**Configuration of SSH and Installation of Jenkins**

Previous steps:

* As we stand up our virtual machines with Vagrant, we can have access and login automatically using the command vagrant ssh inside the folder where our Vagrantfile is located.
* To login as root user we use the next command sudo su.

1. Ubuntu 20.4 LTS:

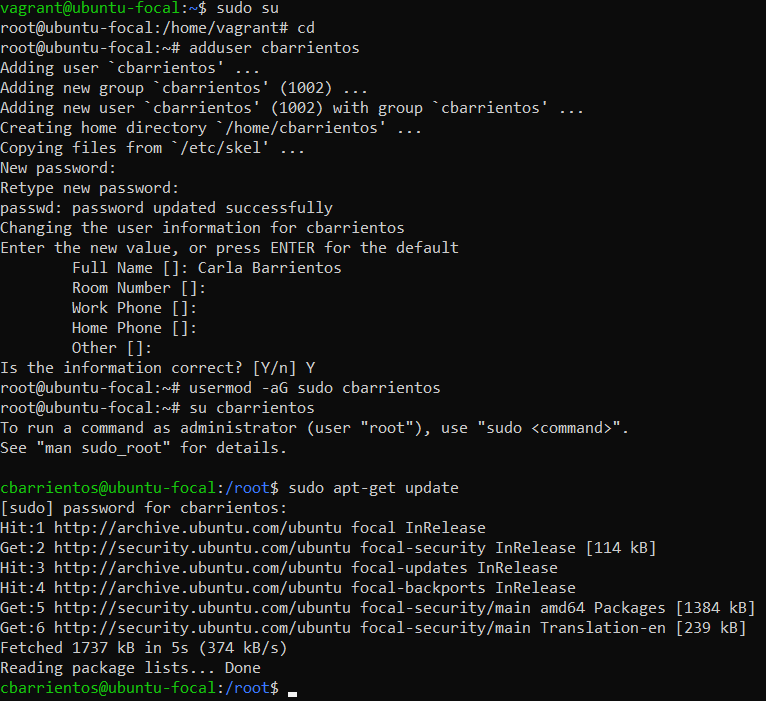
*User creation*

First we start creating a new user with sudo privileges, to achieve this we run the next commands as root user:

adduser <username>

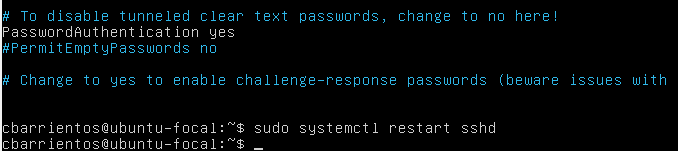
usermod -aG sudo <username>

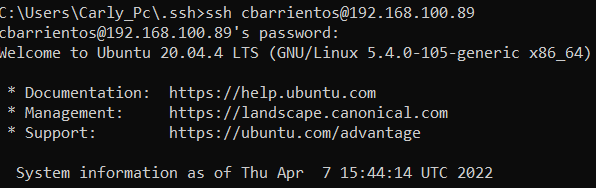
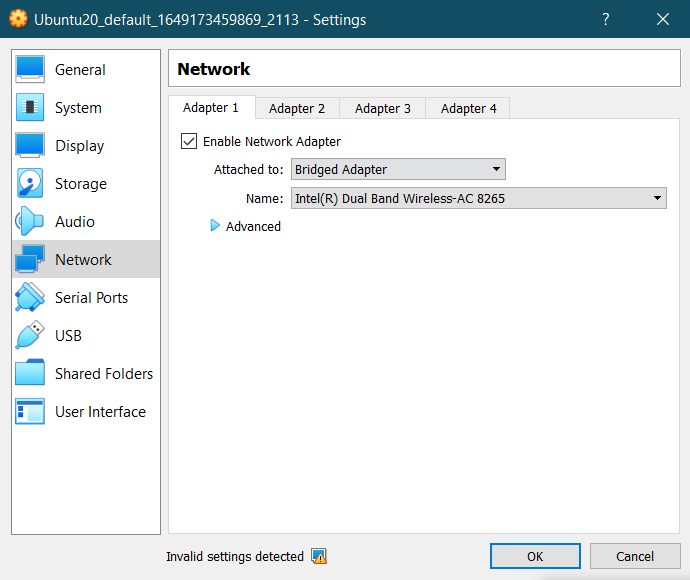
The result of running both commands and testing that the new created user has sudo privileges is shown in the next image:



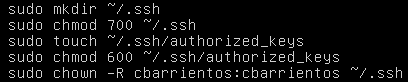
*SSH Configuration*

To check that the ssh service is running properly, first we need to enable tunneled clear text passwords. We run the next command sudo nano /etc/ssh/sshd\_config, change the value of PasswordAuthentication to yes and restart the ssh service with sudo systemctl restart sshd.



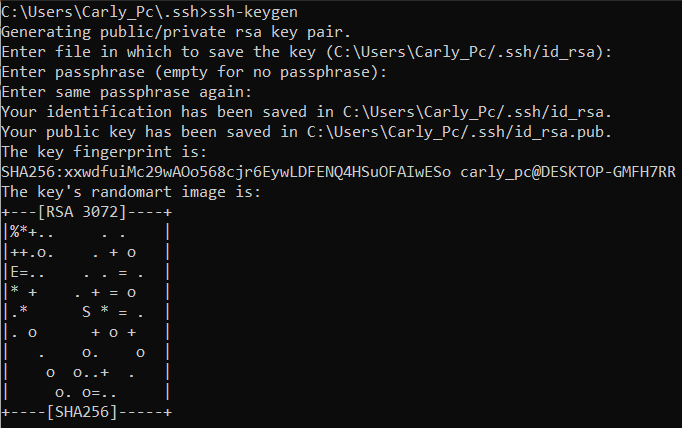
Then we have to change the network adapter of the virtual machine to Bridged so that we can connect via ssh from the host machine using the ip address given by the network.

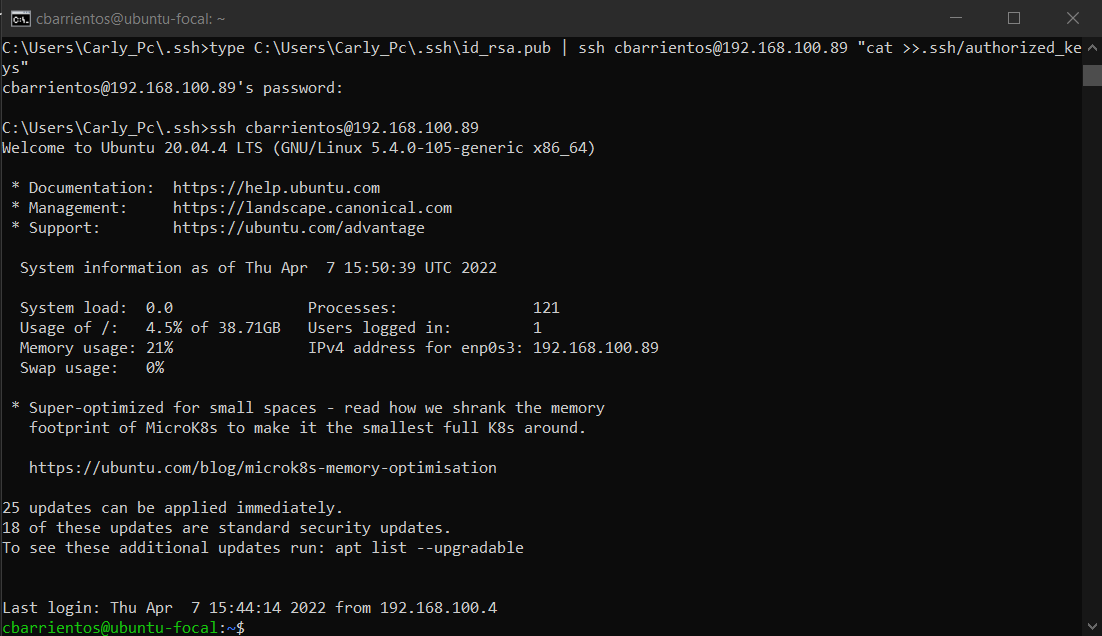
If everything is working properly, the next step is to start configuring ssh keys for the created user, is create the .ssh folder that is going to contain the authorized keys. To achieve this goal we need to run the next commands:



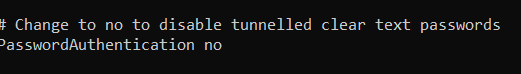
The first line creates the folder, next we add permissions to protect the folder from other users than the current user. On the third line we create the file authorized\_keys and next we change its permission to a private file that can only be changed by the user who entered the command. Finally we change the owner and the group of the folder and its content to our user.

On the host machine that is going to be the client we need to generate the rsa key pair. To do so we execute the command ssh-keygen:



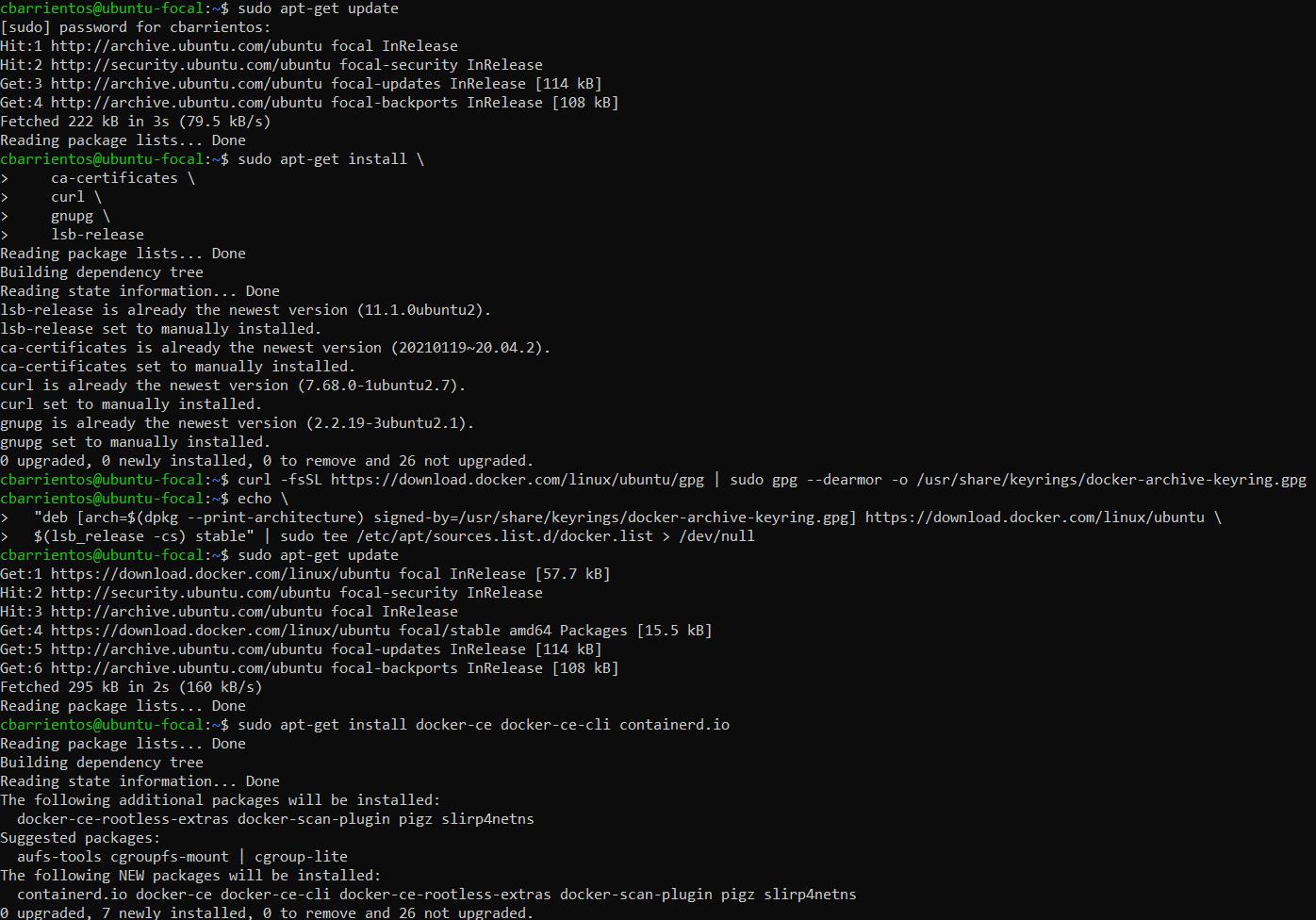
Then we need to copy the client public key to the ssh server authorized\_keys file running the first command shown in the next picture:

We have to change the PasswordAuthentication of the /etc/ssh/shhd\_conf to no again so we can connect without requiring any password, the result is shown in the previous screenshot.

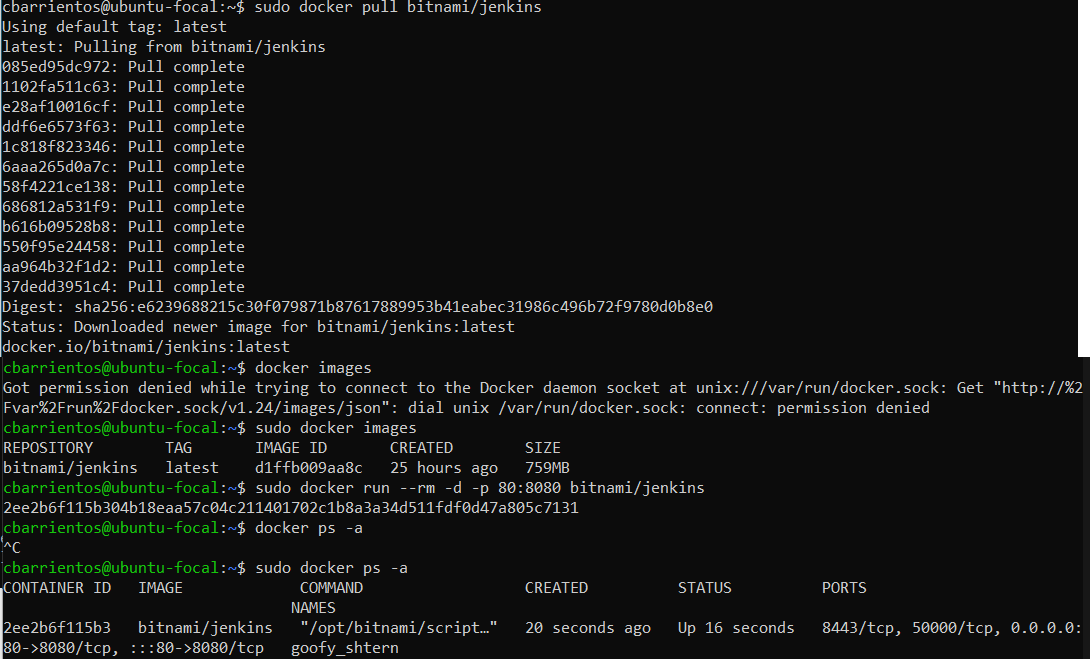


*Docker + Jenkins installation*

To install Docker we only need to follow the official documentation, first we need to update the apt package index and add Docker’s official GPG key, then we set up the stable repository, and finally we need to install Docker engine and containerd.



After that we can pull the Jenkins image of our preference and then start the container with the commands shown in the next image:



1. Fedora 35:

*User creation*

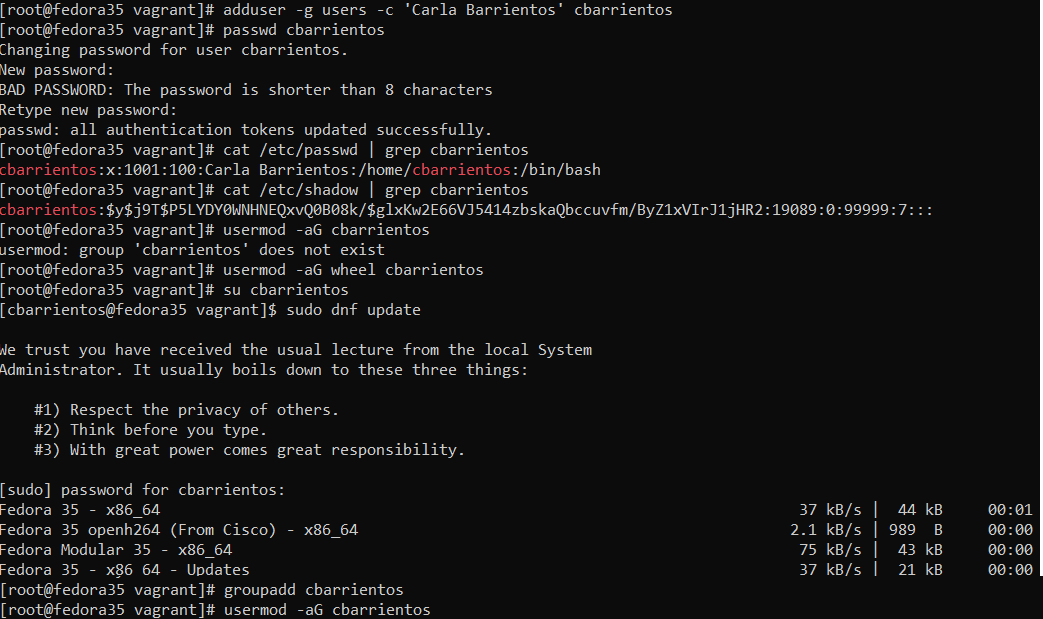
We start creating a new user with sudo privileges, as root user we run the next commands:

adduser -g users -c ‘<Complete name>’ <username>

passwd <username>

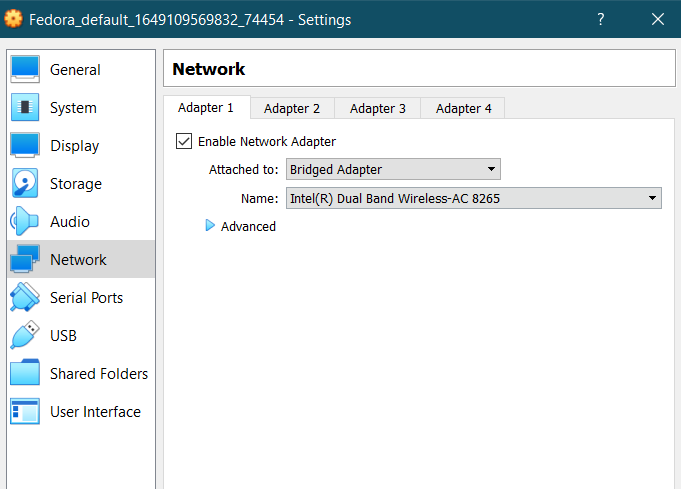
usermod -aG wheel <username>

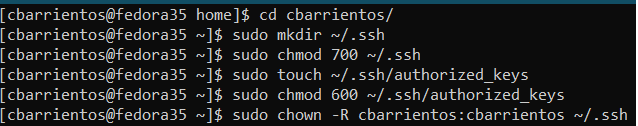
In Fedora distribution is necessary to add passwords for new users with the seconf command, and the third command is to give the user sudo privileges. The result of running these commands and testing that the new created user has sudo privileges is shown in the next image:



*SSH Configuration*

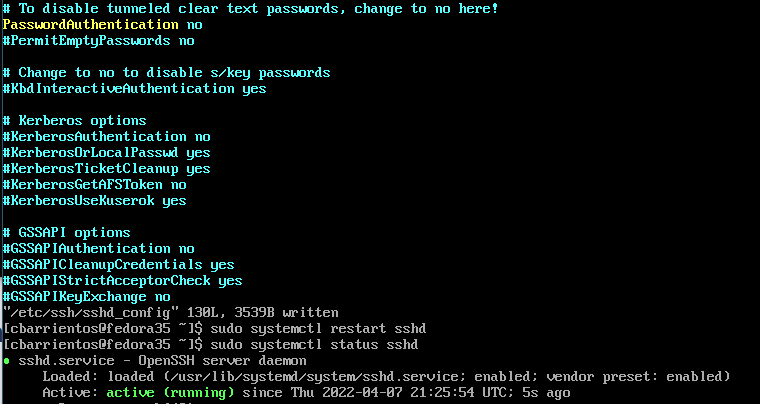
To check that the ssh service is running properly, first we need to enable tunneled clear text passwords, in this distribution it is enabled by default. If it was not we run the next command sudo nano /etc/ssh/sshd\_config, change the value of PasswordAuthentication to yes and restart the ssh service with sudo systemctl restart sshd.

Then we have to change the network adapter of the virtual machine to Bridged so that we can connect via ssh from the host machine using the ip address given by the network.

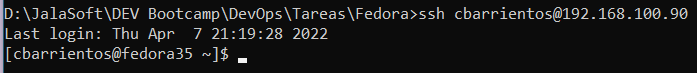
As we did in Ubuntu distribution, the next step to start configuring ssh keys for the created user is create the .ssh folder that is going to contain the authorized keys running these commands:

As we already have our rsa key pair generated in the client machine, we only need to copy the client public key to the ssh server authorized\_keys file running the same command used for Ubuntu distribution.

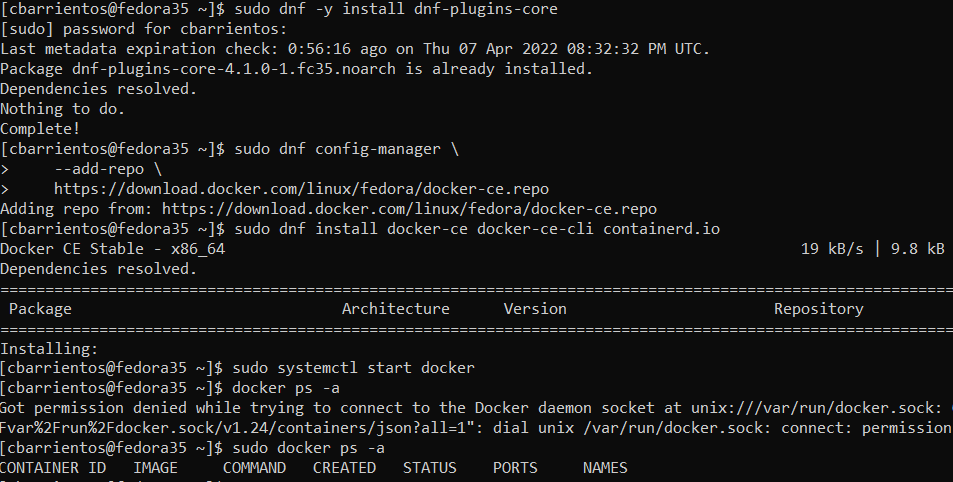
Last step we have to change the PasswordAuthentication of the /etc/ssh/shhd\_conf to no so we can connect without requiring any password, then we need to restart the service:



The result:



*Docker + Jenkins installation*

To install Docker we only need to follow the official documentation, first we need to install the dnf-plugins-core package and set up the stable repository, then we install the latest version of Docker engineand containerd, finally we need to start the Docker service:

After that, we can pull the Jenkins image of our preference and then start the container with the commands shown in the next image:

