Lab Scenario – How to run Tomcat inside Docker with the Maven war file

Steps

1. Copy the War file that is already generated by the Maven project
2. Create the Dockerfile
3. Build the Docker image
4. Create Container

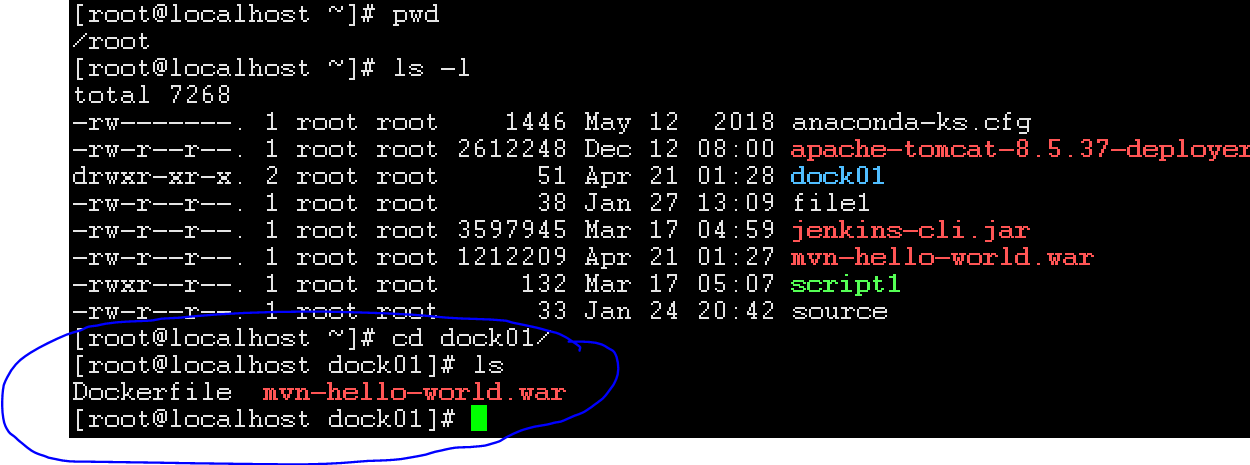
**Step1**: Copy the War file that is already generated by the Maven project

The scenario here is the war file is generated by the Maven project running in the Jenkins server.

Under the folder **/var/lib/jenkins/workspace/maven-proj1/target**

**Copy the file “mvn-hello-world.war” to the docker folder**

**In our case we have created a docker folder called “dock01” as below**



Command:

**$ cp /var/lib/jenkins/workspace/maven-proj1/target/mvn-hello-world.war /root/dock01/mvn-hello-world.war**

**Step2: Create the Dockerfile.**

**Now we would need the “Tomcat” image and also the .war file to be copied to the Tomcat “webapps” folder while creating the container.**

**Below is the Dockerfile content.**

==========================  
FROM tomcat:7.0.90-jre8  
WORKDIR /app1  
  
ADD mvn-hello-world.war /usr/local/tomcat/webapps/mvn-hello-world.war  
  
EXPOSE 8080

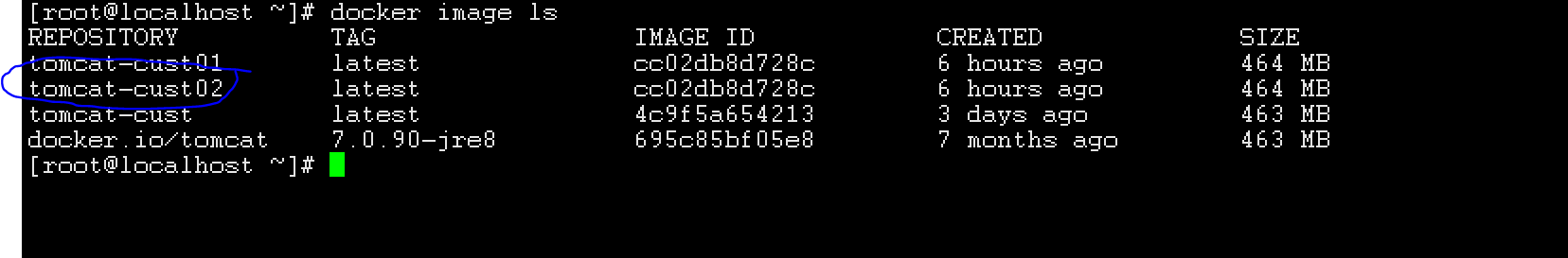
==========================

**Step3: Build the Docker image**

Run the below command

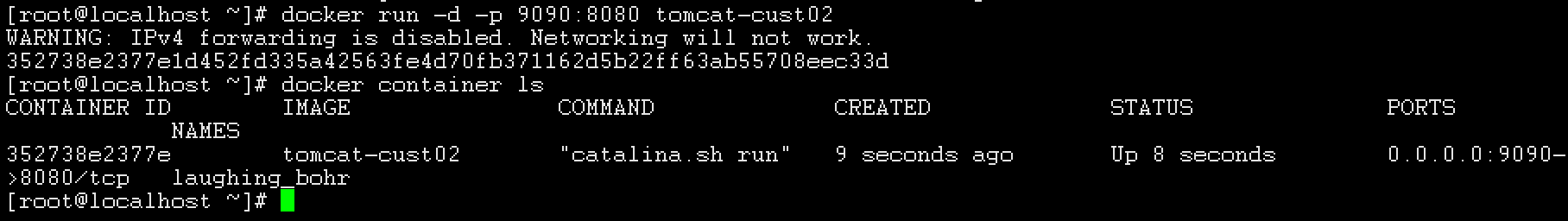
**$ docker build dock01 -t tomcat-cust02**





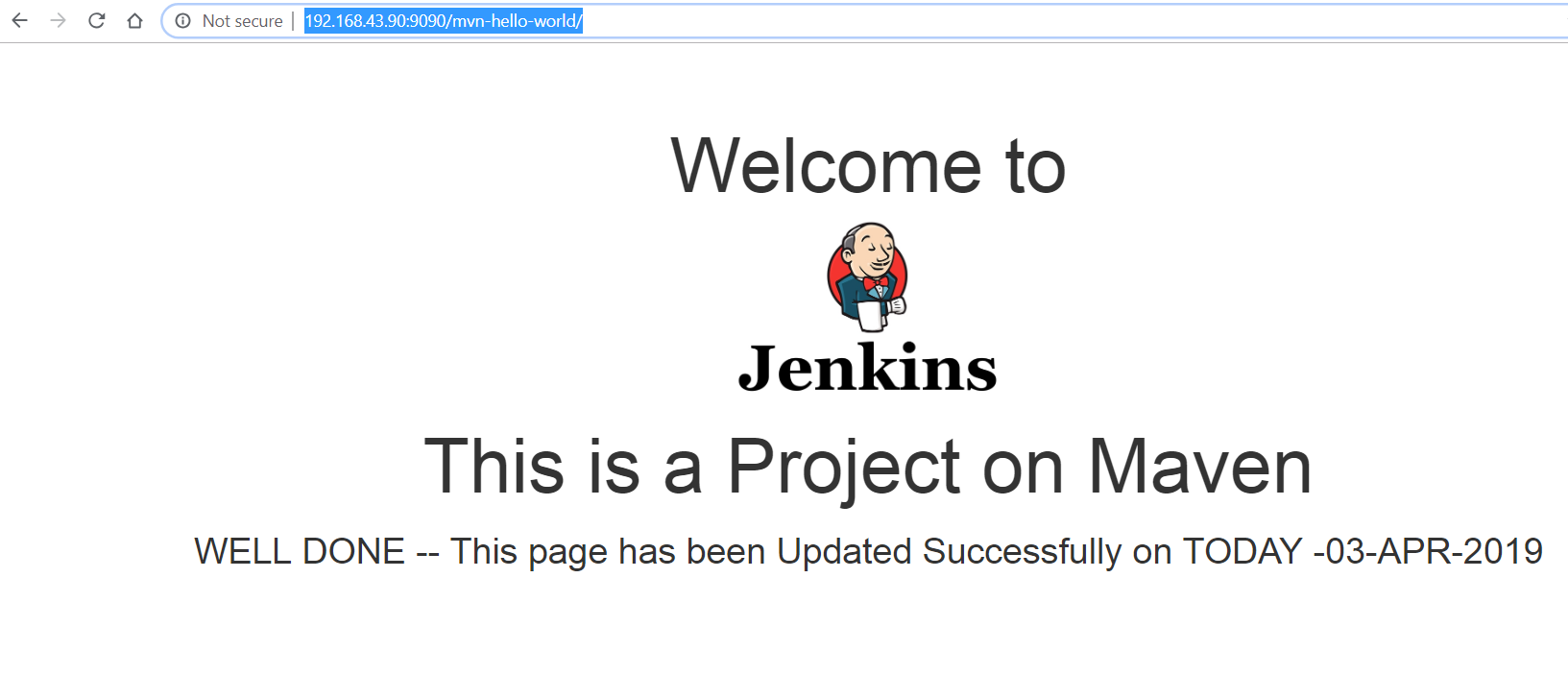
1. **Create the Container**

**$ docker run -d -p 9090:8080 tomcat-cust02**



1. **Test the tomcat output.**

http://<Docker server ip>:9090/mvn-hello-world/



This was very easy and straight forward than setting up the Tomcat server and dependencies.