

S.I.G.D.

SISTEMA OPERATIVOS III

GFORZE

ROL	APELLIDO	NOMBRE	C.I	CORREO	TELÉFONO
Coordinador	Mujica	Martin	5.523.189-7	mujica0106@gmail.com	095003085
Subcoordinador	Farías	Facundo	5.332.076-0	j.facuwebmaster@gmail.com	093448935
Integrante 1	González	Sheila	6.392.488-5	sheilagn2003@gmail.com	096247975
Integrante 2	Vallejos	Diego	5.493.546-5	micezedie@gmail.com	097302419

Docente: Fagúndez, Luis

Fecha de culminación
19/07/2022

PRIMERA ENTREGA

Índice

Estudio de los diferentes roles de los usuarios del sistema.	4
Usuarios Cliente	4
Usuarios Base de Datos	4
Usuarios Servidor	4
Servidor de Respaldo	4
Relevamiento y justificación del Sistema Operativo a utilizar tanto en las terminales de los usuarios como en el servidor.	5
Manual de instalación del Sistema Operativo en el servidor.	8
Paso 1:	8
Paso 2:	9
Paso 3:	10
Paso 4:	11
Paso 5:	12
Paso 6:	13
Paso 6:	14
Paso 7:	16
Paso 8:	16
Instalación de Docker dentro de la máquina virtual con el sistema operativo a utilizar. (WEB)	20
Paso 1:	20
Paso 2:	20
Paso 3:	21
Paso 4:	21
Paso 5:	22
Instalación de un servidor LAMP en la máquina virtual (usando docker o local).(web)	23
LOCAL	23
PASO 0:	23
PASO 1:	23
PASO 2:	24
PASO 3:	25
PASO 4:	28
PASO 5:	28
Manual de Instalación de MySQL/MariaDB junto a sus requisitos para trabajar con Docker.	29
Paso 0:	29
Paso 1:	29
Paso 2:	29
I.S.B.O.	3°BA
GFORZE	2



Paso 3:	30
Paso 4:	30
Paso 5:	30
Paso 6:	31



Estudio de los diferentes roles de los usuarios del sistema.

Usuarios Cliente

Invitado: Ingresando la C.I. puede cargar(mostrar) un perfil desde cualquier terminal.

Administrador del Sitio: Tiene todos los permisos.

Árbitro: Incidencias. Verifica si los jugadores están habilitados. Registra la veracidad de las incidencias registradas en el encuentro (Anotaciones, Sanciones, Faltas).

DT(Director Tecnico): Genera equipos. Carga la ficha de los Jugadores previamente cargados por el Administrativo.

Administrativo: Carga las fichas. Asigna los roles del Sitio(le otorga los permisos a cada usuario).

Analista: Verifica y sube los datos completos de los partidos.

Usuarios Base de Datos

Root

Test: Crear usuarios para verificar que funcione correctamente.

Usuarios Servidor

Root(Default)

Administrador de Respaldo: Genera un respaldo de todos los datos subidos.

Administrador de Paquetes: Mantiene actualizado el servidor.

Administrador de Base de Datos: Chequea los datos de la base de datos.

Administrador de Logs: Chequea si hay error en el Servidor.

Todos estos usuarios van a estar en /etc/sudoers.

Servidor de Respaldo

Administrador de Respaldo

Root

Relevamiento y justificación del Sistema Operativo a utilizar tanto en las terminales de los usuarios como en el servidor.

Nuestra empresa GFORZE, una empresa de desarrollo de software, realizó una puesta en común con el equipo de desarrollo, sobre la utilización de un sistema operativo que se adapte a las necesidades de un servidor. En base a los requerimientos que nos proporciona el cliente en base al estudio de la especificación de requerimientos llegamos a la conclusión de usar Fedora Server porque se adapta a nuestras necesidades. A continuación se detalla un punteo de los beneficios de aplicar Fedora Server al software:

- Gratuito y flexible.
- No requiere muchas especificaciones para su funcionamiento.
- Es seguro, ya que es robusto y evita el ingreso de intrusos, aplica SELinux(Security-Enhanced-Linux) que es un módulo de seguridad en el linux kernel.
- Es un software libre, dicho de otro modo, código abierto.
- Ofrece un soporte constante.
- Administrar de forma favorable la memoria.
- Arranque del sistema de forma completa y sin errores

Terminal:

Para comprar las terminales vamos a dejarles 2 presupuestos diferentes, uno de bajos costos y uno de costos medios.

Les ofrecemos estos 2 presupuestos ya que básicamente la terminal lo único que va a hacer será abrir google o su navegador preferido para inicializar la pagina, lo unico que cambiaria seria el SO que prefiera utilizar para ejecutar dicha operación.

Primer presupuesto:

De la empresa CDR Medios hemos visualizado distintas computadoras y hemos elegido para este presupuesto de precio bajo:

Equipo HP AMD 3.0Ghz, 4GB, 160GB, DVD que valdría 45 U\$D, esta computadora tendría que utilizar un SO de linux ya que los requerimientos no nos permite utilizar WINDOW.



Gabinete HP Compaq 6005 Pro SFF
Sistema operativo: COA Windows Vista
Procesador AMD Phenom II X3 B75 3.00 Ghz
Disco Duro: 160 GB
Memoria 4GB DDR3
Lectora de DVD.
Fuente 240W

Código	EQU896
Garantía	6 meses contra defecto de fabricación

Segundo presupuesto:

De la empresa CDR Medios hemos visualizado distintas computadoras y hemos elegido para este presupuesto de precio bajo:

Equipo HP Core i5 3.3Ghz, 4GB, 250GB que valdria 125 U\$D, esta computadora puede utilizar cualquier SO hasta Window 10 Home.



Modelo HP 8200 Elite SFF.
 Processor Intel Core i5 2400 3.3Ghz. (2a generacion)
 Memoria RAM 4GB DDR3.
 Disco duro 250 GB.
 Lectora de DVD (sujeto a stock)

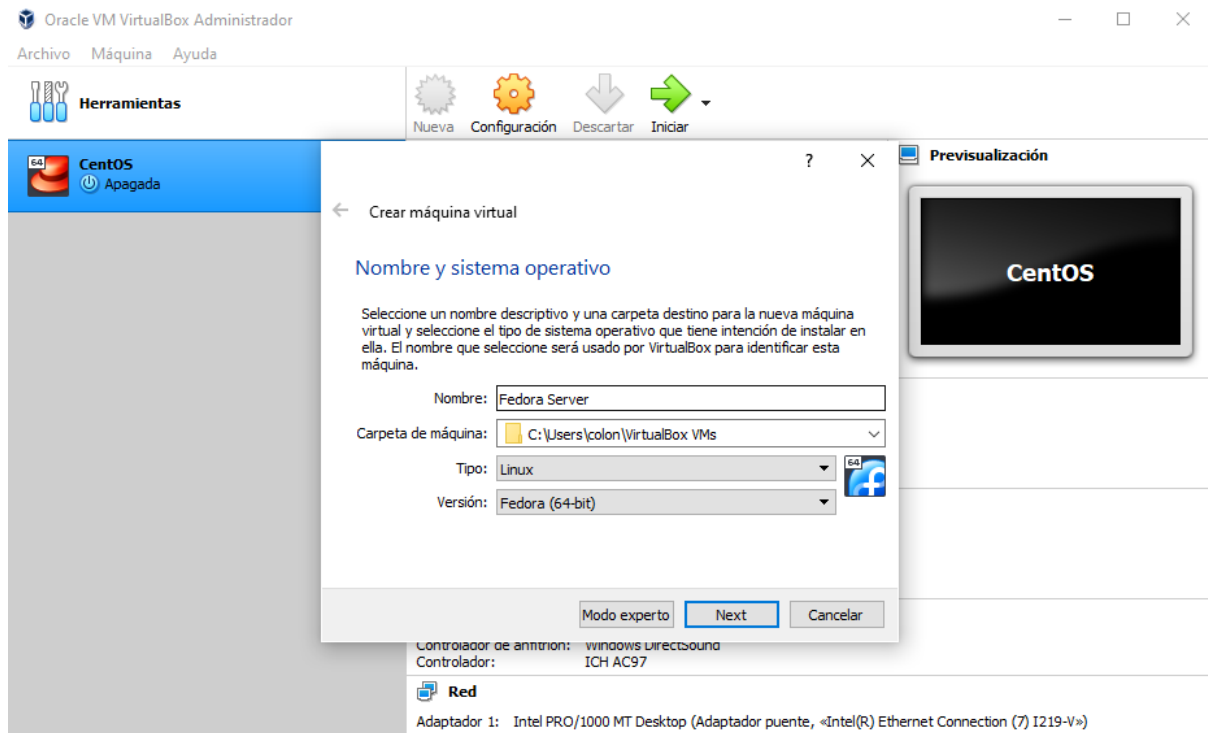
Código EQU774
 Garantía 6 meses contra defecto de fabricación

Ambos presupuestos tendrán el soporte técnico proporcionado por CDRMEDIOS (6 meses contra defectos de fabricación), además que tendrán 1 año de soporte técnico proporcionado por la empresa GFORZE(Mantenimiento, o reparaciones de software).

Manual de instalación del Sistema Operativo en el servidor.

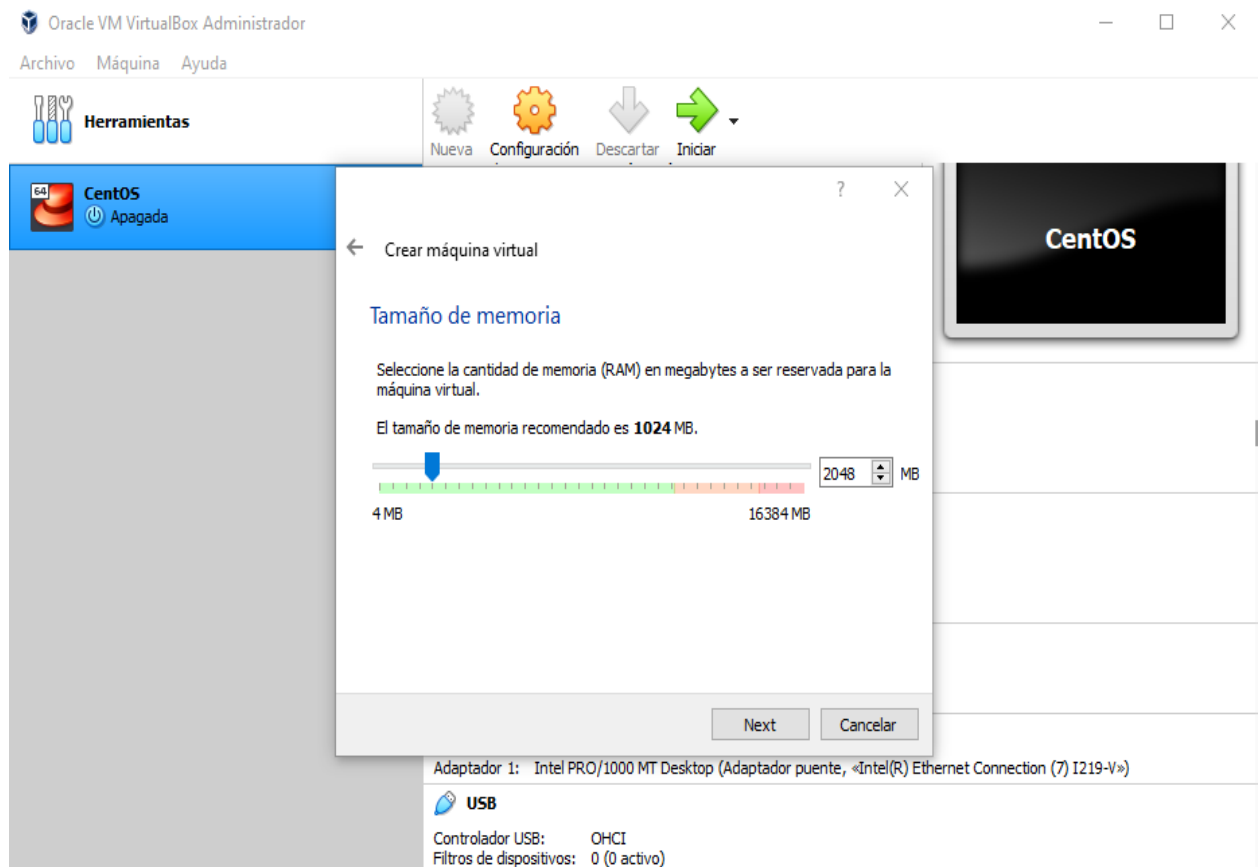
Paso 1:

Primero le debemos dar a Máquina/Nuevo, luego tendremos que escribir Fedora Server y el programa automáticamente te pondrá los datos de Tipo y Versión. Luego le daremos a next y pasaremos al siguiente paso.



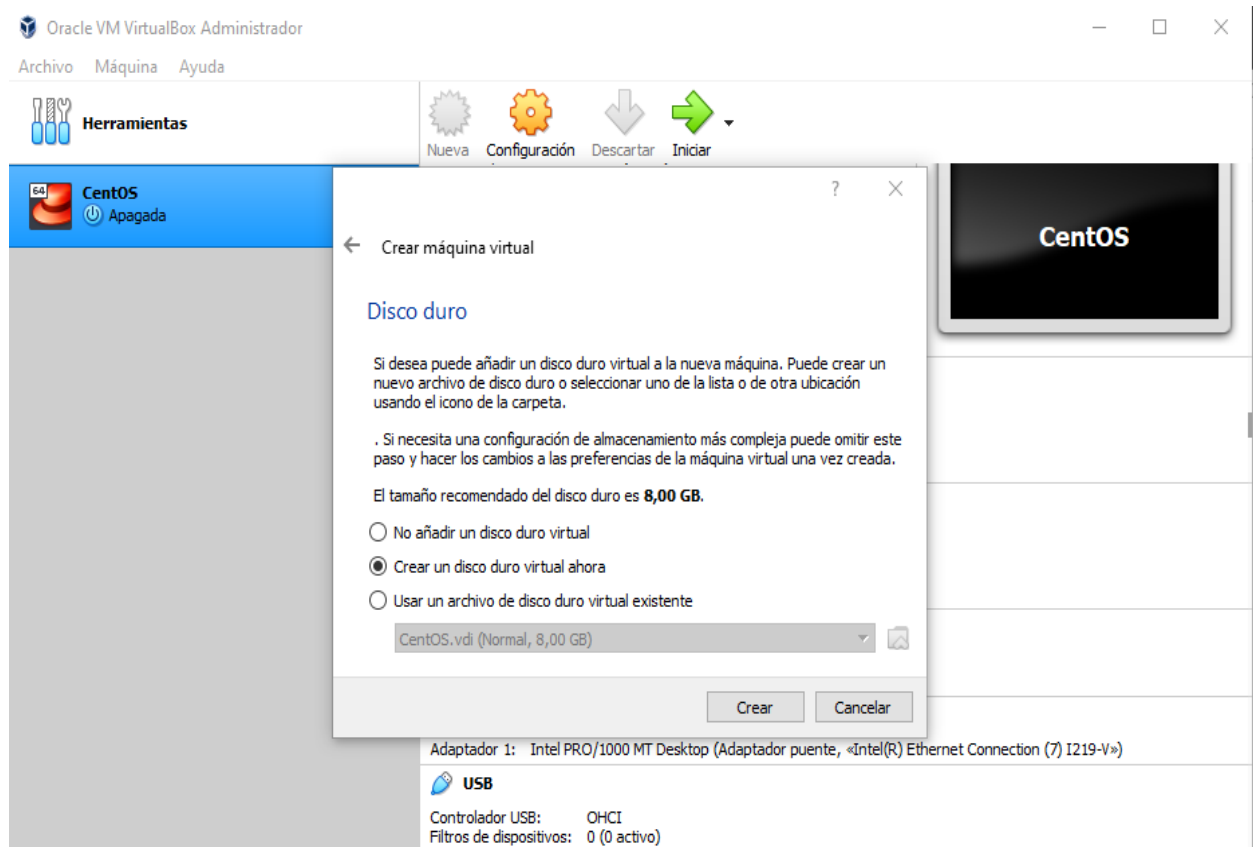
Paso 2:

Luego de darle a next nos aparecerá esta opción. Aquí tendremos que seleccionar el tamaño de la memoria ram que virtualbox utiliza para reproducir nuestro Fedora Server. En este caso pondremos 2GB de memoria RAM pero si quieres le puedes poner más, dependiendo de cuanta memoria ram tengas disponible. Por último le daremos next y procederemos el siguiente paso



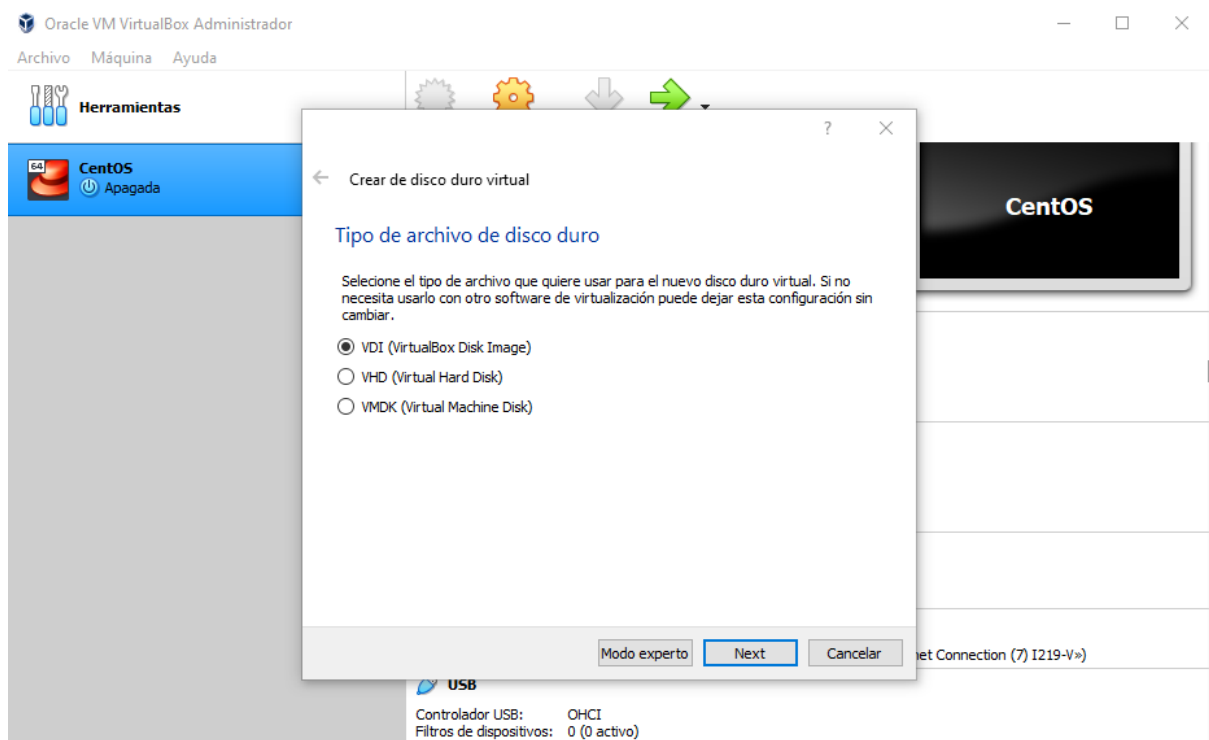
Paso 3:

En este paso debemos de seleccionar un disco duro para que la máquina virtual pueda utilizar, en este caso si o si debemos de poner la opción “Crea un disco duro virtual ahora”. Por último le daremos a next.



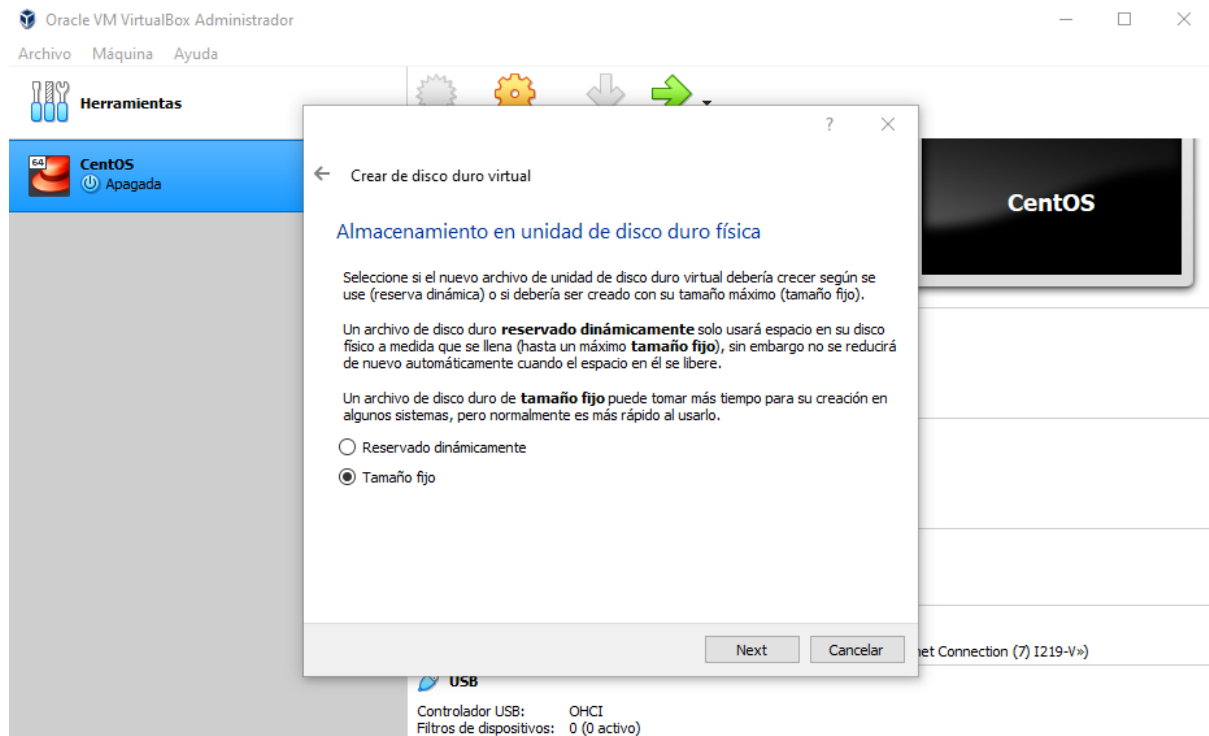
Paso 4:

Aquí seleccionaremos el tipo de disco duro virtual, este básicamente seleccionamos VDI ya que nosotros tenemos un archivo .iso para instalar la máquina virtual.



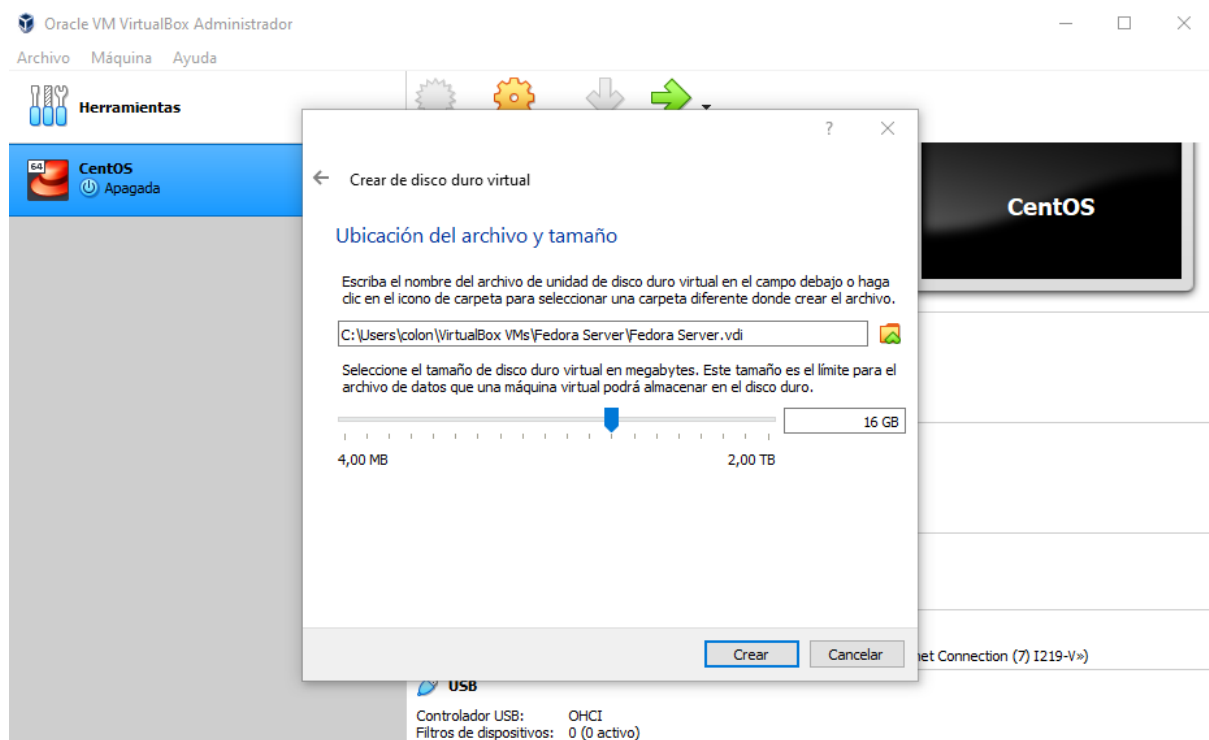
Paso 5:

Aquí seleccionamos tamaño fijo para que si al superar la cantidad de contenido no nos abarca más espacio del definido anteriormente, si tienes espacio de sobra igual puede ponerlo reservado dinámicamente pero eso es a tu elección.



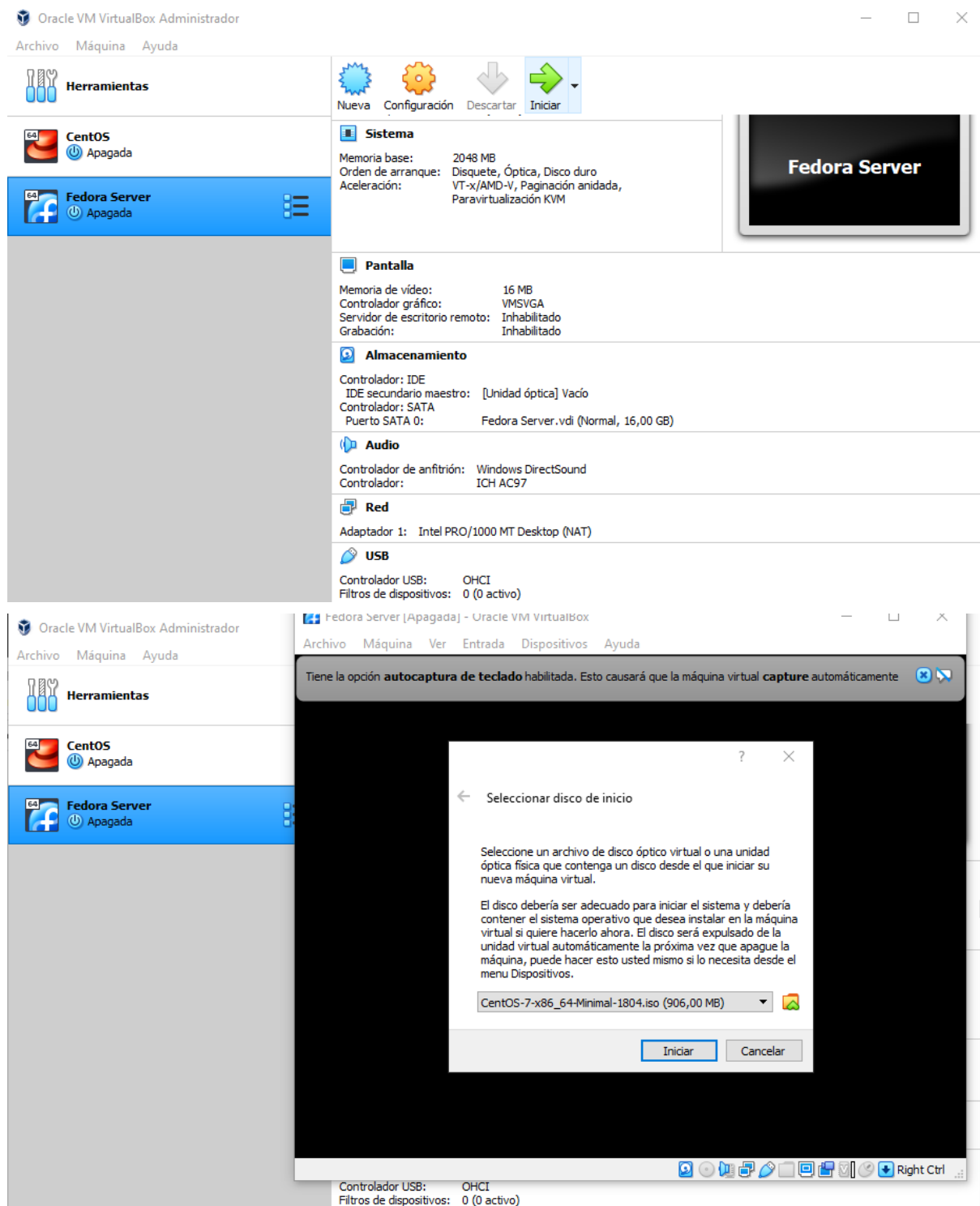
Paso 6:

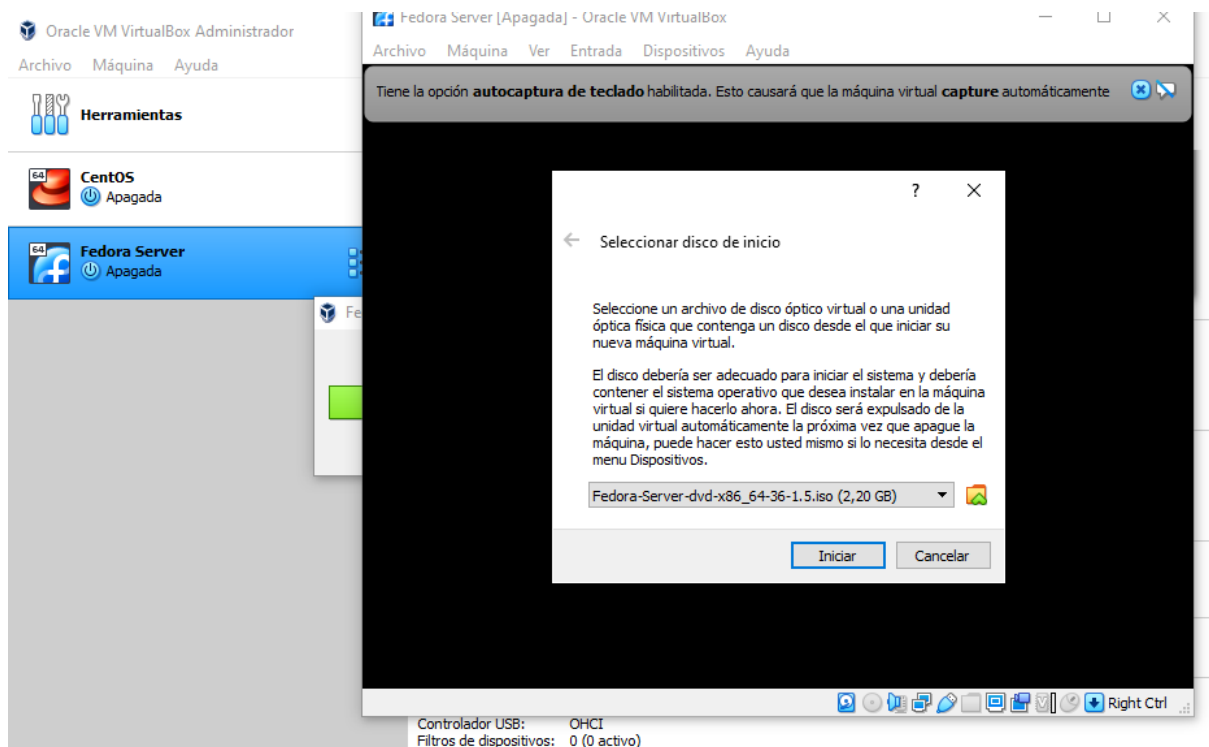
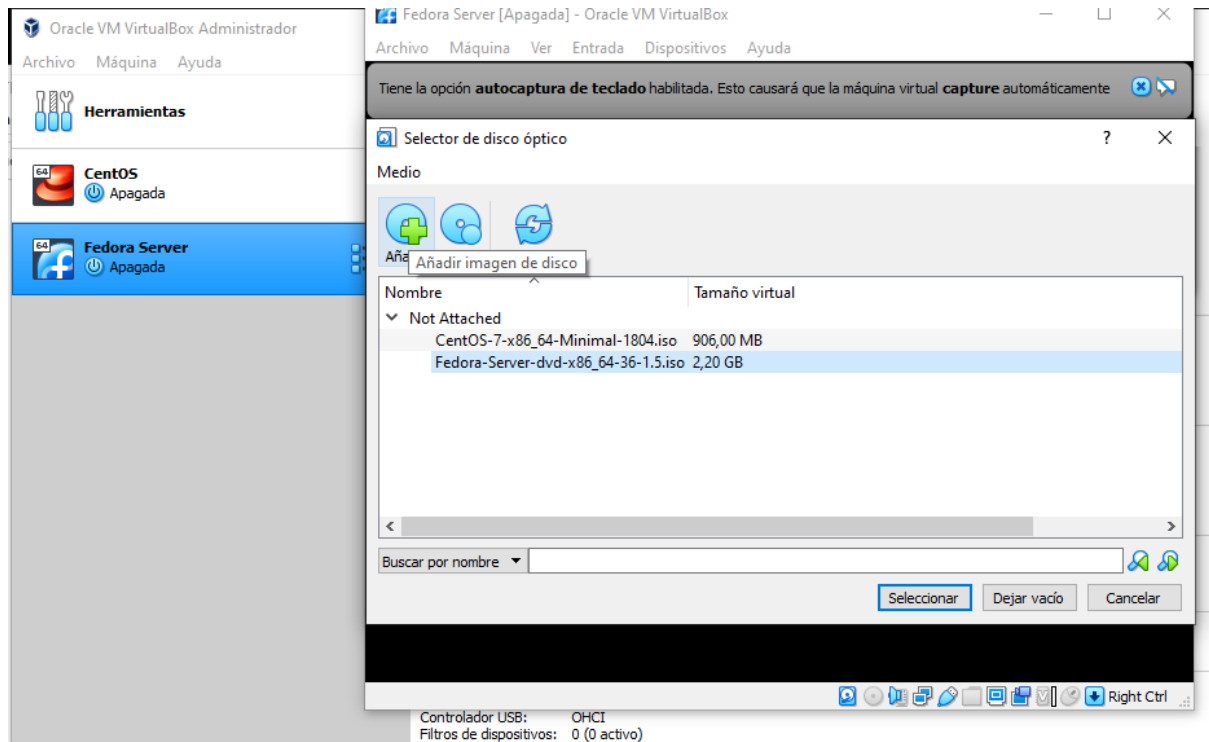
Aquí seleccionamos la cantidad de espacio que queremos que tenga Fedora Server para utilizar el mínimo es 8GB pero a su gusto pueden agregar más.



Paso 6:

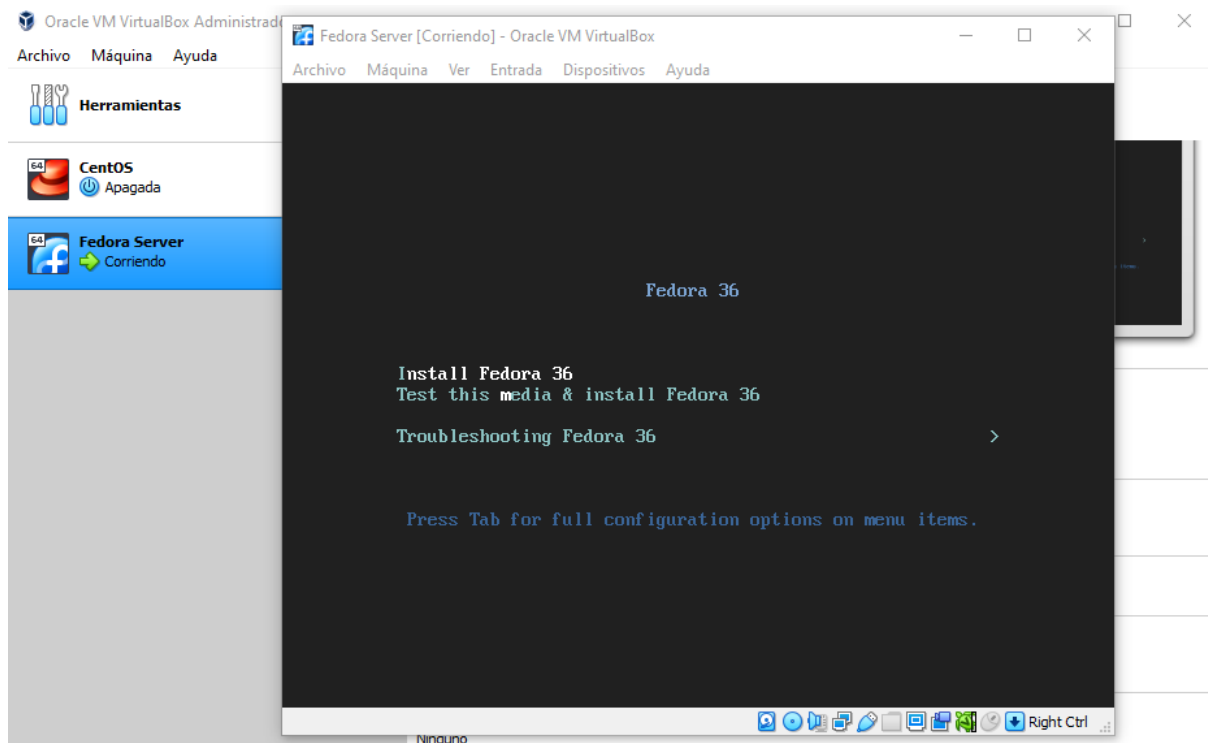
En este paso le daremos a iniciar Fedora Server seleccionaremos el archivo .iso de Fedora Server y le daremos a iniciar.





Paso 7:

Aquí con el teclado (las flechas) seleccionamos la opción install fedora.



Paso 8:

Aquí pasaremos a la instalación de Fedora Server, Seleccionamos idioma, el destino de instalación (Básicamente seleccionar el disco duro), seleccionar el usuario root, activar el usuario y poner una contraseña.

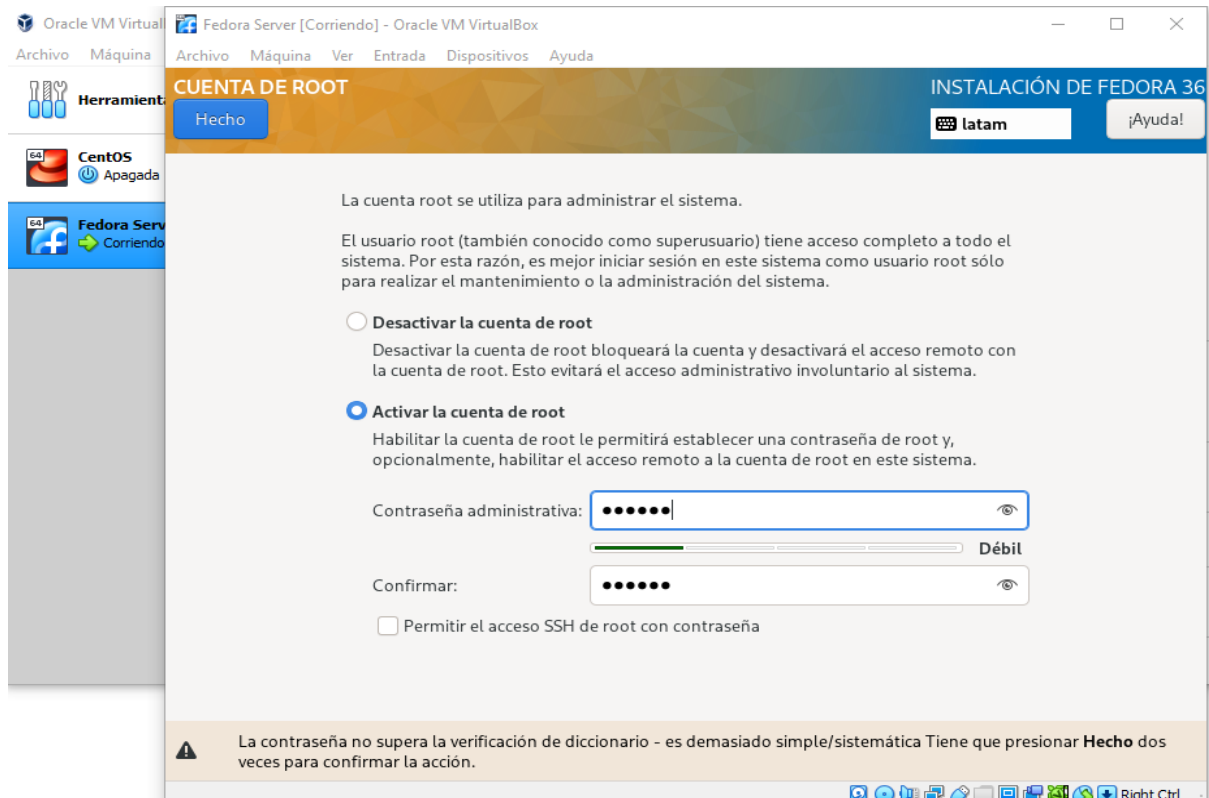
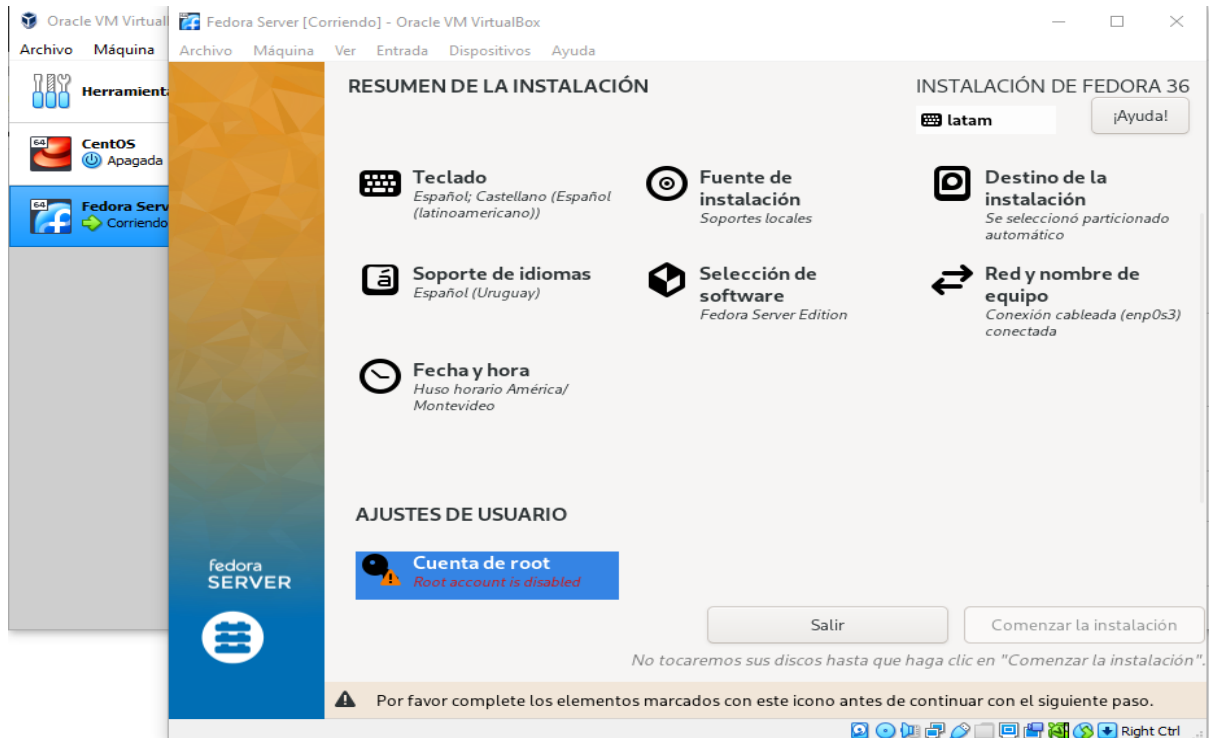
Por último seleccionamos el usuario y llenamos las opción(Nombre y contraseña), por ultimo le damos a COMENZAR INSTALACIÓN y esperamos a que nos instale Fedora Server.

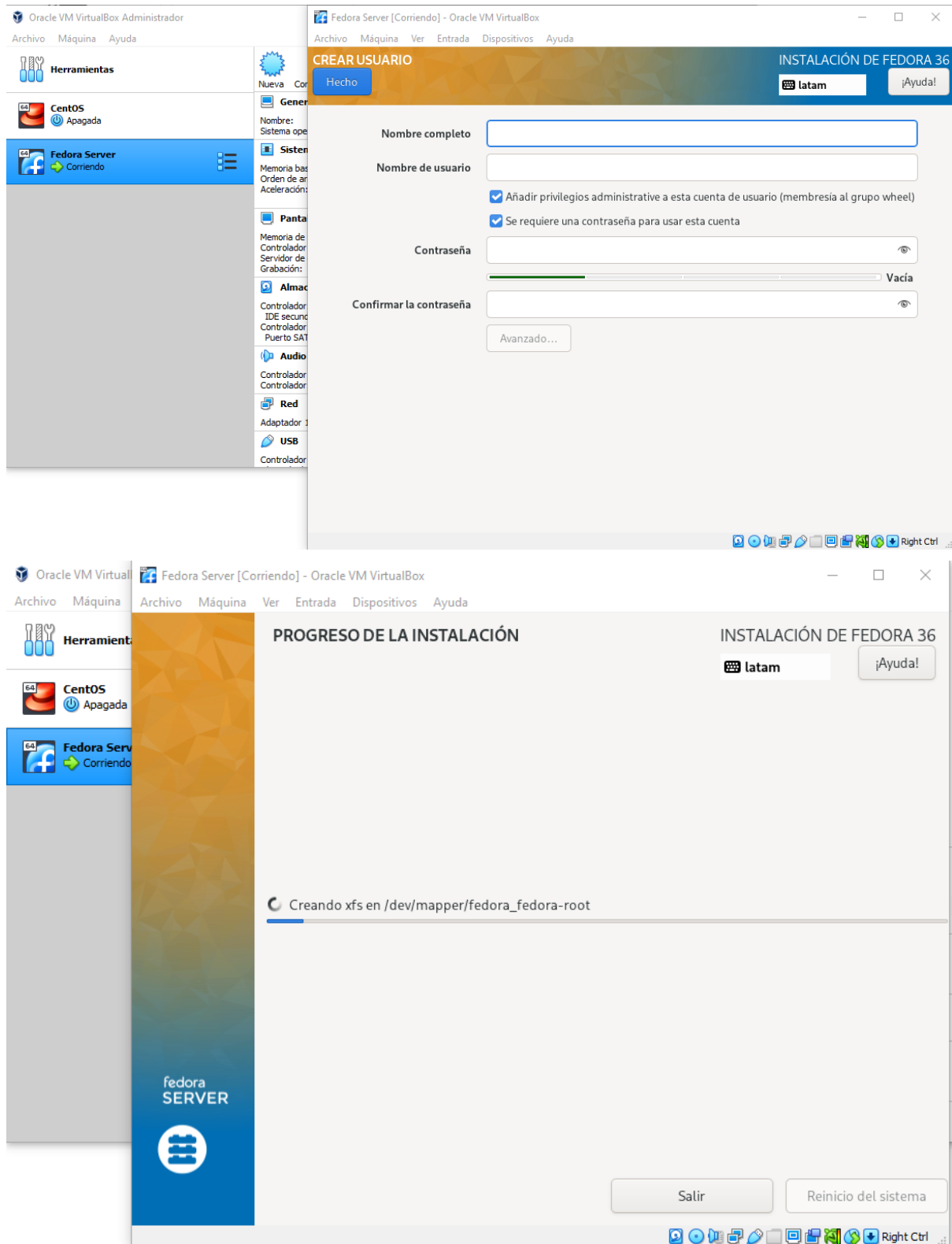


I.S.B.O.

GFORZE

3ºBA



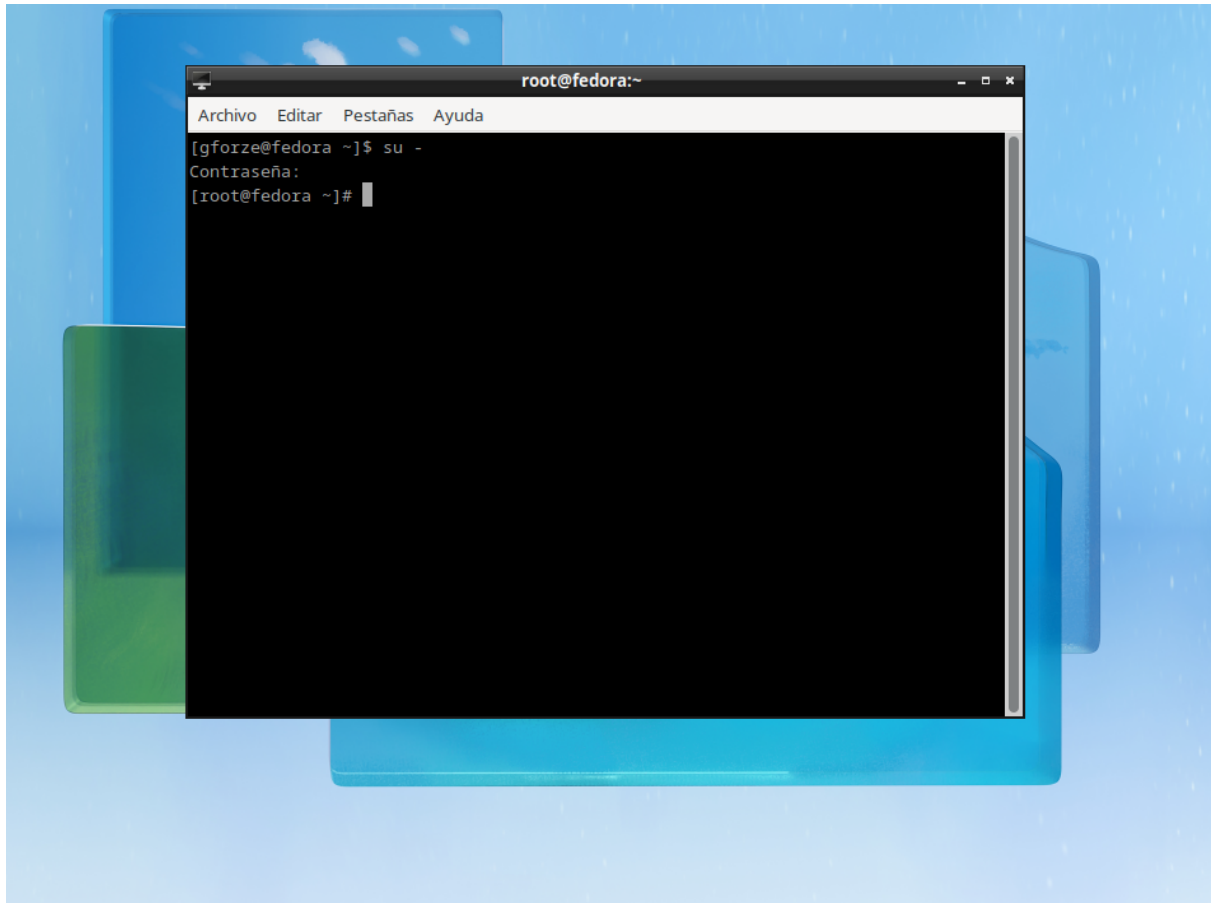


Configuración IP Fija

Paso 1:

Principalmente iniciaremos con nuestro usuario root utilizando en la consola el comando:

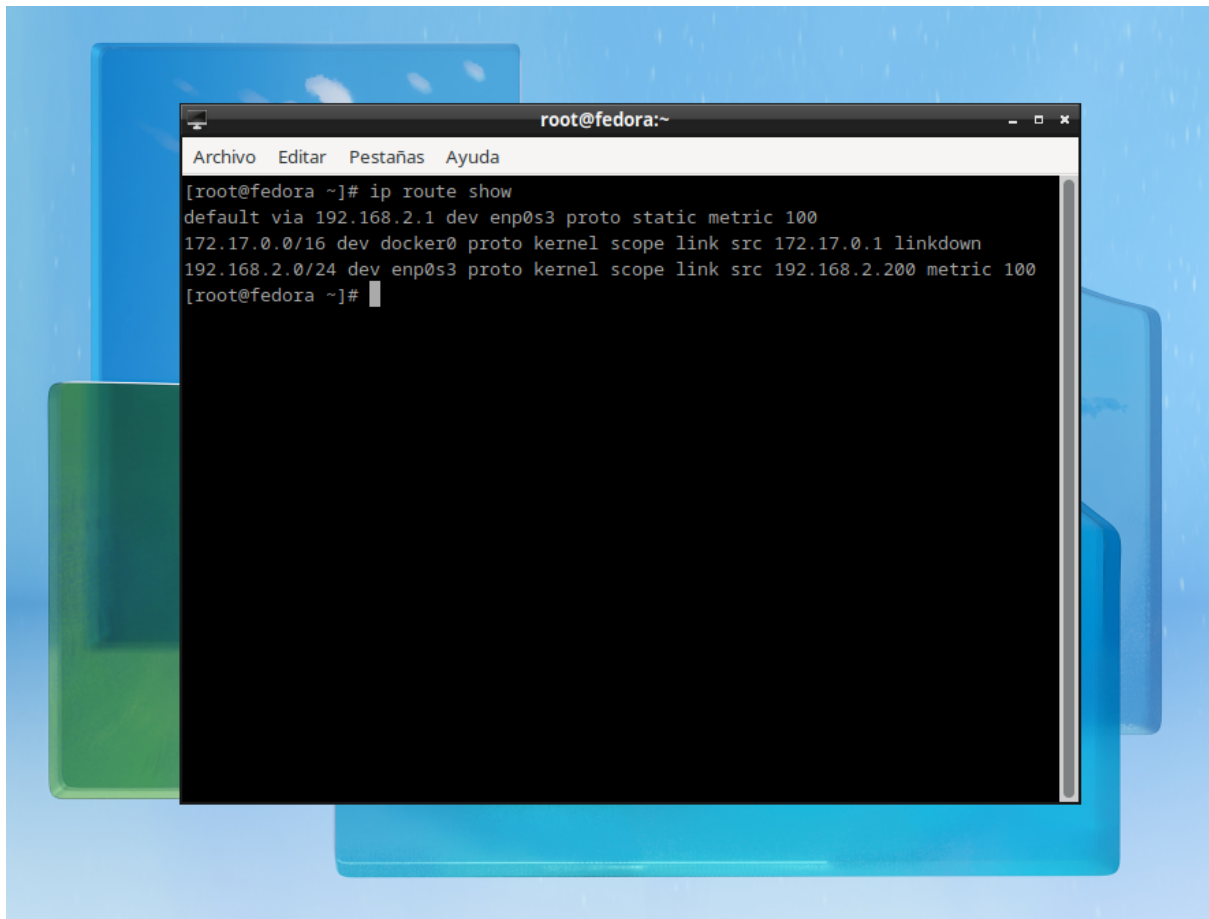
su -



Paso 2:

Luego utilizaremos el comando **ip route show** para ver nuestra IP, el nombre de nuestra tarjeta de red y nuestro GATEWAY.

Le sacaremos una foto con nuestro teléfono porque luego lo vamos a necesitar para utilizarlo, o también lo puedes copiar.



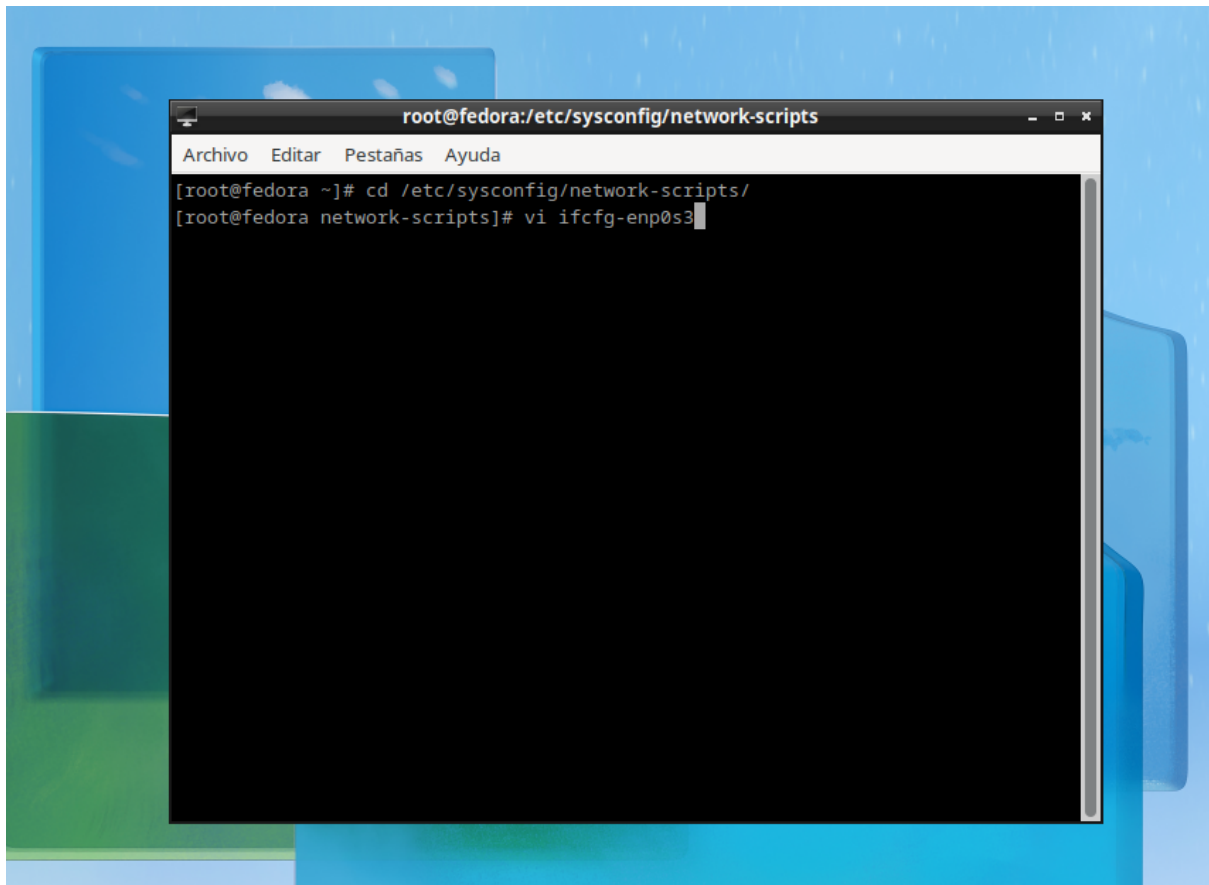
Paso 3:

Luego de sacarle una foto ejecutaremos el comando

cd /etc/sysconfig/network-scripts/ esto lo haremos para ir al directorio en el cual configuramos nuestra IP FIJA.

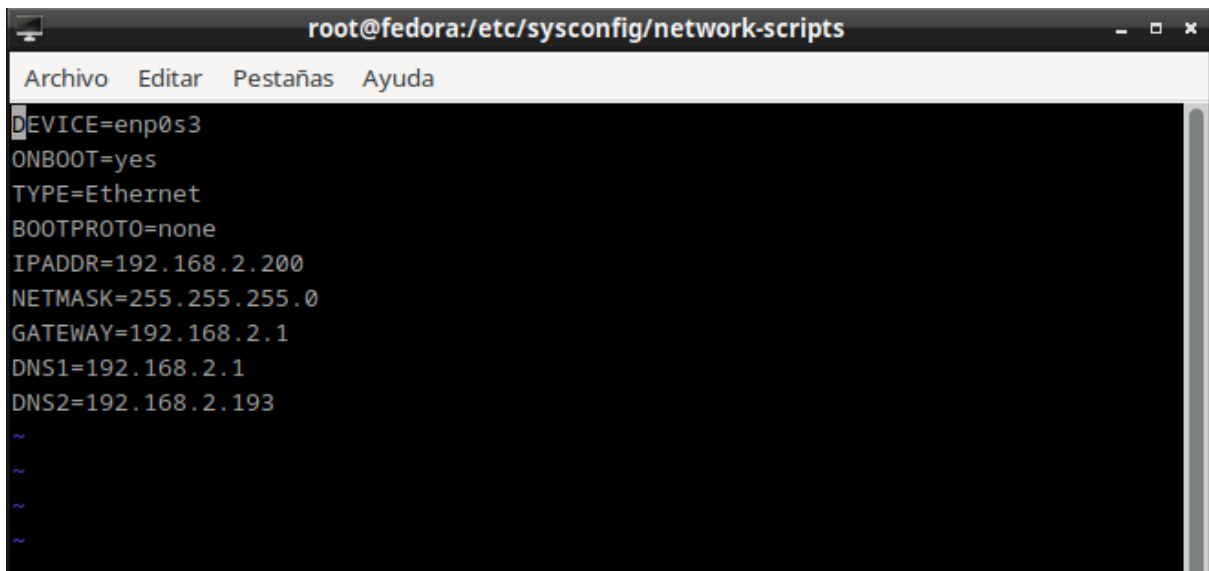
Además de ejecutar ese comando, ejecutaremos el siguiente **vi ifcfg-**(El nombre de nuestra tarjeta de red) en este caso en la foto que sacamos se podría ver que en la primera línea se ve como está escrito enp0s3 este es el nombre de nuestra tarjeta de red.

vi ifcfg-enp0s3



Paso 4:

En este casi último paso llenaríamos el archivo con los siguientes comandos.

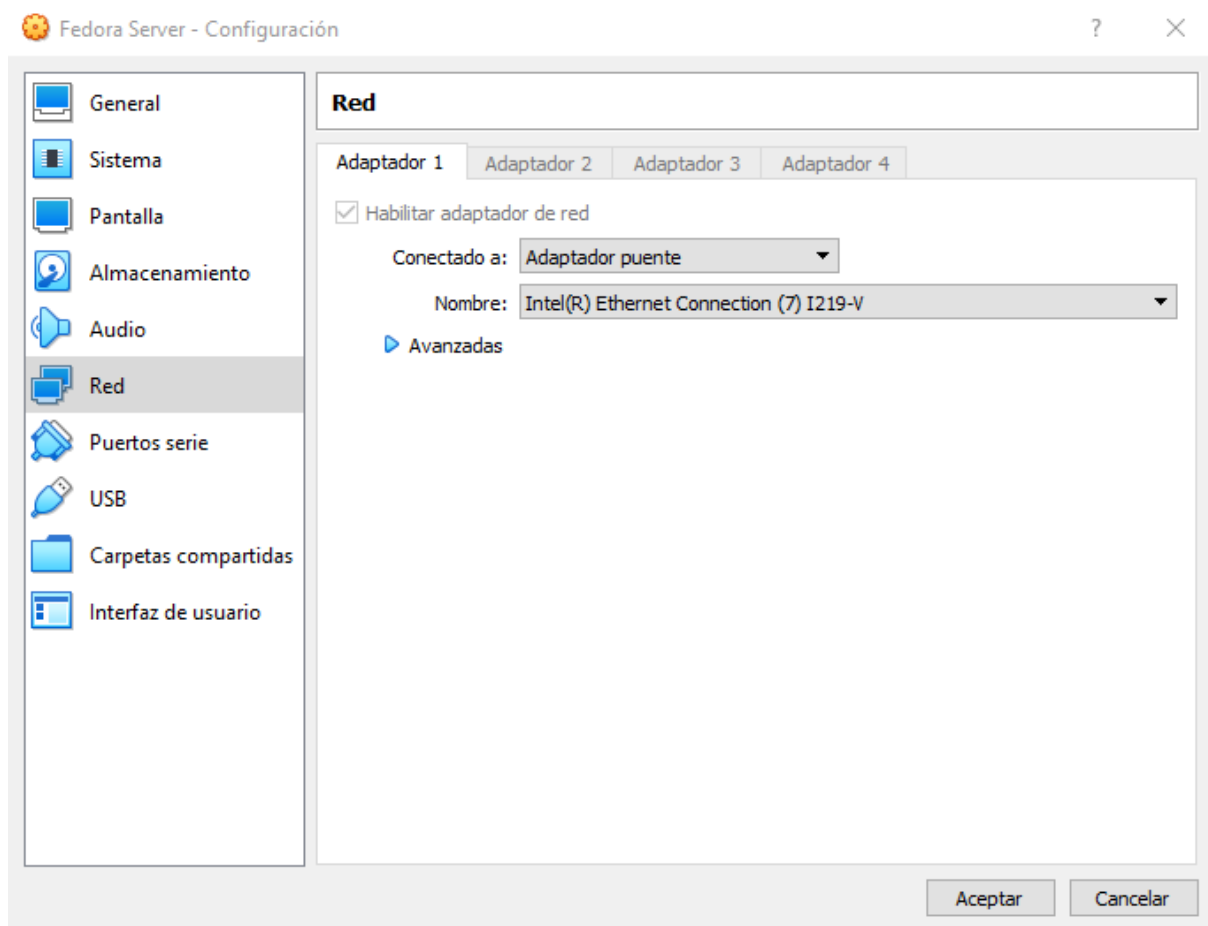


Como otros datos no es necesario que el IPADDR, GATEWAY y ambos DNS sean iguales que al de la foto podremos cambiarlos, obviamente respetando siempre el 192.168 en el caso del IPADDR y el GATEWAY, por el lado de los DNS podremos poner cualquier por ejemplo la del google.

Paso 5:

Por último reiniciamos la máquina virtual y listo estaría hecho, para verificar si está bien hecho podríamos ejecutar en la consola `ip route show` y verificaremos si son los datos puestos.

Puede pasar que no haya funcionado por la configuración del adaptador puente, en este caso apagaremos nuestra máquina virtual entramos en virtual box o su máquina virtualizada y entramos a configuración de nuestro fedora server, seleccionaremos **RED** y en **CONECTADO A:** seleccionaremos **ADAPTADOR PUENTE**



Configuracion SSH

Paso 1:

En este caso nuevamente entramos con nuestro usuario **root**, para luego ejecutar el comando **dnf install openssh-server**, ejecutamos el comando y se nos empezará a descargar archivos.

```
=====
Package                Architecture Version                Repository            Size
=====
Installing:
openssh-server          x86_64             8.7p1-3.fc35          updates                451 k
=====
Transaction Summary
=====
Install 1 Package

Total download size: 451 k
Installed size: 1.0 M
Is this ok [y/N]: █
```

Paso 2:

Listo, ya lo tendríamos instalado, para iniciar pondremos los 3 siguientes comandos

```
[root@fedora ~]# systemctl enable sshd
[root@fedora ~]# systemctl start sshd
[root@fedora ~]# systemctl status sshd
Unknown command verb statusas.
[root@fedora ~]# systemctl status sshd
• sshd.service - OpenSSH server daemon
   Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; vendor preset: enabled)
   Active: active (running) since Thu 2022-07-07 16:56:27 -03; 48min ago
     Docs: man:sshd(8)
           man:sshd_config(5)
   Main PID: 714 (sshd)
     Tasks: 1 (limit: 2319)
    Memory: 2.3M
       CPU: 15ms
    CGroup: /system.slice/sshd.service
            └─ 714 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

jul 07 16:56:27 fedora systemd[1]: Starting sshd.service - OpenSSH server daemon:
jul 07 16:56:27 fedora sshd[714]: Server listening on 0.0.0.0 port 22.
jul 07 16:56:27 fedora sshd[714]: Server listening on :: port 22.
jul 07 16:56:27 fedora systemd[1]: Started sshd.service - OpenSSH server daemon.
lines 1-16/16 (END)
```


Instalación de Docker dentro de la máquina virtual con el sistema operativo a utilizar. (WEB)

Paso 1:

Aquí básicamente estamos instalando los plugins necesarios para instalar el docker.

```
– # dnf -y install dnf-plugins-core
```

```
[root@fedora ~]# dnf -y install dnf-plugins-core
Última comprobación de caducidad de metadatos hecha hace 0:03:59, el vie 10 jun 2022 10:06:27.
El paquete dnf-plugins-core-4.0.24-1.fc35.noarch ya está instalado.
Dependencias resueltas.
Nada por hacer.
¡Listo!
[root@fedora ~]# █
```

Paso 2:

En este paso se agrega el repositorio docker.

```
– # dnf config-manager \
```

```
–
```

```
--add-repo \
```

```
–
```

```
https://download.docker.com/linux/fedora/docker-ce.repo
```

```
[root@fedora ~]# dnf config-manager \
> --add-repo \
> https://download.docker.com/linux/fedora/docker-ce.repo
Agregando repositorio de: https://download.docker.com/linux/fedora/docker-ce.repo
[root@fedora ~]# █
```

Paso 3:

En este paso instalamos el docker.

– # dnf install docker-ce docker-ce-cli containerd.io docker-compose-plugin

```
[root@fedora ~]# dnf install docker-ce docker-ce-cli containerd.io docker-compose-plugin
Docker CE Stable - x86_64
El paquete docker-ce-3:20.10.12-3.fc35.x86_64 ya está instalado.
El paquete docker-ce-cli-1:20.10.12-3.fc35.x86_64 ya está instalado.
El paquete containerd.io-1.4.12-3.1.fc35.x86_64 ya está instalado.
Dependencias resueltas.
=====
Paquete                Arquitectura          Versión                Repositorio            Tam.
=====
Instalando:
docker-compose-plugin   x86_64                2.6.0-3.fc35          docker-ce-stable       6.8 M
=====
Resumen de la transacción
=====
Instalar 1 Paquete

Tamaño total de la descarga: 6.8 M
Tamaño instalado: 25 M
¿Está de acuerdo [s/N]? : s
Descargando paquetes:
docker-compose-plugin-2.6.0-3.fc35.x86_64.rpm
-----
Total
Ejecutando verificación de operación
Verificación de operación exitosa.
Ejecutando prueba de operaciones
Prueba de operación exitosa.
Ejecutando operación
Preparando      :
Instalando      : docker-compose-plugin-2.6.0-3.fc35.x86_64
Ejecutando scriptlet: docker-compose-plugin-2.6.0-3.fc35.x86_64
Verificando     : docker-compose-plugin-2.6.0-3.fc35.x86_64
-----
347 kB/s | 6.8 MB | 00:20
347 kB/s | 6.8 MB | 00:20
-----
1/1
1/1
1/1
1/1

Instalado:
docker-compose-plugin-2.6.0-3.fc35.x86_64

¡Listo!
[root@fedora ~]# █
```

Paso 4:

En este paso iniciamos el docker.

– # systemctl start docker

– # systemctl enable docker

```
[root@fedora ~]# systemctl start docker
[root@fedora ~]# systemctl enable docker
[root@fedora ~]# █
```



Paso 5:

Y finalmente con este comando vemos si está correctamente instalado.

– # docker run hello-world

```
[root@fedora ~]# docker run hello-world
```

Hello from Docker!

This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:

1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
(amd64)
3. The Docker daemon created a new container from that image which runs the executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it to your terminal.

To try something more ambitious, you can run an Ubuntu container with:

```
$ docker run -it ubuntu bash
```

Share images, automate workflows, and more with a free Docker ID:

<https://hub.docker.com/>

For more examples and ideas, visit:

<https://docs.docker.com/get-started/>

```
[root@fedora ~]# █
```

Instalación de un servidor LAMP en la máquina virtual (usando docker o local).(web)

LOCAL

Primeros pasos

LAMP, significado:

- L – Linux → Sería el kernel que estamos utilizando.
- A – Apache → Es el servidor que se encarga de interpretar a HTTP/HTTPS.
- M – MySQL/MariaDB → Sería el servidor de BD encargado.
- P – PHP → Lenguaje de programación que vamos a utilizar en este curso.

PASO 0:

- La consola nos inicia como USUARIO.
- Se debe cambiar a usuario ROOT para poder configurar nuestro FEDORA SERVER.

```
Luis E. Fagúndez  
[usuario@fedora ~]$ su -  
Contraseña:  
Último inicio de sesión: lun mar 21 16:18:07 -03 2022 en pts/1
```

PASO 1:

- Actualizamos FEDORA SERVER:

Con el comando → `# dnf -y update`

- El comando dnf sirve para administrar los paquetes, en este caso solicitamos que se actualicen todos los paquetes de nuestro servidor

```
[root@fedora ~]# dnf -y update  
Última comprobación de caducidad de metadatos hecha hace 0:29:47, el vie 03 jun 2022 09:55:55.  
Dependencias resueltas.
```

- Instalar herramientas básicas para desarrollo:

→ `# yum groupinstall "Development Tools" "Development Libraries"`

```
[root@fedora ~]# yum groupinstall "Development Tools" "Development Libraries"  
Última comprobación de caducidad de metadatos hecha hace 0:33:00, el vie 03 jun 2022 09:55:55.  
Dependencias resueltas.
```

- Se pedirá una instalación [s/n] (si / no).

```
Instalar      106 Paquetes
Actualizar    15 Paquetes
Revertir      2 Paquetes
```

```
Tamaño total de la descarga: 168 M
¿Está de acuerdo [s/N]?: s
Descargando paquetes:
```

PASO 2:

- Instalamos el WEB SERVER (Apache Linux en éste caso junto a PHP en su última versión disponible)

→ # dnf groupinstall "Web Server"

```
[root@fedora ~]# dnf groupinstall "Web Server"
Última comprobación de caducidad de metadatos hecha hace 0:38:42, el vie 03 jun 2022 09:55:55.
Dependencias resueltas.
```

- Se pedirá una confirmación de instalación [s/n] (si / no).

```
Instalar 13 Paquetes
```

```
Tamaño total de la descarga: 5.0 M
Tamaño instalado: 13 M
¿Está de acuerdo [s/N]?: s
Descargando paquetes:
```

- Con el comando systemctl le indicamos que inicie el servidor web y lo configure para iniciar en el arranque del sistema:

→ systemctl start httpd

```
[root@fedora ~]# systemctl start httpd
[root@fedora ~]# █
```

→ systemctl enable httpd

```
[root@fedora ~]# systemctl enable httpd
[root@fedora ~]# █
```

PASO 3:

- Instalamos el motor de Bases de Datos MariaDB-Server

→ # dnf install mariadb mariadb-server

```
[root@fedora ~]# dnf install mariadb mariadb-server
Última comprobación de caducidad de metadatos hecha hace 0:02:01, el vie 03 jun 2022 10:38:10.
El paquete mariadb-3:10.5.13-1.fc35.x86_64 ya está instalado.
El paquete mariadb-server-3:10.5.13-1.fc35.x86_64 ya está instalado.
Dependencias resueltas.
Nada por hacer.
¡Listo!
```

- Usamos el comando systemctl para proceder igual que apache

→ systemctl start mariadb.service

```
[root@fedora ~]# systemctl start mariadb.service
[root@fedora ~]# █
```

→ systemctl enable mariadb.service

```
[root@fedora ~]# systemctl enable mariadb.service
[root@fedora ~]# █
```

- Realizamos las configuraciones del usuario root de MariaDB ejecutando el siguiente comando

→ /usr/bin/mysql_secure_installation



```
[root@fedora ~]# /usr/bin/mysql_secure_installation
```

NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB
SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!

In order to log into MariaDB to secure it, we'll need the current
password for the root user. If you've just installed MariaDB, and
haven't set the root password yet, you should just press enter here.

Enter current password for root (enter for none):
OK, successfully used password, moving on...

Setting the root password or using the unix_socket ensures that nobody
can log into the MariaDB root user without the proper authorisation.

You already have your root account protected, so you can safely answer 'n'.

Switch to unix_socket authentication [Y/n] y
Enabled successfully!
Reloading privilege tables..
... Success!



You already have your root account protected, so you can safely answer 'n'.

Change the root password? [Y/n] n
... skipping.

By default, a MariaDB installation has an anonymous user, allowing anyone to log into MariaDB without having to have a user account created for them. This is intended only for testing, and to make the installation go a bit smoother. You should remove them before moving into a production environment.

Remove anonymous users? [Y/n] n
... skipping.

Normally, root should only be allowed to connect from 'localhost'. This ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n] n
... skipping.

By default, MariaDB comes with a database named 'test' that anyone can access. This is also intended only for testing, and should be removed before moving into a production environment.

Remove test database and access to it? [Y/n] n
... skipping.

Reloading the privilege tables will ensure that all changes made so far will take effect immediately.

Reload privilege tables now? [Y/n] n
... skipping.

Cleaning up...

All done! If you've completed all of the above steps, your MariaDB installation should now be secure.

Thanks for using MariaDB!

PASO 4:

- Vamos a probar que todo funciona correctamente, para eso desde la consola vamos a testear el motor de la base de datos usando el siguiente comando:

→ `$ mysql -u root -p`

- Luego de ejecutar este comando nos pedirá el password del root que colocamos en el anterior comando.

```
[root@fedora ~]# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 9
Server version: 10.5.13-MariaDB MariaDB Server
```

```
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.
```

```
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
```

```
MariaDB [(none)]> █
```

PASO 5:

- Para probar que PHP funciona junto a APACHE nos basta simplemente con crear un archivo .php llamado index.php y lo guardamos en /var/www/html/prueba/index.php, el archivo contendrá el siguiente código:

```
[root@fedora ~]# vi index.php /var/www/html/prueba█
```

→ `<?php
phpinfo();`

→ `?>`

```
?php
phpinfo();
```

```
?>
~
```

- Para ejecutarlo simplemente vamos a ejecutar en un navegador web localhost/prueba



Manual de Instalación de MySQL/MariaDB junto a sus requisitos para trabajar con Docker.

Paso 0:

En este paso procederemos a logear con nuestra cuenta de [Docker Hub](https://hub.docker.com) , utilizando el siguiente comandos:

-\$ docker login

```
[root@fedora ~]# docker login
Login with your Docker ID to push and pull images from Docker Hub. If you don't have a Docker ID, head over to https://hub.docker.com to create one.
Username: gforze
Password:
WARNING! Your password will be stored unencrypted in /root/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
```

Paso 1:

Vamos a descargar la última imagen oficial de MySQL desde docker hub. Para eso ejecutamos:

-\$ docker pull mysql

```
[root@fedora ~]# docker pull mysql
Using default tag: latest
latest: Pulling from library/mysql
Digest: sha256:548da4c67fd8a71908f17c308b8ddb098acf5191d3d7694e56801c6a8b2072cc
Status: Image is up to date for mysql:latest
docker.io/library/mysql:latest
```

Paso 2:

Ahora que tenemos mysql instalado en docker vamos a crear un volumen. En este caso vamos a usar el mismo guión creado previamente.

-\$ docker volume create proyecto_mysql

```
[root@fedora ~]# docker volume create mysql_proyecto
mysql_proyecto
```

Paso 3:

Con el siguiente comando podemos visualizar los contenedores con sus correspondientes nombres.

`-$docker ps`

```
[root@fedora ~]# docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
--------------	-------	---------	---------	--------	-------	-------

Paso 4:

Ejecutamos el contenedor

`-$ docker run -d -p 33060:3306 --name mysql_proyecto -e MYSQL_ROOT_PASSWORD=1234 --mount src=mysql_proyecto,dst=/var/lib/mysql mysql`

```
[root@fedora ~]# docker run -d -p 33060:3306 --name mysql_proyecto -e MYSQL_ROOT_PASSWORD=1234 --mount src=mysql_proyecto,dst=/var/lib/mysql mysql
93bb51889961369c626a328e74fbe52204e1adcd6c20ed66cf5db036ff115e3f
```

Paso 5:

Para acceder al contenedor ejecutamos:

`-$ docker exec -it mysql_proyecto mysql -p`

```
[root@fedora ~]# docker exec -it mysql_proyecto mysql -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.0.29 MySQL Community Server - GPL

Copyright (c) 2000, 2022, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> exit
```

Paso 6:

Estos comandos se usan básicamente para iniciar el contenedor, para detener el contenedor y eliminar el proceso que ejecutamos.

-\$ docker start mysql_proyecto

-\$ docker stop mysql_proyecto

-\$ docker rm mysql_proyecto

```
[root@fedora ~]# docker start mysql_proyecto
mysql_proyecto
[root@fedora ~]# docker start mysql_proyecto
mysql_proyecto
[root@fedora ~]# docker stop mysql_proyecto
mysql_proyecto
[root@fedora ~]# docker rm mysql_proyecto
mysql_proyecto
[root@fedora ~]# █
```