**DevOps Lab 2.2: Docker Images**

Working with Docker images

Open the Cloud Platform Console at [https://console.cloud.google.com](https://console.cloud.google.com/).

Click on the three horizontal bars at the left most side of the blue bar near the top of the browser window. *Select Compute Engine*.

Select *VM Instances*. You should see the virtual machine you created earlier.

Click on the checkbox to the left of the VM name and then select *START*. It will take a few moments to start.

Click on *SSH* to start a terminal window.

**Change the host name to student:** Find the icon that looks like a gear in the upper right-hand corner of this terminal browser window and select *Change Linux User Name*. Enter *student* and *click Change*. Now, notice the prompt that says "student@lab:~$"

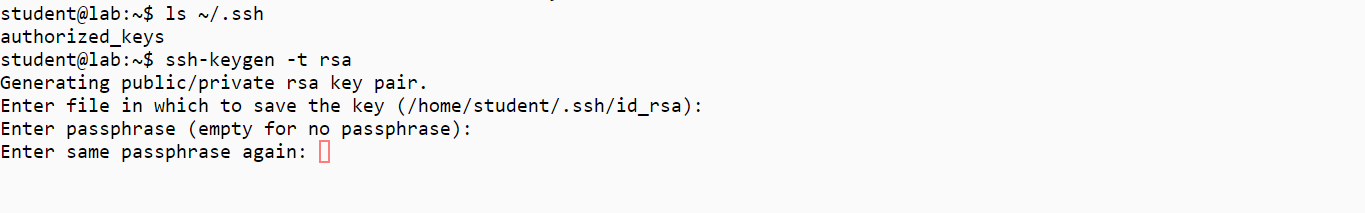


Create an SSH client key pair (public and private). Check to see if you have SSH keys generated.

*ls ~/.ssh*

If you see only the file “authorized\_keys,” then you need to perform the following step to generate an SSH client key.  
--> Accept the default file  
--> Do not enter a passphrase [Please refer the screenshot]

*ssh-keygen -t rsa*



Prepare a Dockerfile to create a new Docker image so create a subdirectory to hold the context for the Docker build operation.

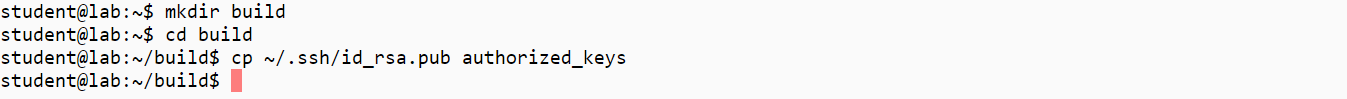
*mkdir build*

Enter that directory.

*cd build*

Copy the public part of the SSH key pair into the build directory.

*cp ~/.ssh/id\_rsa.pub authorized\_keys*



Create a Dockerfile with the following contents. (Use *vi Dockerfile* or *nano Dockerfile*).

FROM alpine:latest  
RUN apk update  
RUN apk add openssh  
RUN adduser -g "Student User" -D student && mkdir /home/student/.ssh  
RUN echo "student:student" | chpasswd  
ADD authorized\_keys /home/student/.ssh  
RUN chown -R student.student /home/student  
RUN chmod 700 /home/student/.ssh && chmod 600 /home/student/.ssh/authorized\_keys  
RUN ssh-keygen -t rsa -f /etc/ssh/ssh\_host\_rsa\_key -q -N ""  
EXPOSE 22  
CMD ["/usr/sbin/sshd", "-D"]

Create the docker image using the Dockerfile commands.

*docker build -t ssh:alpine .*

**Note:** Build status should be successful as shown below



See that the image got created.

*docker images*

Run the new container which contains your public SSH key, and run the SSH daemon.

*docker run -d -p 2022:22 --name ssh ssh:alpine*

If you encounter the below error, if not skip this step:

docker: Error: image library/ssh:alpine not found.

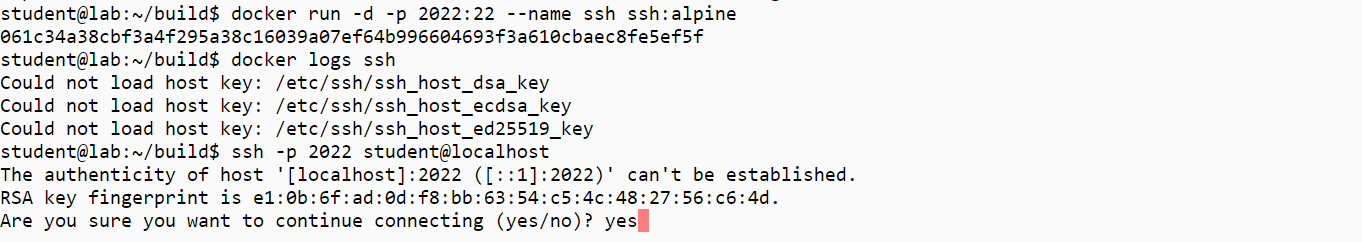
Then try:

docker run -d -p 2022:22 --name ssh alpine

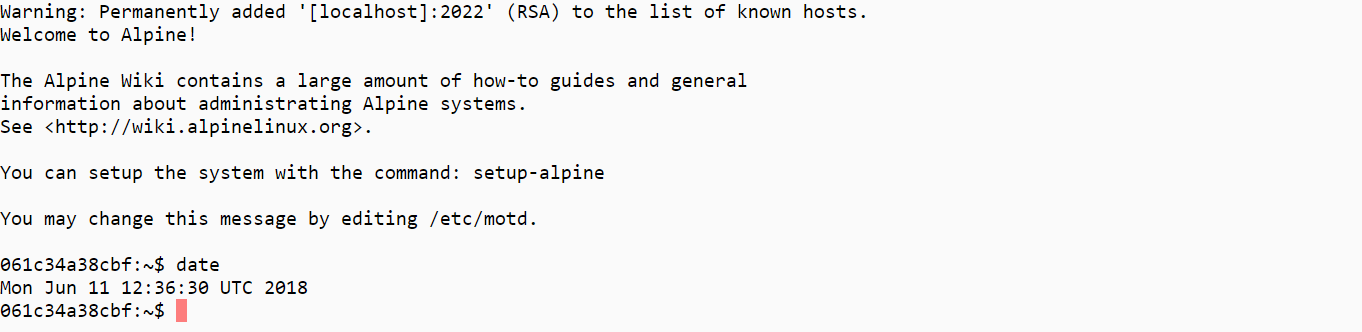
Make sure there are no errors.

*docker logs ssh*

Connect to the Docker container using your Linux ssh client on your computer.



You are now inside the Alpine Linux container.   
Please explore!



Exit by typing exit or control-D and Clean up

*docker stop ssh  
docker rm ssh  
docker rmi ssh:alpine  
docker rmi alpine:latest*



You will need to stop the lab computer at the end of each day to prevent it from accumulating costs during the evening and night.

From the Web UI, you can navigate to the Compute Engine section and select your lab computer. When it is selected, click on the icon representing the "Stop" operation as shown below:

