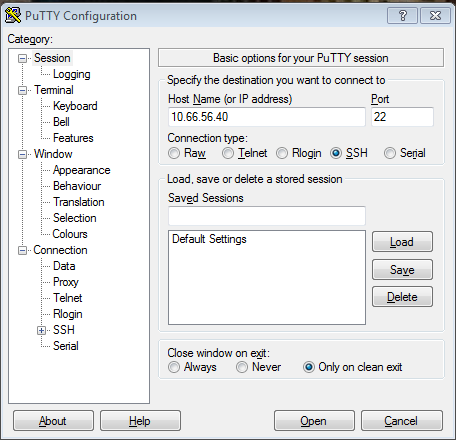
***Steps for login-***

1.Download putty 64 from software house. Start virtual machine from Putty.

Using IP of your VM start the machine.

******

2.login to the VM using ID and password of the VM.

Initially we have to login as admin.

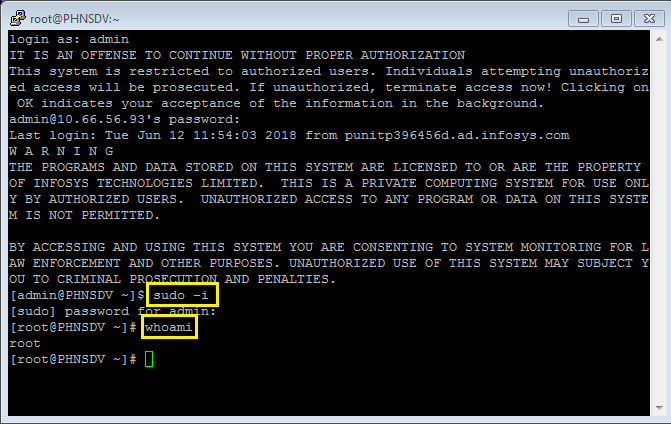
3. After that we have to login as root.

Commands for that-

#sudo -s or #sudo –i

Command to check current logged in user

#whoami



***For internet access-***

navigate to /etc folder and add the following proxies inside the bin folder.

export https\_proxy="https://username:password @10.136.64.150:80"

export http\_proxy="http://username:password @10.136.64.150:80"



Note- sometimes adding proxy will not work at that time we have to unset proxy.

***Installation of Git, Docker and Jenkins-***

We have to Raise AHD for downloading and installing any software because we are using Infosys Network, if we try using commands for downloading, it will give error of Red Hat Satellite not registered or RHN Classic support will be disabled.

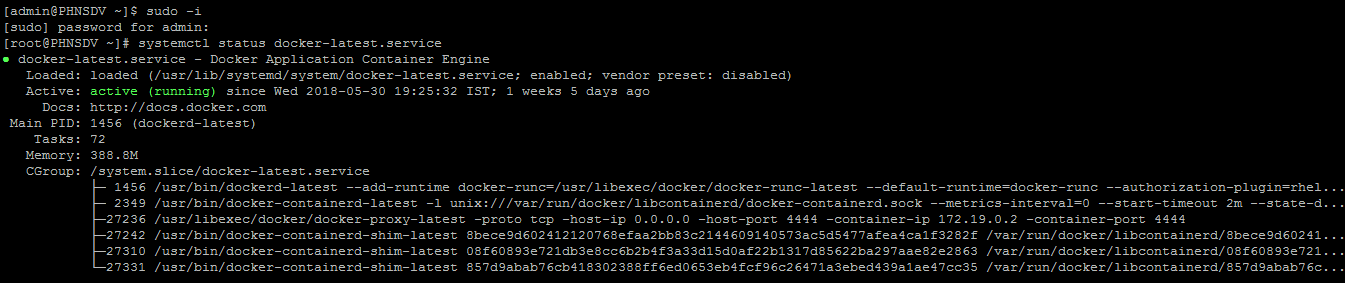
**1. Docker-**

For Installing Docker, raise an AHD under CCD. (install latest version of docker).

i.e 1.13.1

To check the docker status-

#systemctl status docker-latest.service

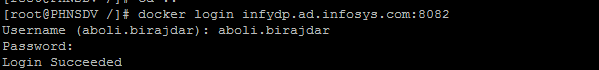


***Accessing Infosys nexus server*** –

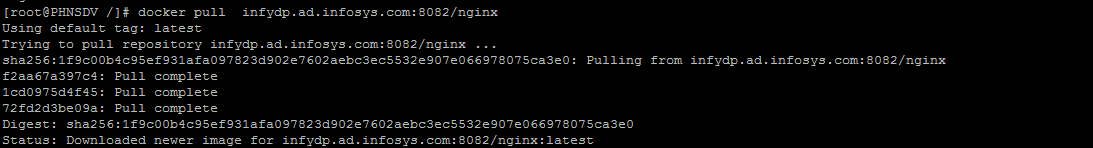
For Docker we have to login using our Infosys Id and password to docker login infydp.ad.infosys.com:8082

Command for the same-

#docker login infydp.ad.infosys.com:8082



* Then test the Pull & Run command for the required docker Image and make sure that the download is done.



For more information about Nexus based Docker proxy, you can refer the website <http://wiki/Docker>

* If environment variables are not present in daemon.json file then it will show Error response from daemon: Get <https://infydp.ad.infosys.com:8082/v1/users/>: http: server gave HTTP response to HTTPS client "   message.

Then we have to add environment variables.

Steps for that as follows-

1. Place the following content in /etc/docker/daemon.json

{

"registry-mirrors": [

"infydp:8082",

"infydp.ad.infosys.com:8082",

"10.82.13.222:8082",

"infydp.ad.infosys.com:8082",

"infydp:8082","idpwinv05:8082"

],

"insecure-registries": [

"infydp:8082",

"infydp.ad.infosys.com:8082",

"10.82.13.222:8082",

"infydp.ad.infosys.com:8082",

"infydp:8082","idpwinv05:8082"

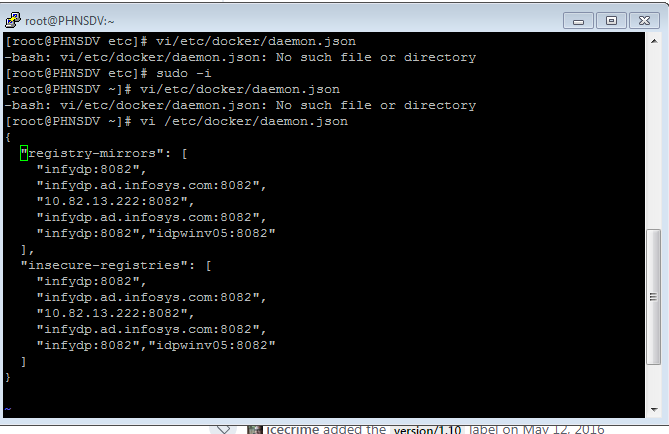
]

}

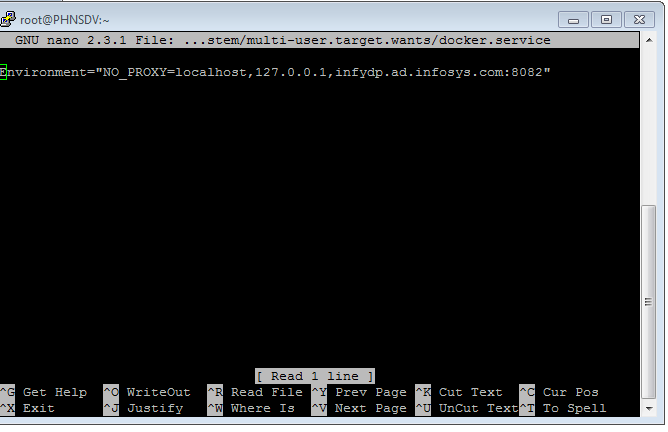
2.Then run the below commands:

#systemctl restart docker. service

# systemctl daemon-reload



3.If that also, doesn’t work, insert the following line in /etc/systemd/system/multi-user.target.wants/docker.service Environment="NO\_PROXY=localhost,127.0.0.1,infydp.ad.infosys.com:8082"



**Steps to install Docker compose-**

* Contact with CCD person or Raise AHD to install Docker compose.
* If you are using AKASH portal, then follow the steps to install Docker compose-

1. Download docker-compose Linux executable from:

<https://github.com/docker/compose/releases>

1. Go to the directory where the executable is downloaded in terminal and execute the following commands:
2. sudo mv docker-compose-Linux-x86\_64 /usr/bin

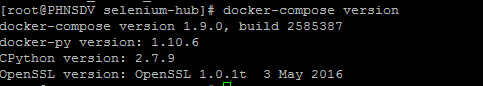
cd /usr/bin

sudo mv docker-compose-Linux-x86\_64 docker-compose

sudo chmod 755 docker-compose

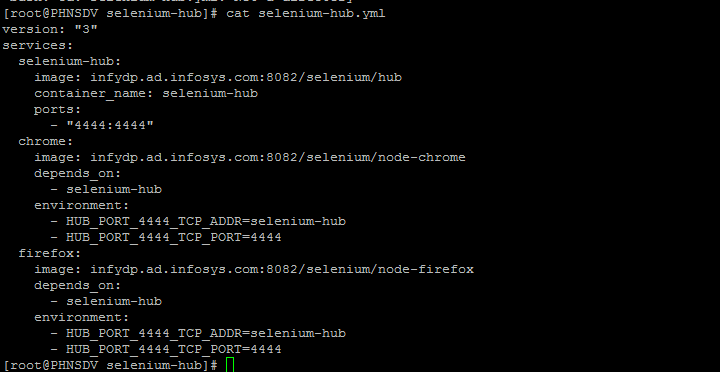
Command to check docker compose version

#docker-compose version



***Selenium hub setup through docker-compose: -***

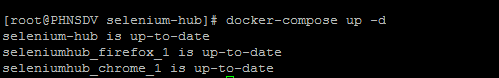
* Create a docker-compose.yml file. This file should be placed in a directory with the following contents



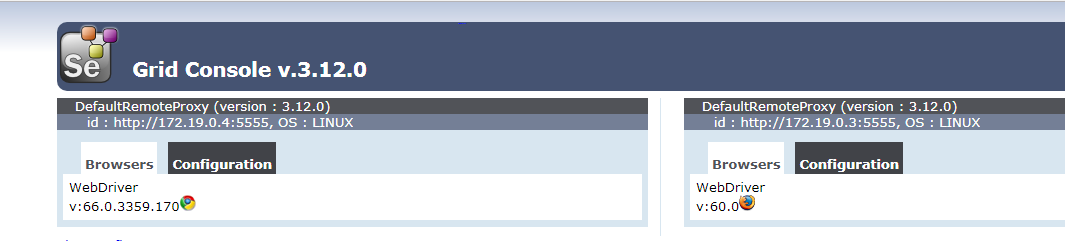
use the following command after navigating to the folder of docker compose file:

#docker-compose up –d

* This will pull the official images for hub, node-chrome, and node-firefox from Docker if doesn’t exist in docker and start a grid with one Instance of each browser available.

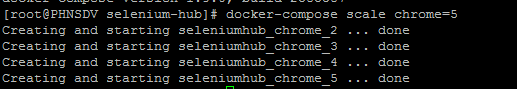


* Your hub will be available through the [http://10.66.56.93:4444/grid/console URL where 10.66.56.93](http://10.66.56.93:4444/grid/console%20URL%20where%2010.66.56.93) is IP of your VM.

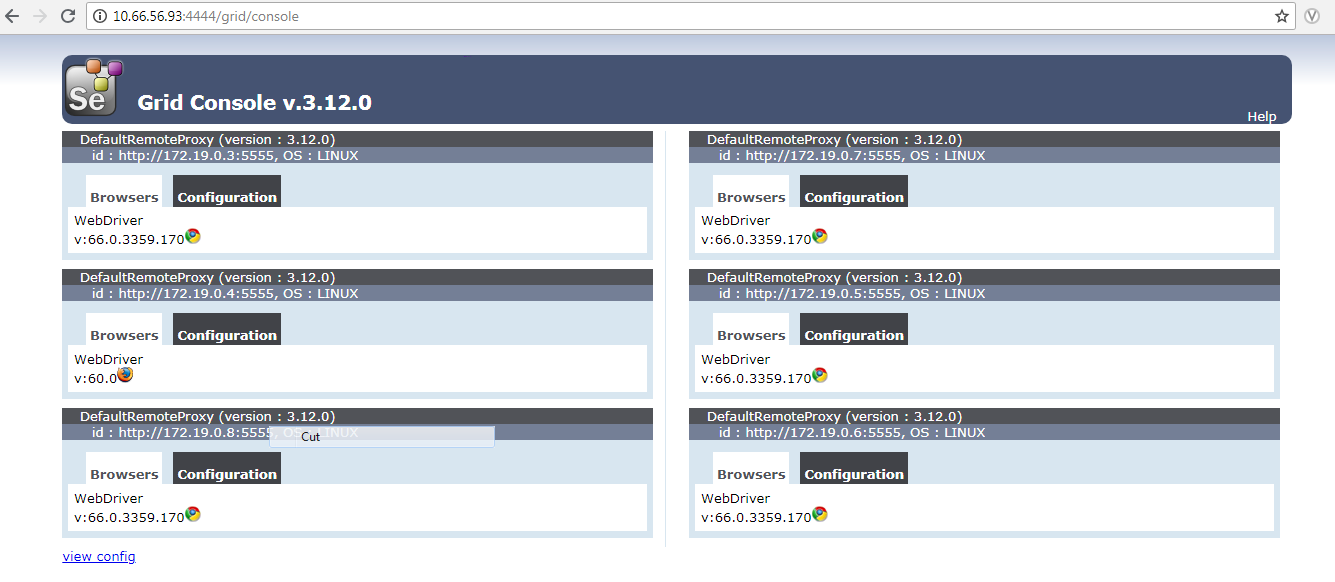


* We can increase the number of nodes by setting the number of containers to run for a service though the following command.

#docker-compose scale chrome=5

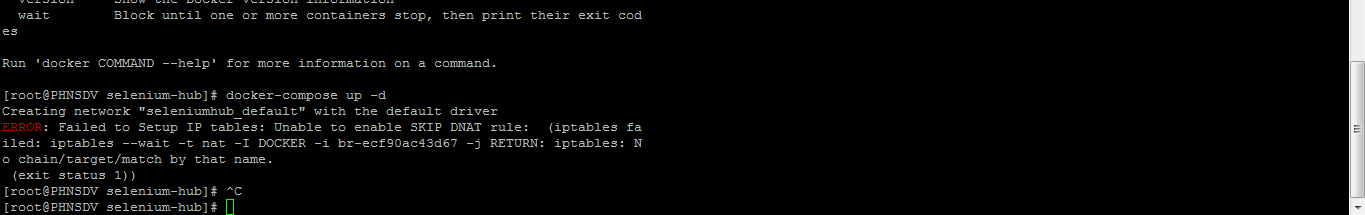


* Here 5 Nodes were created after executing the above command.



***IP tables issue in Docker-***

Sometimes while running Docker-compose up we can get IP tables not found issue then stop the firewall service. or we can use command



service iptables stop

**2.Git-**

For Installing Git raise an AHD under CCD.

Command to check the status of Git using command

#which git

C:\Users\aboli.birajdar\Desktop\sc2.PNG

***Steps to add code in Infygit and pull it in VM-***

1. Login into Infygit using Infosys User ID and Password

Link- http://infygit.ad.infosys.com/users/sign in

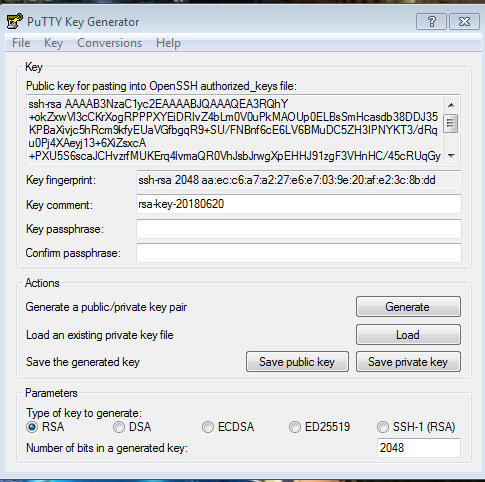
1. Create a new public project in Infygit.
2. To pull or push any project from infygit we have to add an SSH key in the infygit.

**Steps to create an SSH key from putty-**

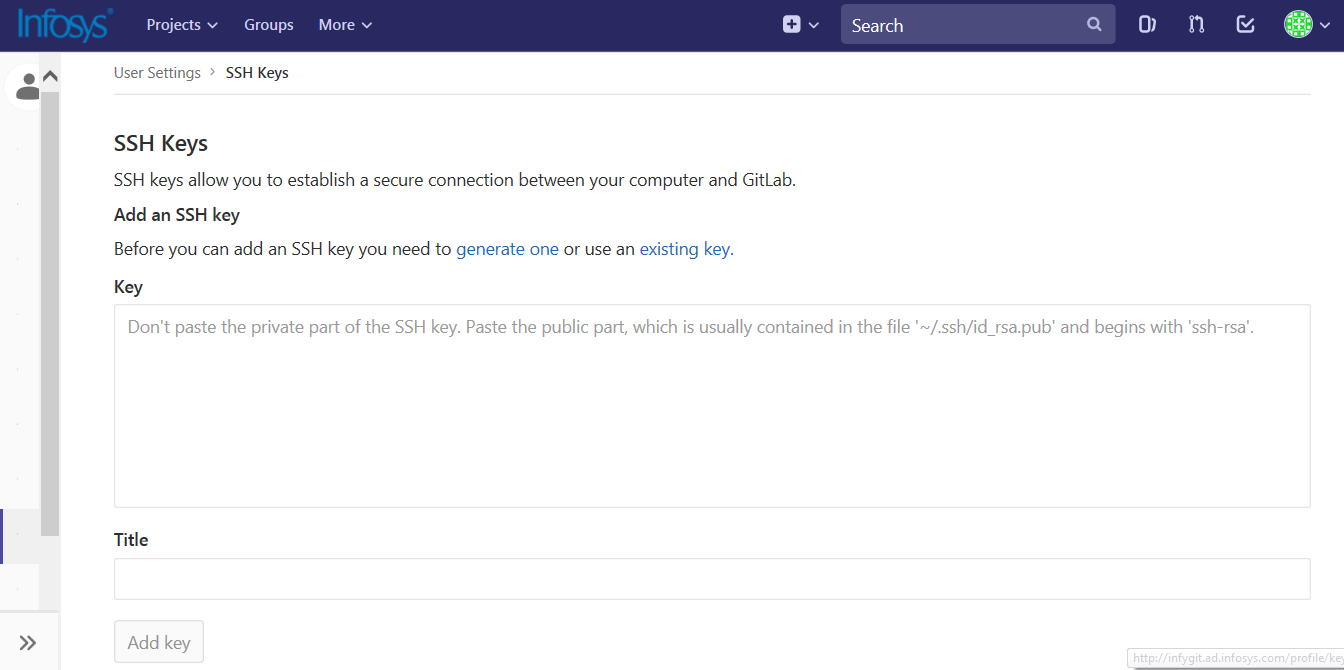
PuTTY is an SSH client for Windows that you will use to generate your SSH keys.

PuTTYgen is what you will use to generate your SSH key for a Windows VM.

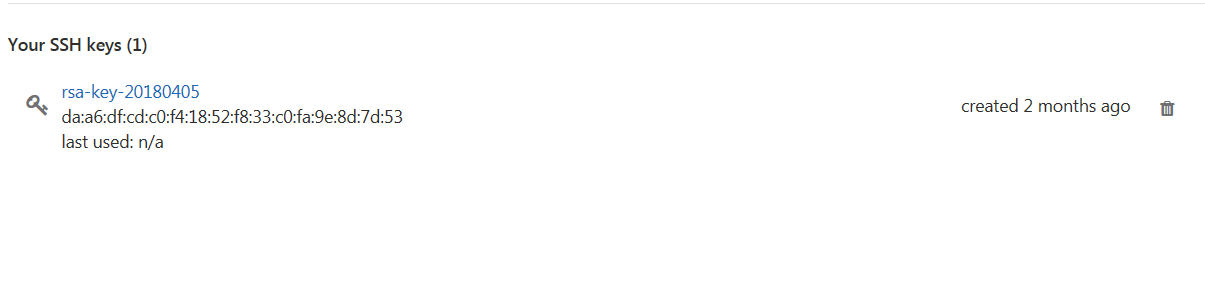
1. Download putty 64 from software house.
2. Open the PuTTYgen program.
3. For Type of key to generate, select SSH-2 RSA.
4. Click the Generate button.
5. Move your mouse in the area below the progress bar. When the progress bar is full, PuTTYgen generates your key pair.
6. Type a passphrase in the Key passphrase field. Type the same passphrase in the Confirm passphrase field. You can use a key without a passphrase, (Not mandatory).
7. Click the Save public key button to save the public key.to use further if needed.
8. Right-click in the text field labeled Public key for pasting into OpenSSH authorized\_keys file and choose Select All.
9. Right-click again in the same text field and choose Copy.



1. After you copy the SSH key to the clipboard, return to infy git page navigate to user settings—SSH keys
2. Choose to **Import Public Key** and paste your SSH key into the Public Key field.
3. Paste the key in infygit—user settings—SSH keys



1. **Add** the key. It will now appear in your table of keys under SSH.



4.Run gitbash as Administrator (for Admin rights required).

* **Commands to push the code from Folder to infygit -**
* **git config**

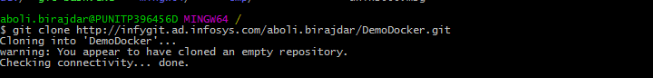
#git config --global user.name "user name"

#git config --global user.email "user@domain.com"

Sets configuration values for your user name, email

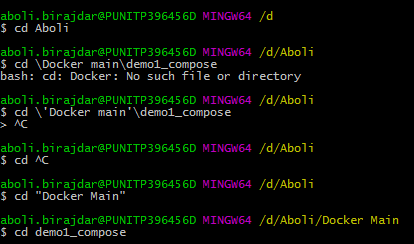
* **git clone**

#git clone “URL of the newly created infy git project “



Creates a GIT repository copy from a remote source. Also adds the original location as a remote so you can fetch from it again and push to it if you have permissions.

* **Navigate to the folder which you wanted to add into the infygit**



|  |  |
| --- | --- |
|  | * **git init**   #git init  C:\Users\aboli.birajdar\Desktop\sc15.PNG |

Initialized empty Git repository in /home/username/GIT/.git/

* **git add**

#git add .

|  |  |
| --- | --- |
| 1 | C:\Users\aboli.birajdar\Desktop\sc17.PNG |
|  | * Adds files changes in your working directory to your index. |

* **git commit**

|  |  |  |
| --- | --- | --- |
| #git commit -m “Folder Name first commit”  committing added changes.  #git commit -a -m  committing all changes, equals to git add and git commit.  C:\Users\aboli.birajdar\Desktop\sc18.PNG  Takes all of the changes written in the index, creates a new commit  object pointing to it and sets the branch to point to that new commit.   * **git push**  |  |  | | --- | --- | |  | #git push origin master |   C:\Users\aboli.birajdar\Desktop\sc19.PNG  Pushes all the modified local objects to the remote repository and advances its branches.   * **Now code will add from your folder to infy git.**   C:\Users\aboli.birajdar\Desktop\gnfH.PNG   * **Other useful git commands -**      * #git rm filename   Removes files from your index and your working directory so they will not be tracked.  #git status  Shows you the status of files in the index versus the working directory. It will list out files that are untracked (only in your working directory), modified (tracked but not yet updated in your index), and staged (added to your index and ready for committing).  #git branch-a \* master remotes/origin/master  Lists existing branches, including remote branches if ‘-a’ is provided. Creates a new branch if a branch name is provided.  #git merge  Merges one or more branches into your current branch and automatically creates a new commit if there are no conflicts.  #git pull  Fetches the files from the remote repository and merges it with your local one.   * Now we can clone the project which is created in infygit to VM using following commands-   #git config –global user.name “user name”  #git config –global user.email “user email @domain”  #git init  #git clone URL of infygit project  C:\Users\aboli.birajdar\Desktop\gitcloning.png |

**Steps to build an docker image using dockerfile which will pull all dependencies including maven and java-**

* Create a directory and add your project folder (clone from your git repository), Dockerfile and maven settings file in that.

C:\Users\aboli.birajdar\Desktop\dezrha.PNG

* docker file should contain following contents:

****

Currently facing some issues with this Dockerfile. Will try to solve it.

***About settings.xml file-***

While running maven project if plugins were unable to pull then we have to add following settings.xml file inside the folder

<?xml version="1.0" encoding="UTF-8"?>

<settings>

<pluginGroups>

<pluginGroup>org.sonarsource.scanner.maven</pluginGroup>

</pluginGroups>

<servers>

<server>

<id>artifactory</id>

<username>admin</username>

<password>password</password>

</server>

<server>

<id>nexus</id>

<username>#username#</username>

<password>#password#</password>

<configuration>

<httpConfiguration>

<all>

<params>

<param>

<name>http.authentication.preemptive</name>

<value>%b,true</value>

</param>

</params>

</all>

</httpConfiguration>

<httpHeaders>

<property>

<name>username</name>

<value>#username#</value>

</property>

</httpHeaders>

</configuration>

</server>

</servers>

<mirrors>

<mirror>

<!--This sends everything else to /public -->

<id>nexus</id>

<mirrorOf>\*</mirrorOf>

<url>http://infynp/repository/maven-public/</url>

</mirror>

</mirrors>

<profiles>

<profile>

<id>nexus</id>

<!--Enable snapshots for the built in central repo to direct -->

<!--all requests to nexus via the mirror -->

<repositories>

<repository>

<id>central</id>

<url>http://central</url>

<releases>

<enabled>true</enabled>

</releases>

<snapshots>

<enabled>true</enabled>

</snapshots>

</repository>

</repositories>

<pluginRepositories>

<pluginRepository>

<id>central</id>

<url>http://central</url>

<releases>

<enabled>true</enabled>

</releases>

<snapshots>

<enabled>true</enabled>

</snapshots>

</pluginRepository>

</pluginRepositories>

</profile>

</profiles>

<profile>

<id>sonar</id>

<activation>

<activeByDefault>true</activeByDefault>

</activation>

<properties>

<!-- Optional URL to server. Default value is http://localhost:9000 -->

<sonar.host.url>

http://localhost:9000

</sonar.host.url>

</properties>

</profile>

<activeProfiles>

<!--make the profile active all the time -->

<activeProfile>nexus</activeProfile>

</activeProfiles>

</settings>

* Now build the docker image using following command-

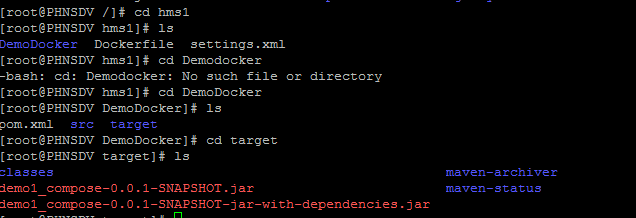
#docker build –t image name .

Command to run docker image-

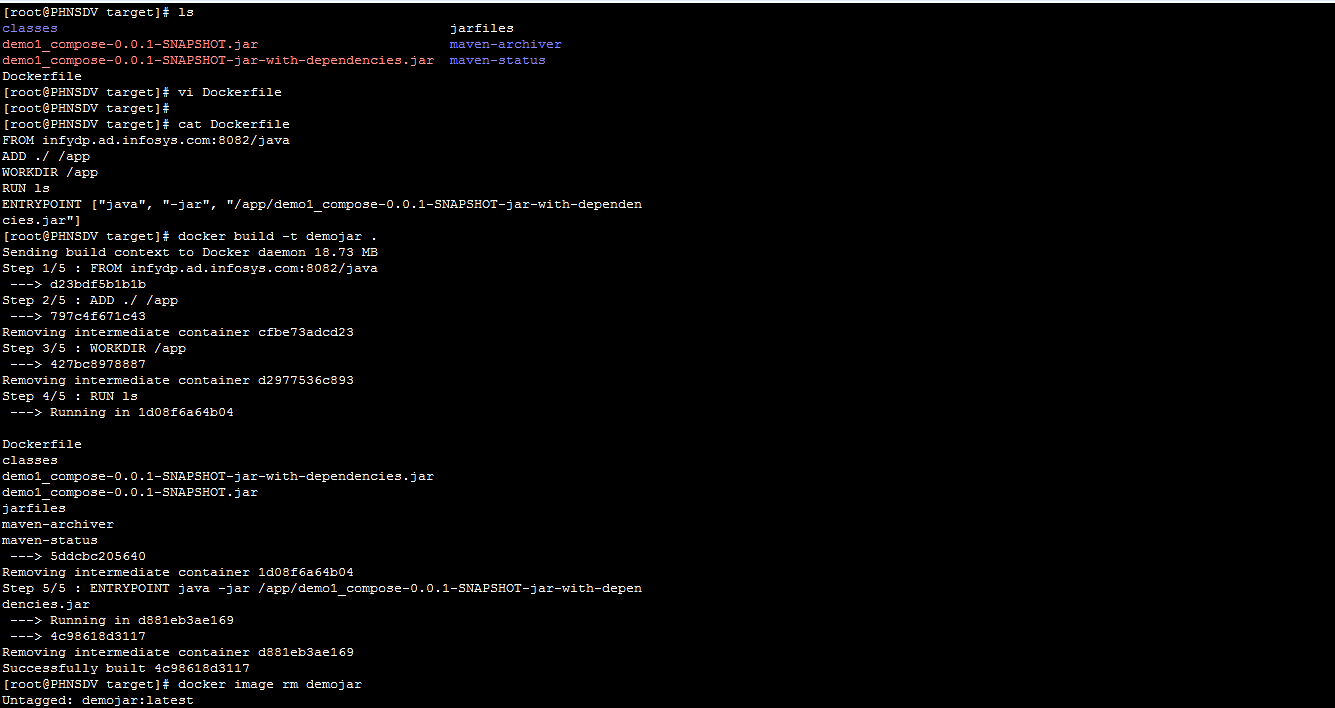
#docker run –name container name image name

***Steps to build an docker image using jar file of project-***

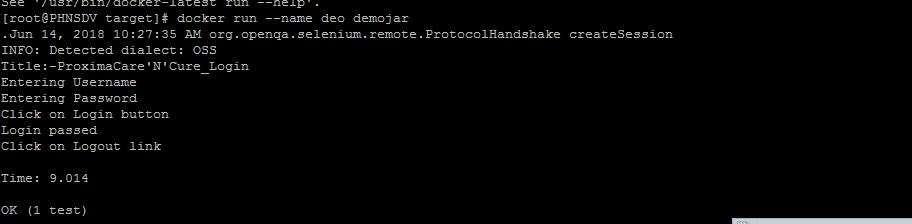
* Navigate to the folder in which jar files are added



* Build a docker image



* Run that Docker image-

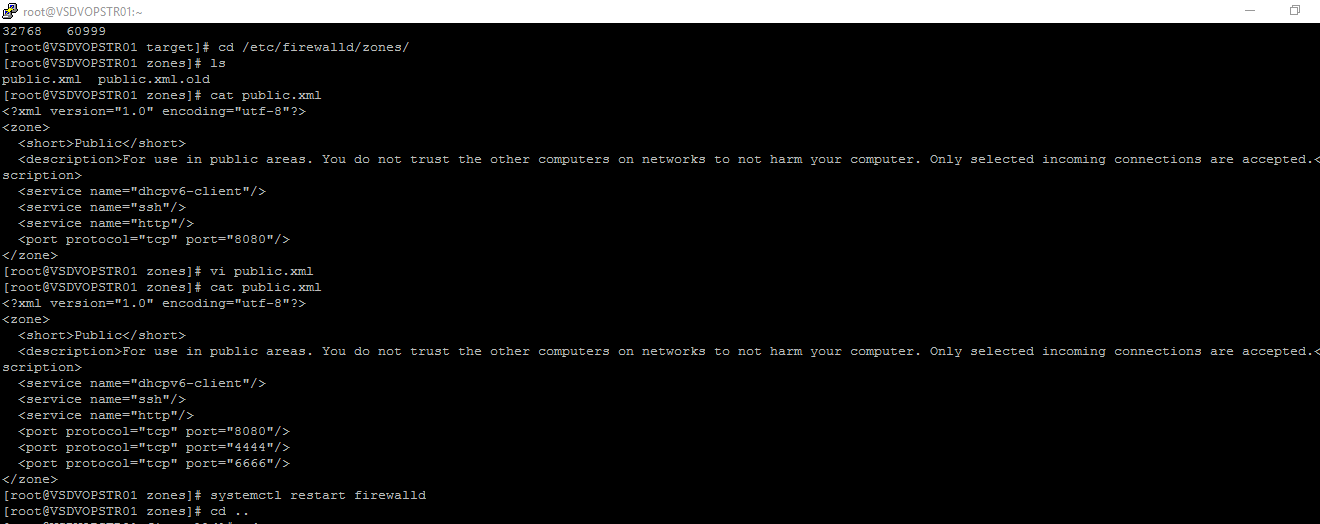


***Unreachable browser exception issue-***

This can be caused while running jar files inside the container because of selenium grid as jar file is trying to access selenium grid hosted in VM.

Steps to follow-

* Navigate to cd /etc/firewalld/zones/
* Navigate to public.xml file and modify it.
* Add the ports on which you are running your selenium grid.



**3. jenkins-**

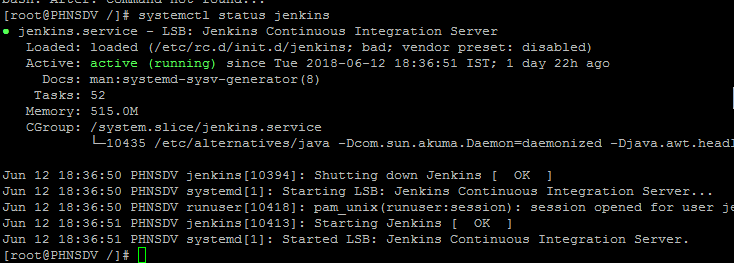
For downloading Jenkins, raise AHD request to download Jenkins in software download section. They will provide path to download Jenkins. After that raise AHD to install Jenkins. Give path provided by CCD.

Command to check status of Jenkins

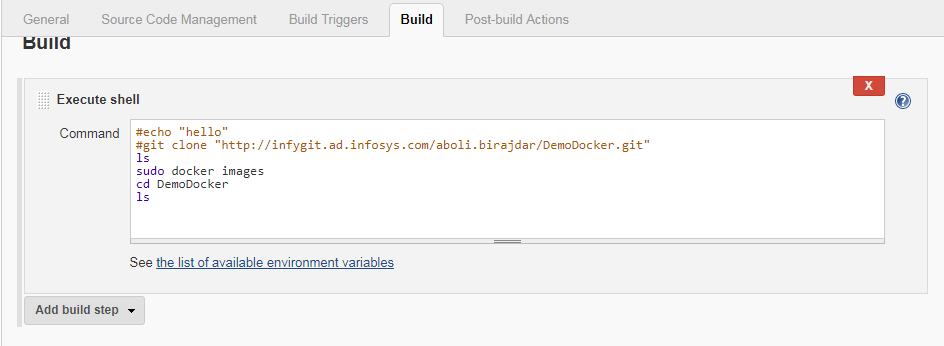
#systemctl status jenkins

Command to start jenkins

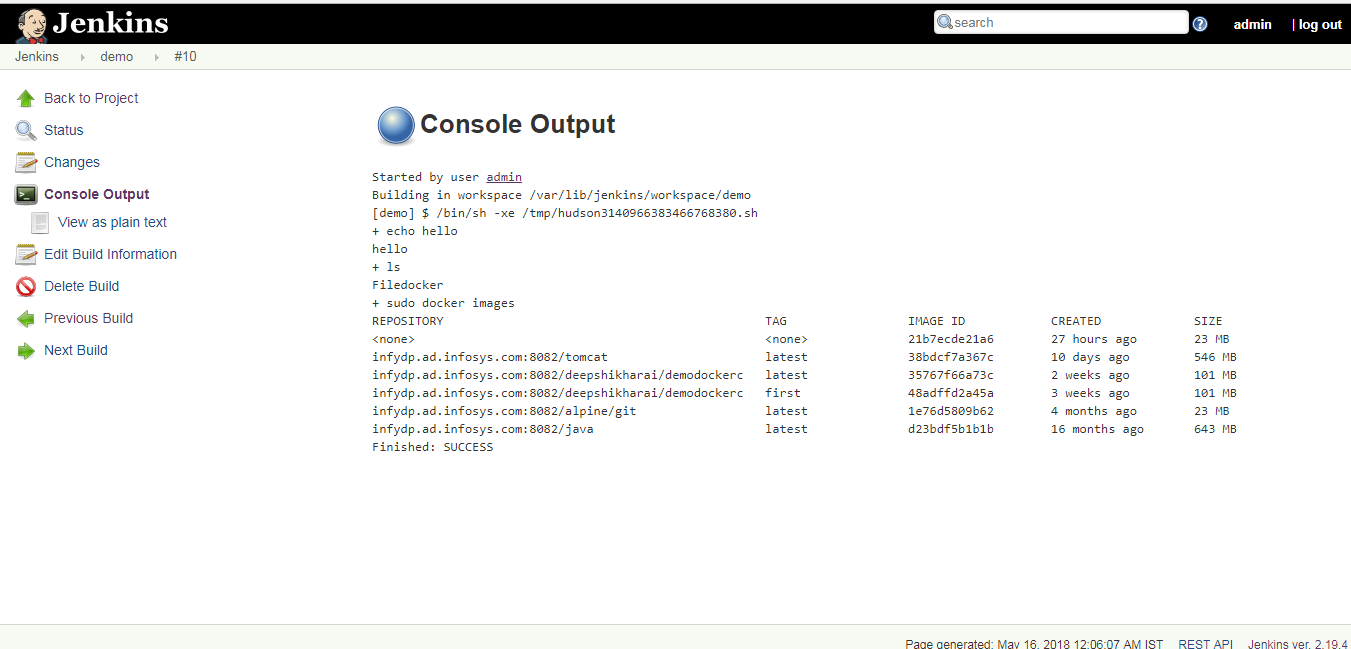
#systemctl start jenkins

****

* Create a project in jenkins and clone the infygit project inside that.



* build the project

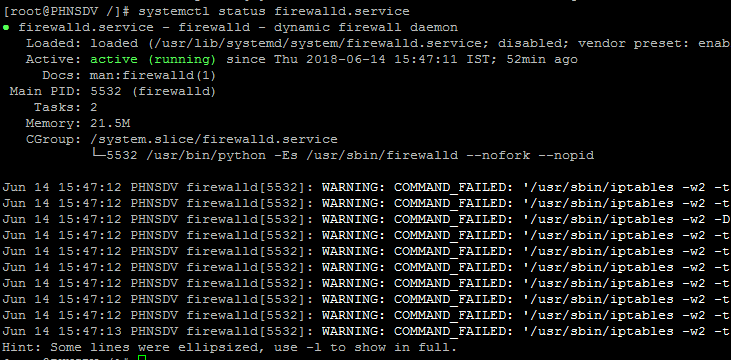


***Firewall issue in jenkins-***

If jenkins is running in VM and it is not accessible outside (in browser).

Check the status of firewall, it should be in active state

systemctl status firewalld.service



Command to start firewall

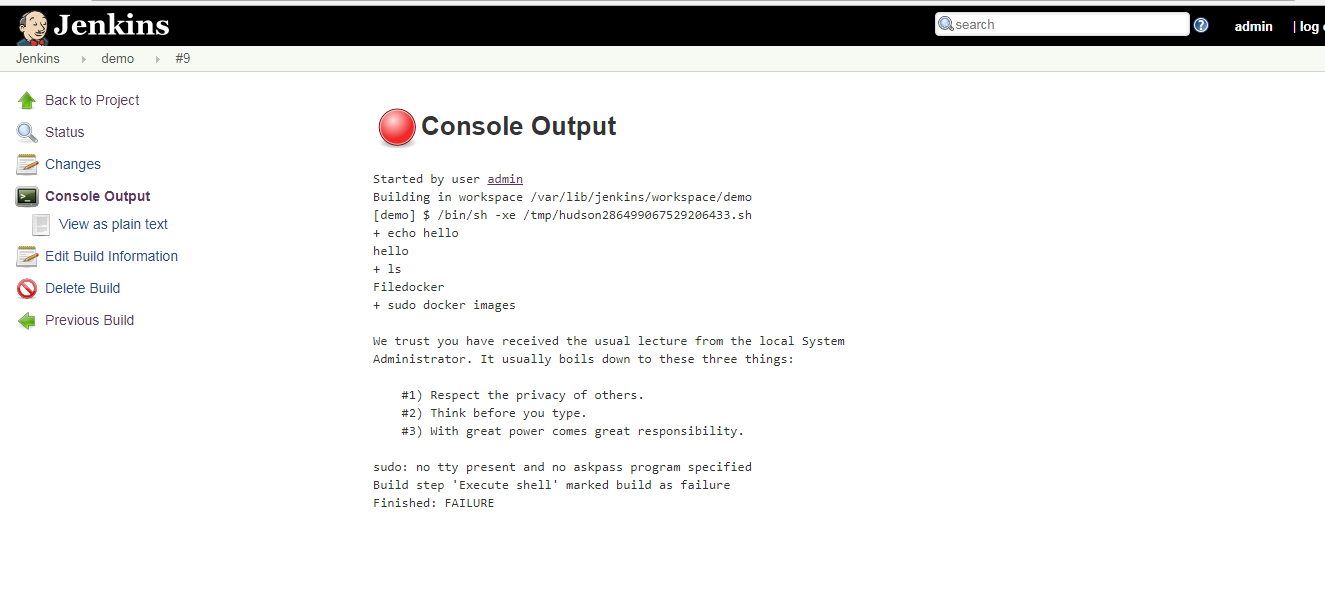
systemctl start firewalld.service

Command to stop firewall

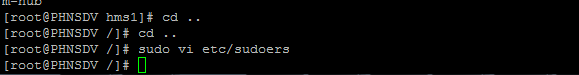
systemctl stop firewalld.service

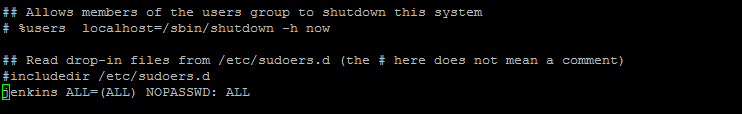
***For no tty present error in Jenkins-***

Steps to resolve No tty present error while running docker command in jenkins-



* Go out from all the directories in VM.
* Execute below command.
* sudo vi etc/sudoers
* Add this line in file :  “Jenkins ALL=(ALL) NOPASSWD: ALL”





* If you are getting error after adding this line as well add below line before the “Jenkins ALL=(ALL) NOPASSWD: ALL” and try to run.

%sudo ALL=(ALL:ALL) ALL