

## Primer Parcial

Steven Cardona Pérez  
240220192008

Ingeniería de Software  
Computación en la nube: contenedores  
Docente Johan Sebastián Giraldo Hurtado  
Institución universitaria EAM  
2023

# Tabla de contenido

<b>Instalación del servidor Nginx</b>	<b>3</b>
<b>Montaje de un disco duro en linux</b>	<b>7</b>

## Instalación del servidor Nginx

Nginx (pronunciado "engine-x") es un popular servidor web y proxy inverso de código abierto. Fue diseñado para ser eficiente, escalable y de alto rendimiento en la entrega de contenido web estático y dinámico. Originalmente creado por Igor Sysoev en 2004, Nginx se ha convertido en una herramienta esencial para muchos administradores de sistemas y desarrolladores web debido a su capacidad para manejar una gran cantidad de solicitudes simultáneas con recursos mínimos.

Las principales características de Nginx incluyen:

**Servidor Web:** Nginx puede servir archivos estáticos, como HTML, CSS, JavaScript y otros recursos, directamente a los navegadores de los usuarios. Su enfoque eficiente permite servir estas solicitudes con un uso mínimo de recursos.

**Proxy Inverso:** Nginx es ampliamente utilizado como un proxy inverso, que actúa como intermediario entre los clientes y los servidores de aplicaciones. Puede distribuir el tráfico a varios servidores backend para lograr equilibrio de carga y alta disponibilidad.

**Balanceo de Carga:** Nginx puede distribuir las solicitudes entrantes entre varios servidores backend para evitar la sobrecarga de un solo servidor y mejorar la estabilidad y el rendimiento del sistema.

**Proxy para Aplicaciones:** Es común utilizar Nginx para dirigir el tráfico a aplicaciones web como Node.js, Ruby on Rails, Python WSGI y otros servidores de aplicaciones, mejorando el rendimiento y la seguridad al actuar como un intermediario.

**TLS/SSL:** Nginx puede funcionar como un servidor proxy SSL, lo que significa que puede gestionar la encriptación y descifrado de datos, lo que garantiza conexiones seguras entre el cliente y el servidor.

**Redirecciones y Rewrite:** Permite realizar redirecciones y reescrituras de URL, lo que es útil para el manejo de direcciones URL amigables o para redireccionar tráfico de una manera específica.

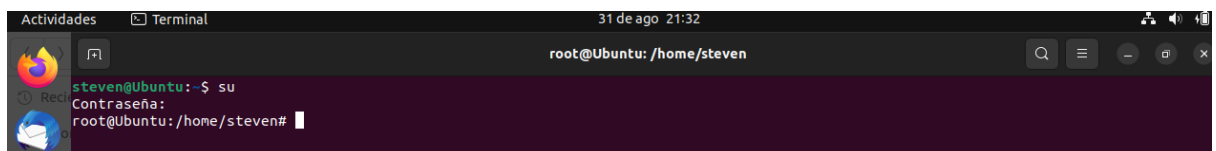
Caché: Nginx puede almacenar en caché contenido estático y dinámico, lo que acelera la entrega de contenido a los usuarios y reduce la carga en los servidores backend.

Control de Acceso: Proporciona herramientas para controlar y limitar el acceso a ciertos recursos basados en reglas de configuración, mejorando la seguridad.

En resumen, Nginx es una herramienta versátil que se utiliza ampliamente para mejorar el rendimiento, la escalabilidad y la seguridad de las aplicaciones web y los sitios web. Es especialmente valioso en entornos donde se espera un alto volumen de tráfico y se busca optimizar la entrega de contenido.

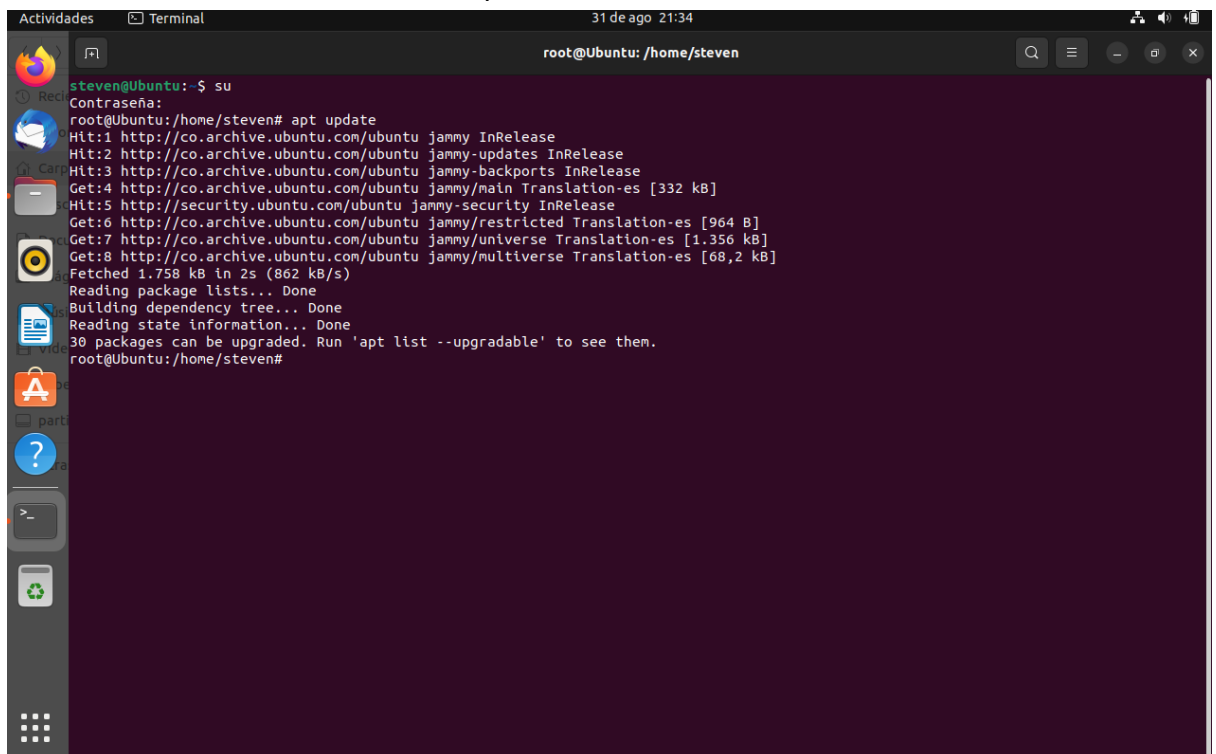
### Proceso de instalación en Linux ubuntu:

1. Se debe abrir la terminal de Linux
2. Se inicia como superusuario con el comando su y posteriormente pone su contraseña



```
Actividades Terminal 31 de ago 21:32
root@Ubuntu: /home/steven
steven@Ubuntu:~$ su
Contraseña:
root@Ubuntu: /home/steven#
```

3. Se actualiza el SO con el comando update



```
Actividades Terminal 31 de ago 21:34
root@Ubuntu: /home/steven
steven@Ubuntu:~$ su
Contraseña:
root@Ubuntu: /home/steven# apt update
Hit:1 http://co.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://co.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://co.archive.ubuntu.com/ubuntu jammy-backports InRelease
Get:4 http://co.archive.ubuntu.com/ubuntu jammy/main Translation-es [332 kB]
Hit:5 http://security.ubuntu.com/ubuntu jammy-security InRelease
Get:6 http://co.archive.ubuntu.com/ubuntu jammy/restricted Translation-es [964 B]
Get:7 http://co.archive.ubuntu.com/ubuntu jammy/universe Translation-es [1.356 kB]
Get:8 http://co.archive.ubuntu.com/ubuntu jammy/multiverse Translation-es [68,2 kB]
Fetched 1.758 kB in 2s (862 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
30 packages can be upgraded. Run 'apt list --upgradable' to see them.
root@Ubuntu: /home/steven#
```

4. Instalamos el servidor con el comando apt install nginx, finalmente tecleamos S para confirmar

```

root@Ubuntu:/home/steven# apt install nginx
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libnginx-mod-http-geoip2 libnginx-mod-http-image-filter libnginx-mod-http-xslt-filter libnginx-mod-mail libnginx-mod-stream
  libnginx-mod-stream-geoip2 nginx-common nginx-core
Suggested packages:
  fcgiwrap nginx-doc
The following NEW packages will be installed:
  libnginx-mod-http-geoip2 libnginx-mod-http-image-filter libnginx-mod-http-xslt-filter libnginx-mod-mail libnginx-mod-stream
  libnginx-mod-stream-geoip2 nginx nginx-common nginx-core
0 upgraded, 9 newly installed, 0 to remove and 30 not upgraded.
Need to get 697 kB of archives.
After this operation, 2.395 kB of additional disk space will be used.
Do you want to continue? [S/n] s

```

5. Para verificar que el servicio está corriendo correctamente, ejecutamos el comando `systemctl status nginx`, como se puede ver en la imagen, el servidor se esta ejecutando correctamente

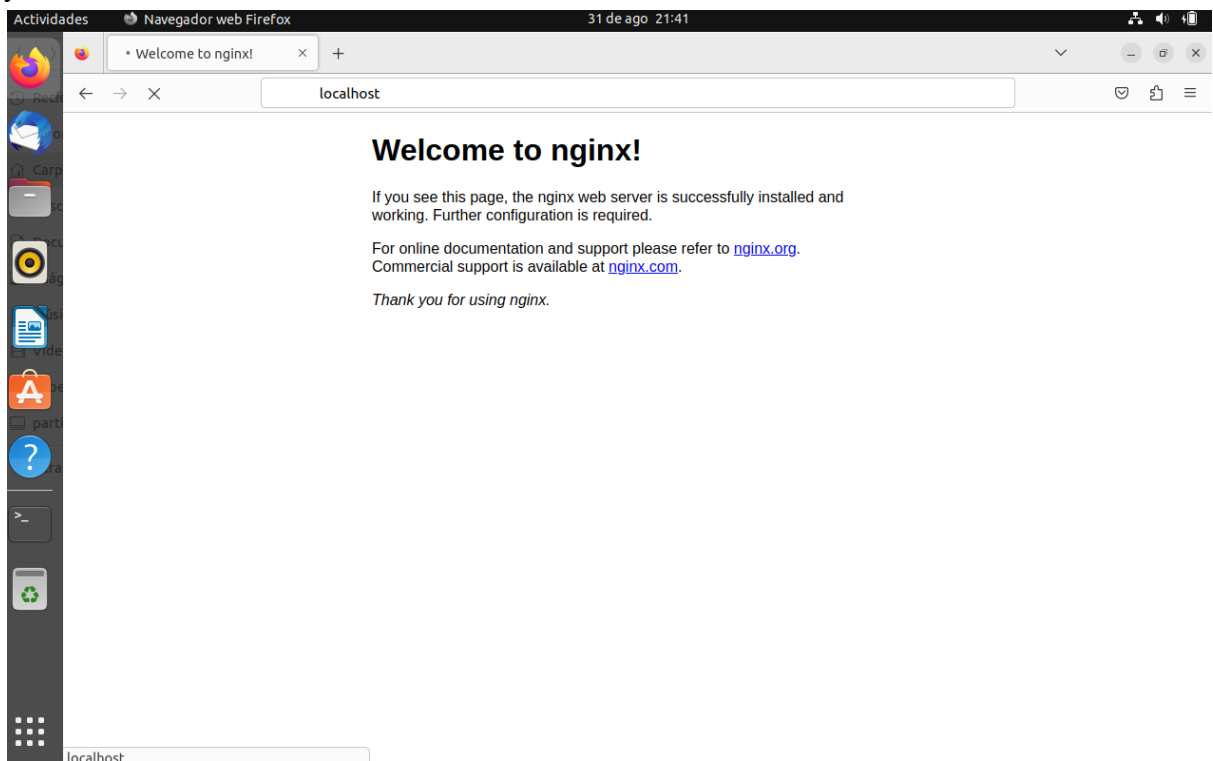
```

root@Ubuntu:/home/steven# systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
   Loaded: loaded (/lib/systemd/system/nginx.service; enabled; vendor preset: enabled)
   Active: active (running) since Thu 2023-08-31 21:35:58 -05; 3min 16s ago
     Docs: man:nginx(8)
  Process: 5165 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
  Process: 5166 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
 Main PID: 5259 (nginx)
    Tasks: 6 (limit: 4871)
   Memory: 6.1M
      CPU: 97ms
   CGroup: /system.slice/nginx.service
           └─5259 "nginx: master process /usr/sbin/nginx -g daemon on; master_process on;"
             └─5262 "nginx: worker process"
             └─5263 "nginx: worker process"
             └─5264 "nginx: worker process"
             └─5265 "nginx: worker process"
             └─5266 "nginx: worker process"

ago 31 21:35:58 Ubuntu systemd[1]: Starting A high performance web server and a reverse proxy server...
ago 31 21:35:58 Ubuntu systemd[1]: Started A high performance web server and a reverse proxy server.
root@Ubuntu:/home/steven#

```

6. Desde el navegador también podemos verificar que efectivamente está funcionando: en el navegador escribimos `localhost` en la barra de direcciones y damos enter.



7. Para modificar el archivo HTML seguimos los siguientes comandos

```

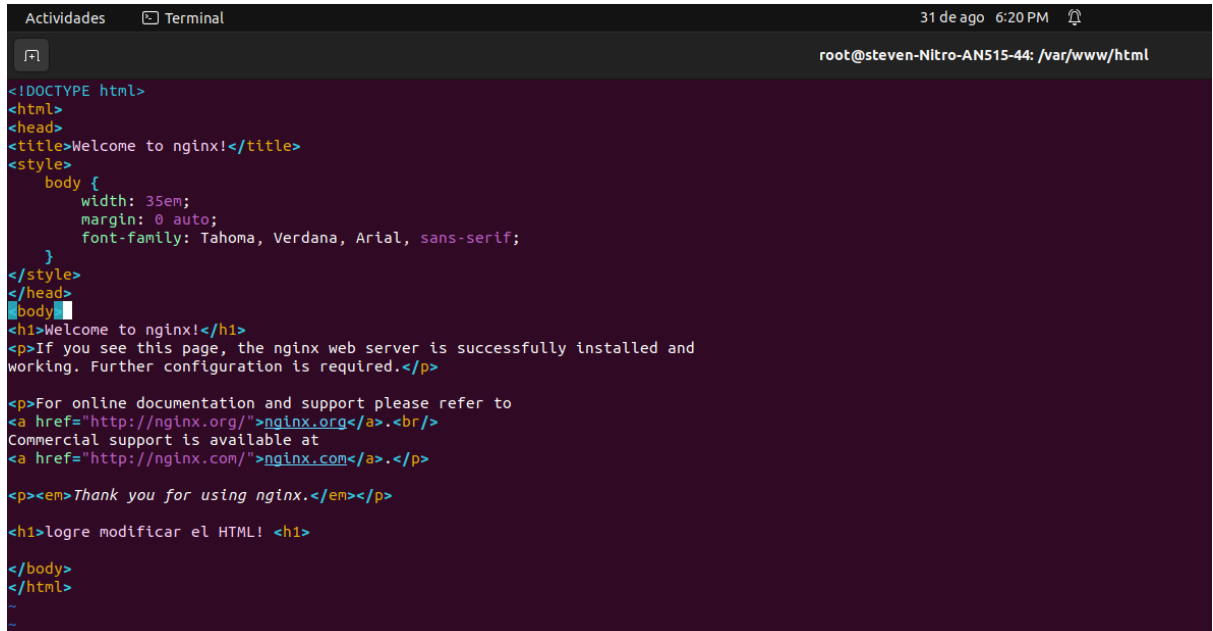
root@steven-Nitro-ANS15-44:~# cd etc/nginx
bash: cd: etc/nginx: No existe el archivo o el directorio
root@steven-Nitro-ANS15-44:~# cd /etc/nginx
root@steven-Nitro-ANS15-44:/etc/nginx# cd sites-enabled
root@steven-Nitro-ANS15-44:/etc/nginx/sites-enabled# ls
default
root@steven-Nitro-ANS15-44:/etc/nginx/sites-enabled# cd /var/www/html/
bash: cd: /var/www/html/: No existe el archivo o el directorio
root@steven-Nitro-ANS15-44:/etc/nginx/sites-enabled# cd /var/www/html/
root@steven-Nitro-ANS15-44:/var/www/html# ls
index.nginx-debian.html
root@steven-Nitro-ANS15-44:/var/www/html# cat index.html
cat: index.html: No existe el archivo o el directorio
root@steven-Nitro-ANS15-44:/var/www/html# cat index.nginx-debian.html
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
    body {
        width: 35em;
        margin: 0 auto;
        font-family: Tahoma, Verdana, Arial, sans-serif;
    }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
root@steven-Nitro-ANS15-44:/var/www/html# vim index.nginx-debian.html
root@steven-Nitro-ANS15-44:/var/www/html# vim index.nginx-debian.html

```

## 8. Una vez abierto el archivo con vim, editamos el HTML



```

Actividades Terminal 31 de ago 6:20 PM
root@steven-Nitro-ANS15-44: /var/www/html

<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
    body {
        width: 35em;
        margin: 0 auto;
        font-family: Tahoma, Verdana, Arial, sans-serif;
    }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>

<h1>logre modificar el HTML! <h1>

</body>
</html>

```

## 9. Resultado

**Welcome to nginx!**

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to [nginx.org](http://nginx.org).  
Commercial support is available at [nginx.com](http://nginx.com).

Thank you for using nginx.

**logre modificar el HTML!**

## Montaje de un disco duro en linux

Montar un disco duro en Linux es el proceso de hacer que el sistema operativo reconozca y acceda a las particiones o unidades de almacenamiento físicas, como discos duros o unidades SSD. Cuando conectas un disco duro nuevo o una unidad de almacenamiento externa a un sistema Linux, el sistema operativo no lo utiliza automáticamente. Debes montar el disco para que el sistema pueda acceder a los datos almacenados en él.

El proceso de montaje implica asignar un punto de montaje, que es una ubicación en el sistema de archivos, donde los contenidos del disco duro se harán disponibles. Esto permite a los usuarios y aplicaciones acceder a los archivos y directorios almacenados en el disco como si fueran parte del sistema de archivos local.

Aquí hay un resumen de los pasos involucrados en el montaje de un disco duro en Linux:

1. **Identificar el Disco:** Antes de montar un disco, debes saber qué dispositivo representa. Esto se puede hacer utilizando comandos como `lsblk` o `fdisk -l`. Estos comandos te mostrarán una lista de discos y particiones en tu sistema.
2. **Crear un Punto de Montaje:** Decide en qué directorio deseas montar el disco. Puedes crear un nuevo directorio o utilizar uno existente. Por ejemplo, podrías crear un directorio llamado "nuevo\_disco".
3. **Montar el Disco:** Utiliza el comando `mount` para montar el disco en el punto de montaje que has creado. La sintaxis básica es:

mount /dev/sdXN /ruta/punto/de/montaje. Primero creamos el directorio llamado “particion” mkdir particion, luego listamos los discos con el comando fdisk -l

```

steven@Ubuntu:~$ su
Contraseña:
root@Ubuntu:~# mkdir particion
root@Ubuntu:~# fdisk -l
Disk /dev/loop0: 4 KiB, 4096 bytes, 8 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/loop1: 63,45 MiB, 66531328 bytes, 129944 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/loop2: 73,88 MiB, 77463552 bytes, 151296 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/loop3: 237,21 MiB, 248729600 bytes, 485800 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/loop4: 349,7 MiB, 366682112 bytes, 716176 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/loop5: 485,52 MiB, 509100032 bytes, 994336 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/loop5: 485,52 MiB, 509100032 bytes, 994336 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/loop6: 91,69 MiB, 96141312 bytes, 187776 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/loop7: 12,32 MiB, 12922880 bytes, 25240 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/sda: 15,48 GiB, 16624394240 bytes, 32469520 sectors
Disk model: VBOX HARDDISK
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: 00900385-AA47-4771-9483-916D1A30A1B0

Device      Start      End  Sectors  Size Type
/dev/sda1    2048     4095     2048    1M BIOS boot
/dev/sda2    4096   1054719   1050624   513M EFI System
/dev/sda3   1054720 32468991 31414272   15G Linux filesystem

Disk /dev/loop8: 53,26 MiB, 55844864 bytes, 109072 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/loop9: 452 KiB, 462848 bytes, 904 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

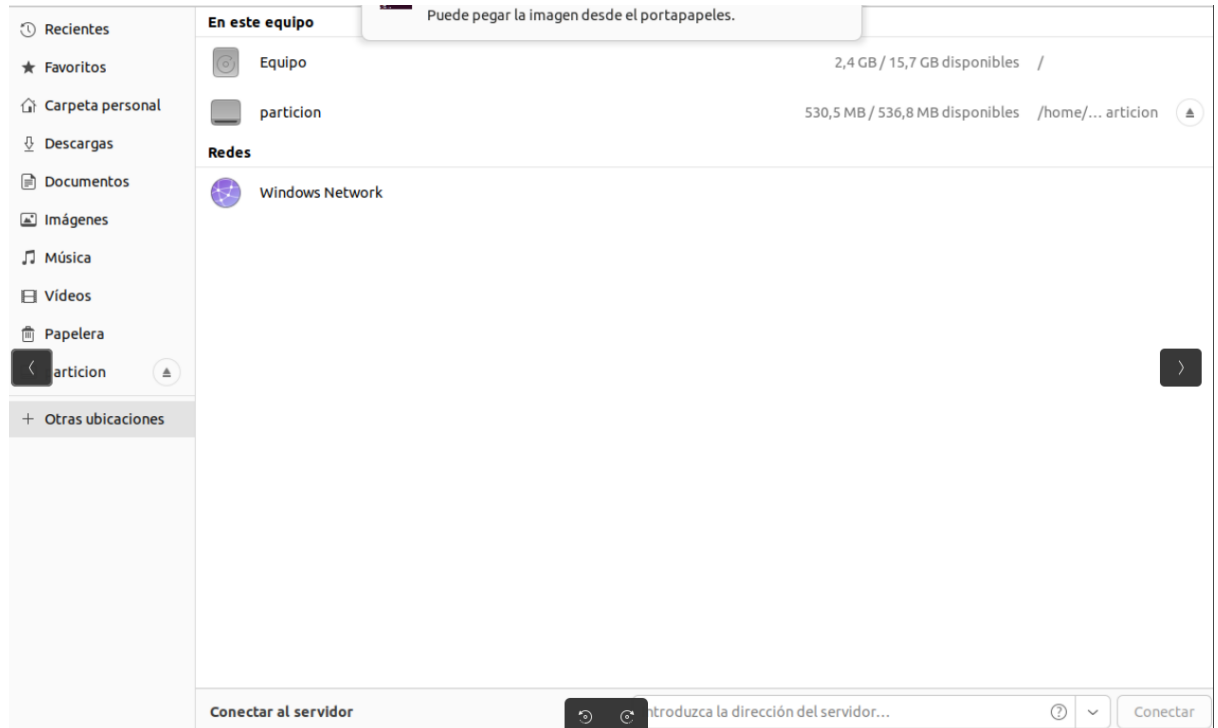
```



4. a continuación con el comando antes mencionado, montamos el disco, en este caso el disco que quiero montar se llama: /dev/sda2

```
root@Ubuntu:/home/steven# mount /dev/sda3 particion
mount: /home/steven/particion: /dev/sda3 already mounted on /.
root@Ubuntu:/home/steven# mount /dev/sda2 particion
root@Ubuntu:/home/steven#
```

5. El resultado



6. Para desmontarlo usamos el comando umount /ruta/punto/de/montaje

```
ntu:/etc/nginx# umount /dev/sda2
ntu:/etc/nginx#
```

