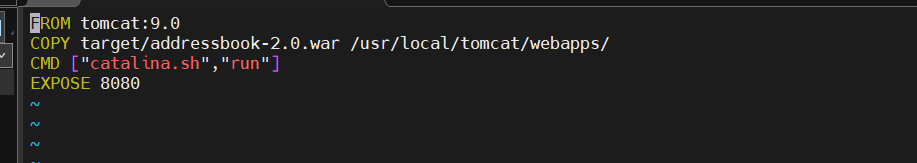
**Docker Containerized AddressBook Application**

Step1 : First clone addressbook application using git clone command

git clone <https://github.com/Abhijeet41/addressbook-demo.git>

Step2: navigate to this address-book application create docker file  
o/p

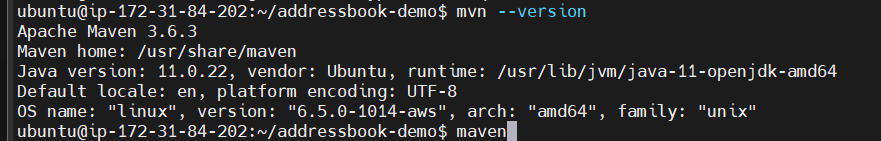


Step3: first install maven to run this addressbook application so it will generate .war file so we can copy

that in ../tomcan/webapps path.

sudo apt install maven

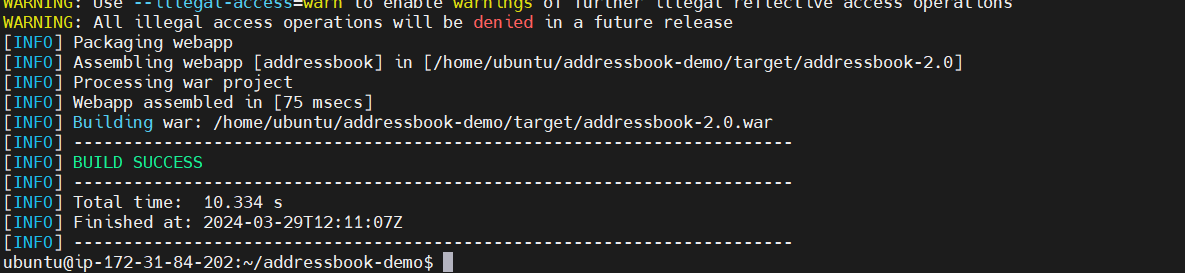
o/p



Step4: now install below command to create .war file

mvn clean package

o/p : here we get our .war file

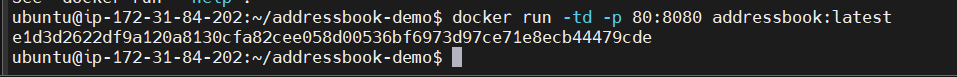


Step5: Now lets create Docker Build

docker build -t addressbook .

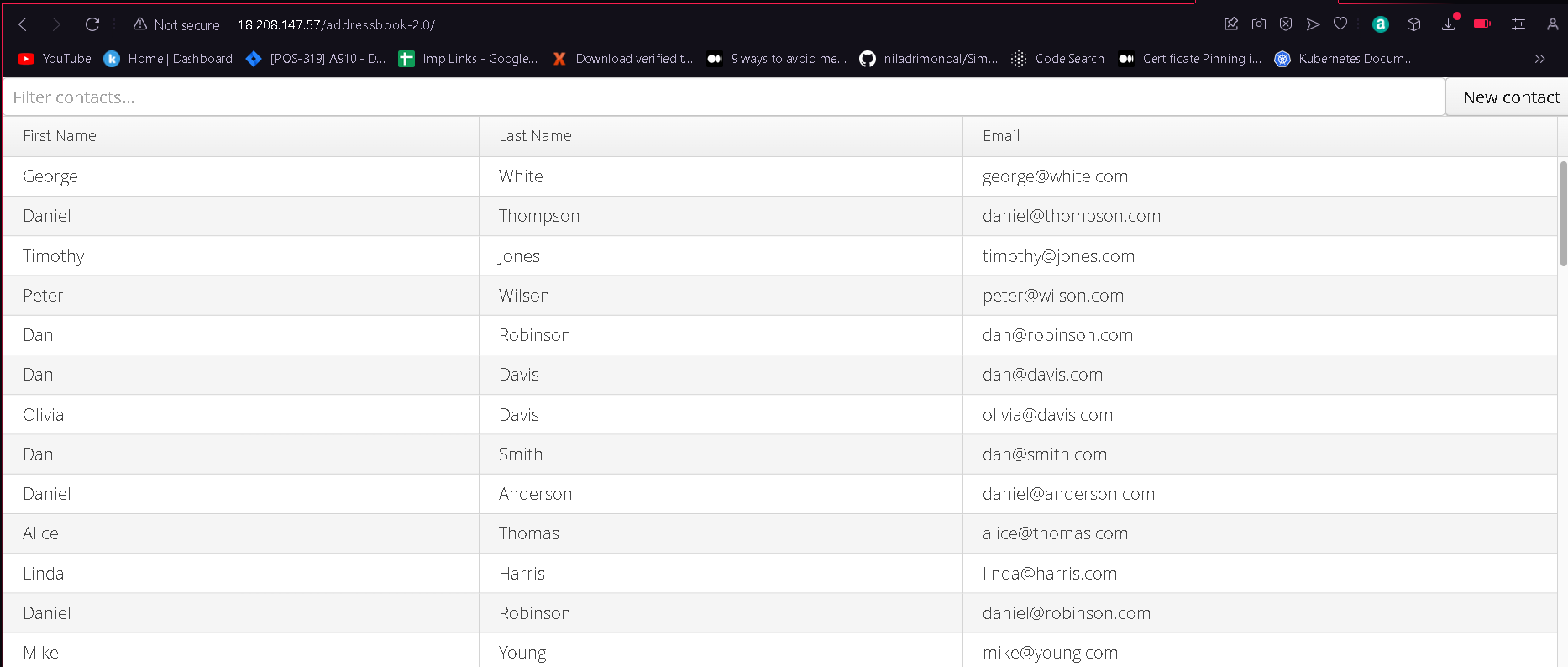
Step6: run Addressbook image

docker run -td -p 80:8080 addressbook:latest



Finally you will be able to see on browser addressbook app is hosted but before check on browser make sure port 80 is enabled in security group

Note .war file name should be put in browser along with ip and port .war = addressbook-2.0



**-------------------Another MVN based application host using docker---------------------------**

Step 1: Make sure docker is already installed and clone project repository from github

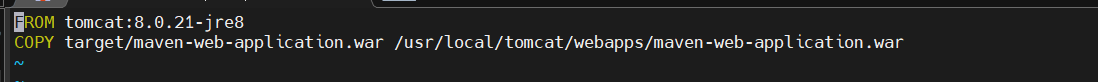
git clone <https://github.com/MithunTechnologiesDevOps/maven-web-application.git>

Step2: Get into that project repositoly and creat Dockerfile using below command

vi Dockerfile

FROM tomcat:8.0.21-jre8

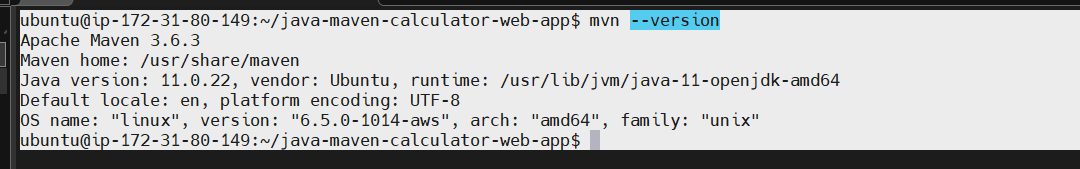
COPY target/maven-web-application.war /usr/local/tomcat/webapps/maven-web-application.war



Step3: first install maven to run this addressbook application so it will generate .war file so we can copy

that in ../tomcan/webapps path.

sudo apt install maven



Step4: now install below command to create .war file

mvn clean package

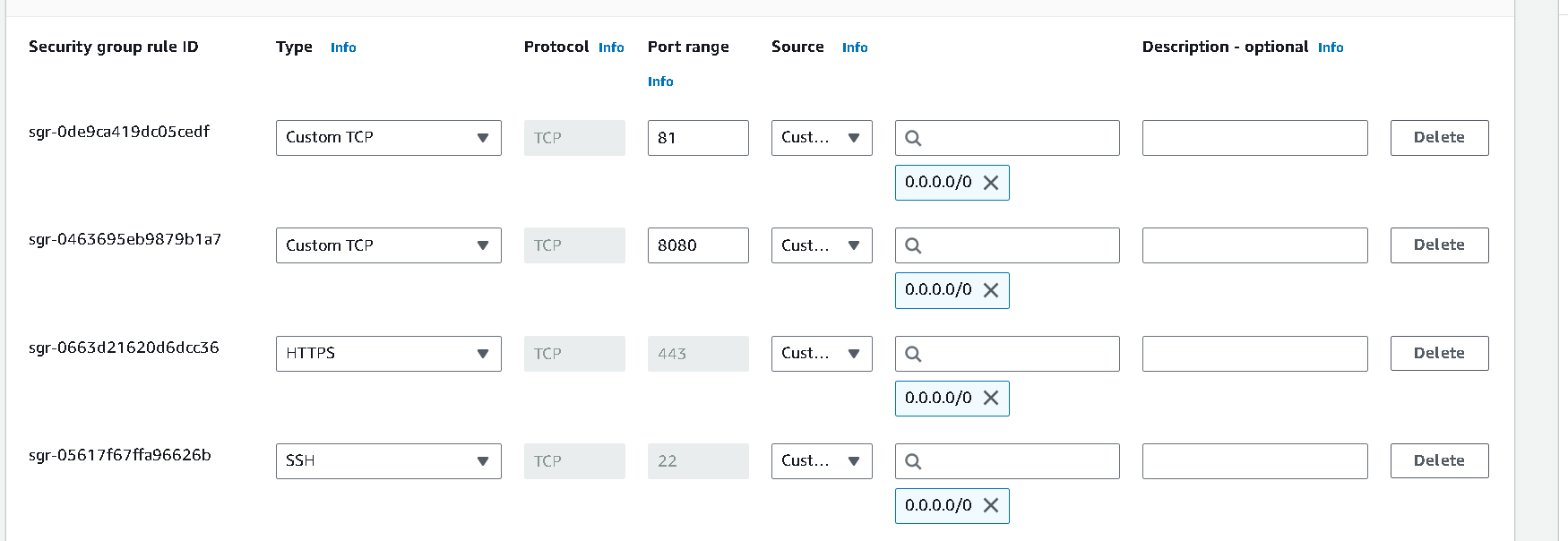
Step5: Now lets create Docker Build

docker build -t mithun \_technology .

Step6: run mithun \_technology image

docker run -td -p 81:8080 mithun \_technology :latest

Finally you will be able to see on browser mithun \_technology app is hosted but before check on browser make sure port 81 is enabled in security group



Now finally see the output on web browser

Note .war file name should be put in browser along with ip and port .war = maven-web-application/



