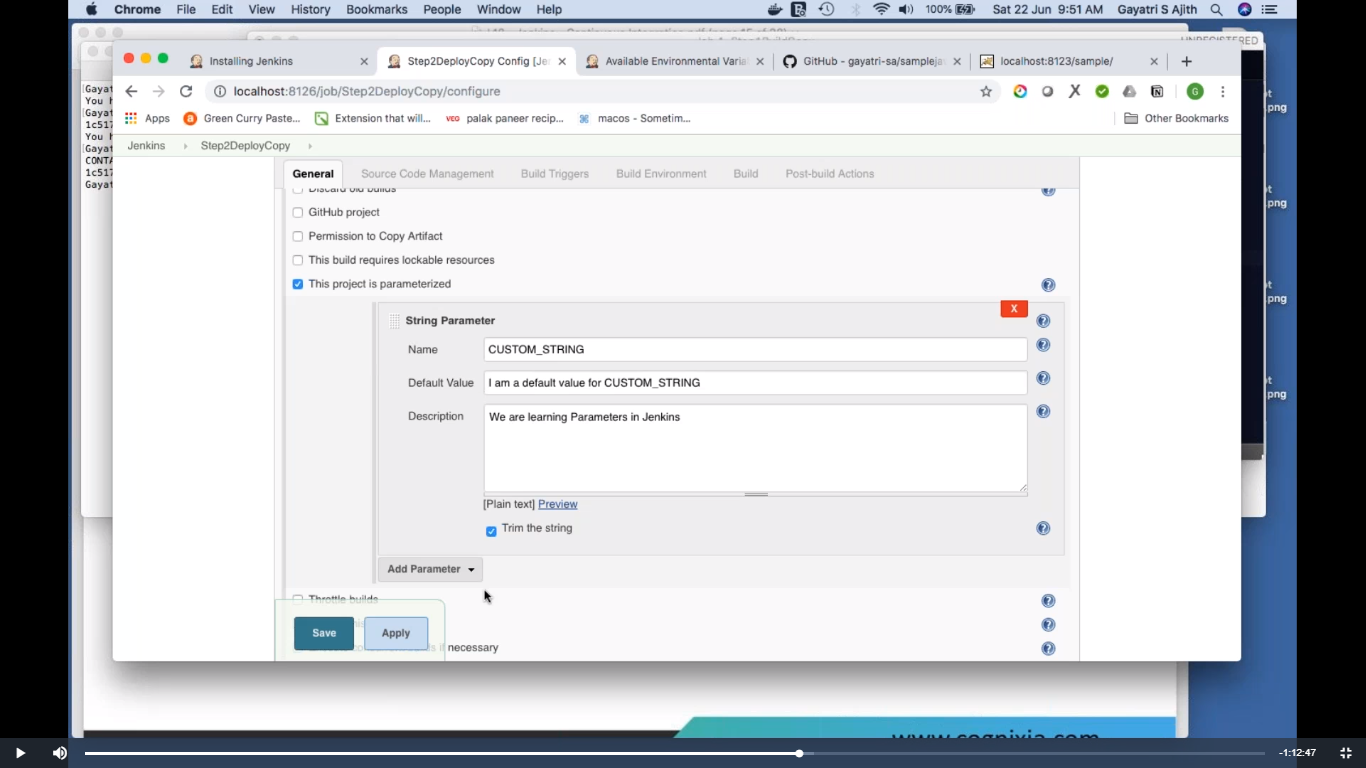
docker run –itd -P –name abc1 –v /home/sample.war:/usr/local/tomcat/webapps/sample.war tomcat

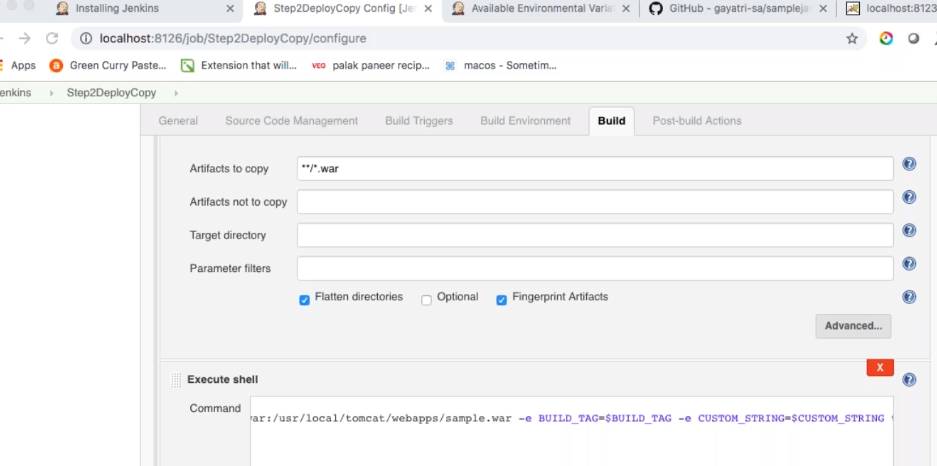
docker run –itd -P –name abc1 –v /workspacefullpath/sample.war:/usr/local/tomcat/webapps/sample.war tomcat

Environment Variable and Parameter build we define in shell script when we create new container(deploy2 and means second job which we will define in first job to execute)

Environment variable Jenkins automatic pickup but parameter build we have to define( in Jenkins project is parameterized option)

-e CUSTOM\_STRING="$CUSTOM\_STRING"





<https://github.com/gayatri-sa/samplejava.git> (java project)

FROM ubuntu

ARG BUILD\_TIME=abc

ENV RUN\_TIME=123

RUN touch /env.txt

RUN printenv > /env.txt

You can override the build arg as you have done with docker build -t temp --build-arg BUILD\_TIME=def .. Then you get what you expect:

> docker run temp cat /env.txt

HOSTNAME=b18b9cafe0e0

RUN\_TIME=123

HOME=/root

PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin

BUILD\_TIME=def

PWD=/

**Docerk file using ENV**

FROM ubuntu:trusty

RUN apt-get update

RUN apt-get install -y nano vim nginx

ADD index.html /usr/share/nginx/html/index.html

ARG BUILD\_TIME=abc

ENV run\_time=123

ENV HOSTNAME=NODE1.sbi

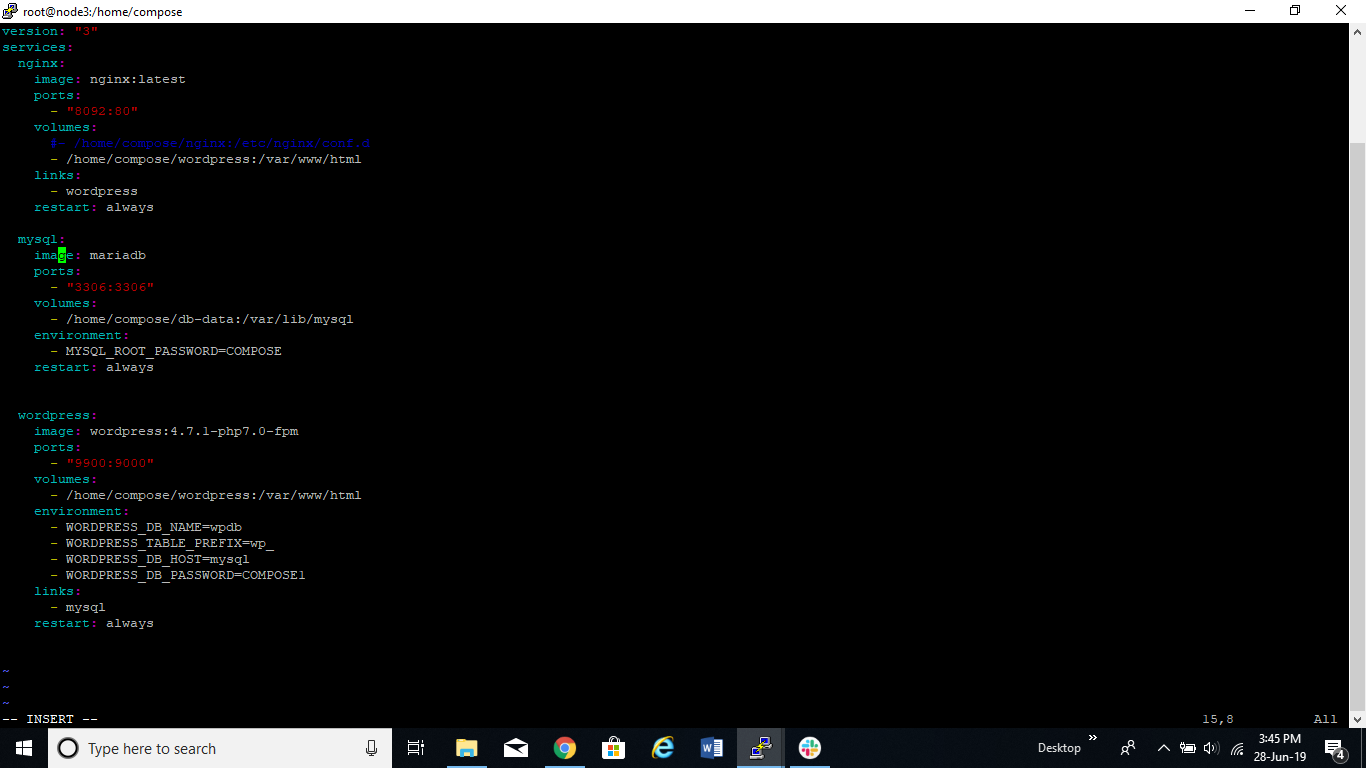
RUN touch /env.txt

RUN printenv > /env.txt

EXPOSE 80

CMD /usr/sbin/nginx -g "daemon off;"

Docker-compose



(Galaxy 1 to 4 project)

<https://www.youtube.com/watch?v=nMLQgXf8tZ0> (Jenkins git docker)

<https://www.youtube.com/watch?v=Z9G5stlXoyg>

<https://www.youtube.com/watch?v=le0WVjLFAv8> (ansible on docker)

<https://github.com/ValaxyTech/hello-world.git> (git repo)

MobaXterm tool for ssh ec2

**(CICD ) tested working fine (Output – create container on docker using Jenkins)**

<https://www.youtube.com/watch?v=nMLQgXf8tZ0>

Git- Jenkins – docker

Plugin public over sssh

Jenkins machine

Docker machine

Create docker user on docker machine

Useradd dockeradmin

Usermod –a –G docker dockeradmin

/etc/ssh/sshd\_config

PasswordAuthentication yes

Chown –R dockeradmin:dockeradmin /opt/docker

Chmod 777 /var/run/docker.sock

//opt//docker (in remote dir // should be there)

Dockerfile

FROM tomcat:8-jre8

COPY ./webapp.war /usr/local/tomcat/webapps

URL is <http://192.168.1.149:8090/webapp/>

1 step

Cd /opt/docker

docker stop valaxy\_demo; docker rm -f valaxy\_demo; docker image rm -f valaxy\_demo; cd /opt/docker; docker build -t valaxy\_demo .

**2 step**

docker run -d --name valaxy\_demo -p 8090:8080 valaxy\_demo

