**Capstone - Exercise 4: Jenkins and Git**

In this exercise, you will integrate Jenkins and Git in Docker containers.

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

**Step-1**

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:~$"



Check that the Firewall allows traffic on port 8080. It should have been set up in an earlier lesson.

Delete SSH hosts to prevent problems later.

*rm -r ~/.ssh/known\_hosts*

Change to the exercise directory.

*cd  
cd devops-capstone/lab-4*

Create a directory for Jenkins data.

*mkdir jenkins  
chmod 777 jenkins*

**Step 2**

Get a Jenkins image from Docker Hub.

*docker pull jenkins*

Start jenkins.

*docker run -d --name jenkins -p 8080:8080 -p 50000:50000 -v $PWD/jenkins:/var/jenkins\_home/ jenkins*

Wait until Jenkins is up and running. Use:

*docker logs jenkins*

**Step 3**

We need to ensure that we can connect from Jenkins to Git via SSH. First, run an interactive Jenkins container and create SSH keys.

*docker run -it --rm -v $PWD/jenkins:/var/jenkins\_home jenkins /bin/sh*

Generate SSH keys.

*ssh-keygen -t rsa*

*Press* ***enter*** *3 times.*

Exit the container with control-D.

Copy the public keys for the student and Jenkins users.

*sudo cat ~/.ssh/id\_rsa.pub jenkins/.ssh/id\_rsa.pub > git/authorized\_keys*

Build a Git image.

*docker build -t git git*

Start a Git container.

*docker run -d -p 2022:22 --name git git*

Find its IP address and make a note of it.

*docker inspect git*

Now run a Jenkins container again, and verify that you can connect.

*docker run -it --rm -v $PWD/jenkins:/var/jenkins\_home jenkins /bin/sh*

Now connect to the Git container using SSH. You may have to change the last digit of the IP address to that of your container.

*ssh git@172.17.0.3*

Answer yes to accept the key.  
You should connect OK.  
Exit both containers with control-D (twice).

Clone the Git repository. Ignore the warning about the repository being empty.

*git clone ssh://git@localhost:2022/home/git/project.git*

Type **yes**

Check in some data.

*cd project  
echo "First message" > readme.txt  
git add readme.txt  
git commit -m "First checkin"  
git push*

**Step 4**

You will need an admin password. Get the Jenkins logs, and scroll back to find the password. Copy it ready to paste later.

*docker logs jenkins*

Start a browser on your local machine and enter the URL x.x.x.x:8080, replacing x.x.x.x with the external IP address of your virtual machine.

You will see an *Unlock Jenkins* screen. Paste the admin password and hit *Continue*.

Run the below command with the path provided.

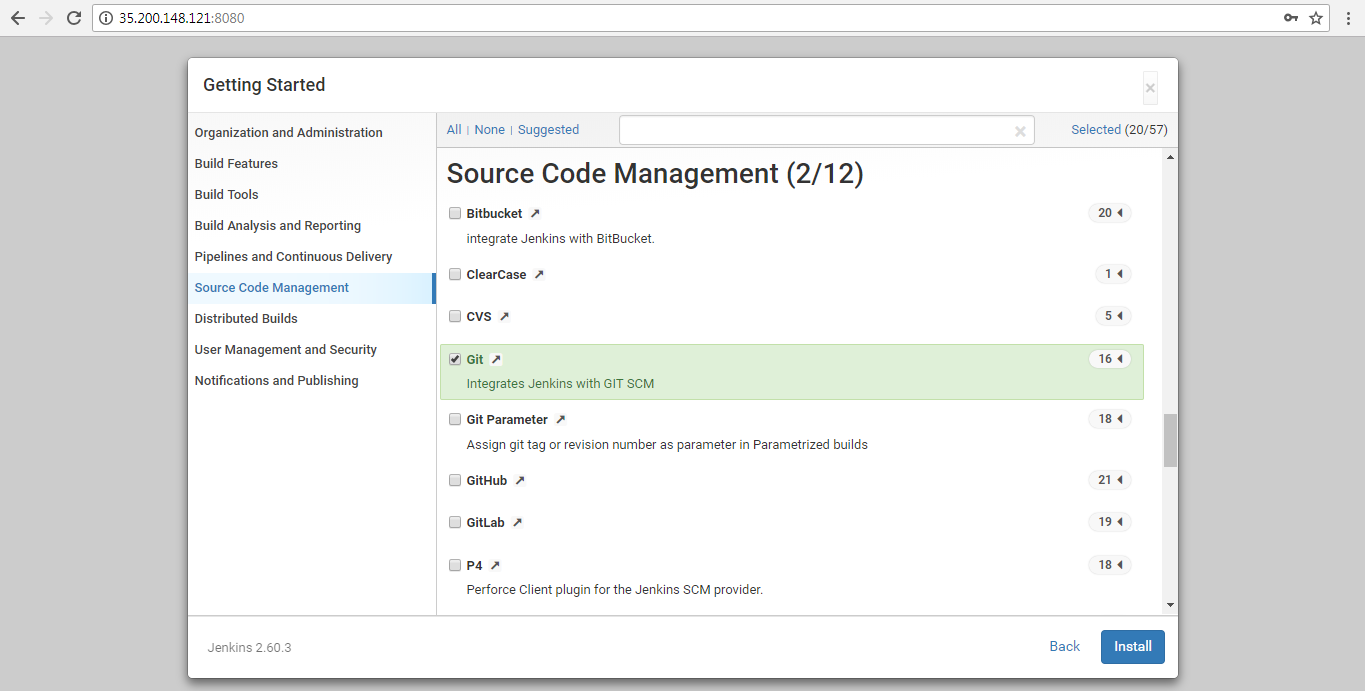
*sudo cat <pathname>.*

If you don’t find the path then locate the Jenkins folder and once you find it, run the following command.

*Sudo cat /Jenkins<Pathname>*

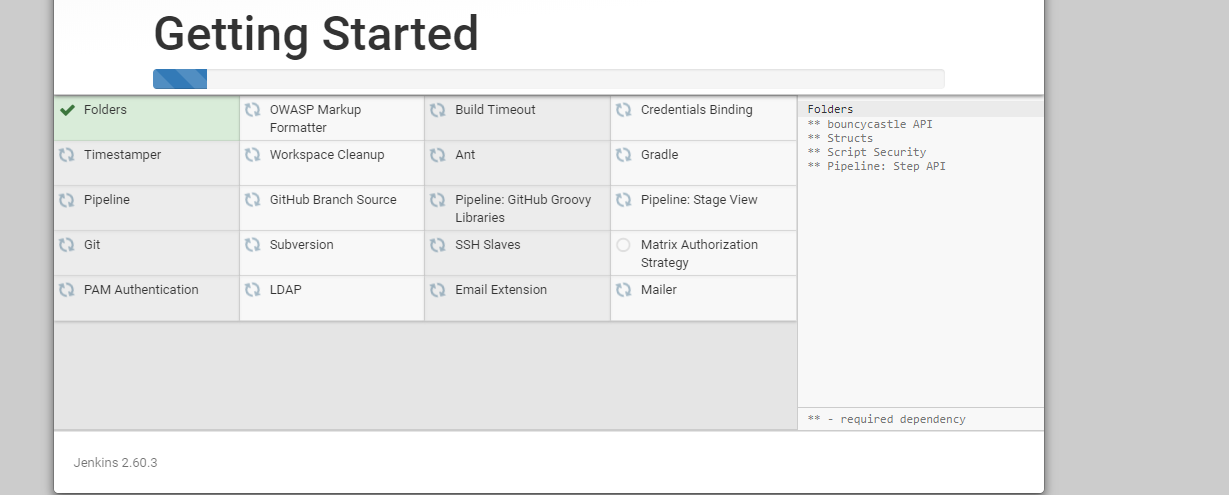
The next screen is *Customize Jenkins*. Hit *Select plugins to install*.

Select *Source Code Management*. Make sure that *Git plugin* is selected.



Select *Install*.

A *Getting Started* screen will be displayed, and plugins will start downloading. This will take a few minutes.



You will next be asked to create an admin user. Fill in the form and select *Save and Finish*.

Jenkins is set up. Select *Start using Jenkins*.

**Step 5**

Select *create new jobs*.  
Enter the name *DevOps*.  
Select *Freestyle project*.  
Hit *OK*.

Scroll down to *Source Code Management*.  
Select *Git*.  
Enter the following URL, possibly changing the last digit of the IP to your Git IP.

*ssh://git@172.17.0.5/home/git/project.git*

Hit *Apply*. The URL should be accepted. If there is an error in red, check the URL.  
Hit *Save*.

You will be taken to the *Project DevOps* page.

Select *Build Now*. The project should build.  
Click on *#1* at the bottom of the page. It will show the build and the date and time.  
Select *Console Output*.  
You should see the details of the build.

**Step 6**

Stop the servers.

*docker stop jenkins git*

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:

