

PARCIAL I – ELECTIVA V (Docker y Kubernetes)

Andrés Mauricio Ocampo Arenas

Institución Universitaria Eam

Ing. De Software

Facultad De Ingeniería

Armenia-Quindio

2023

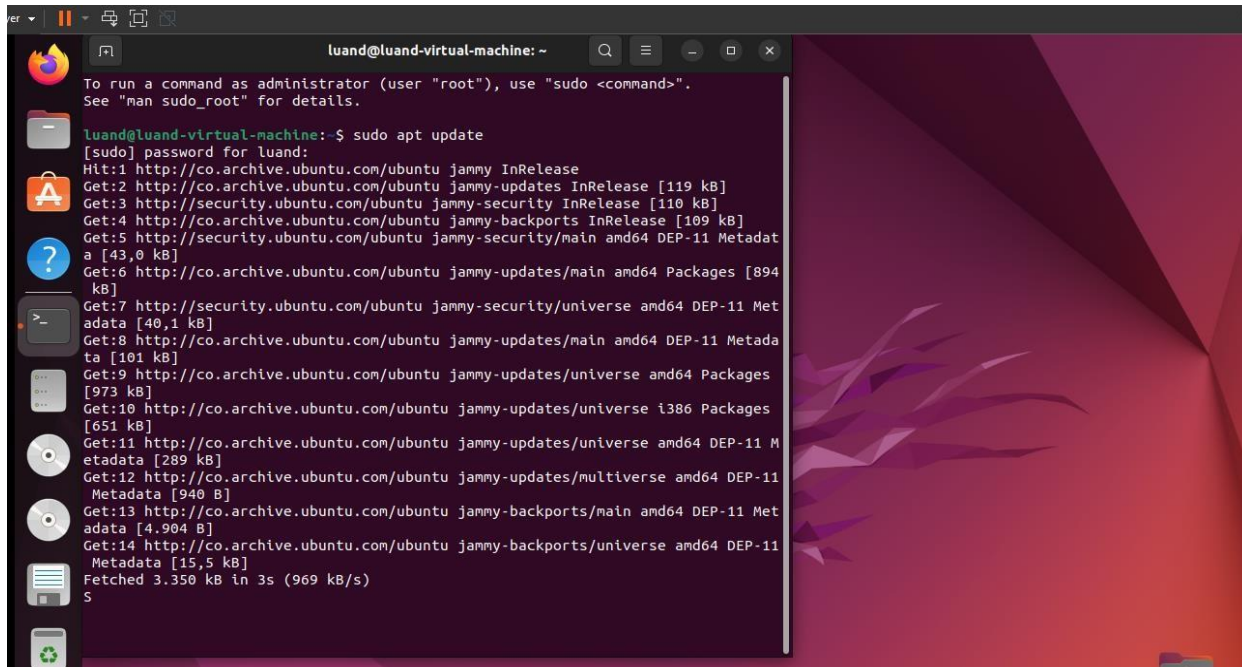
Contenido

GUIA NGINX..... 3

GUÍA PARTICIONES DE DISCO (VMWare)..... 8

GUIA NGINX

Ejecutamos el comando `sudo apt update` para obtener los metadatos actualizados

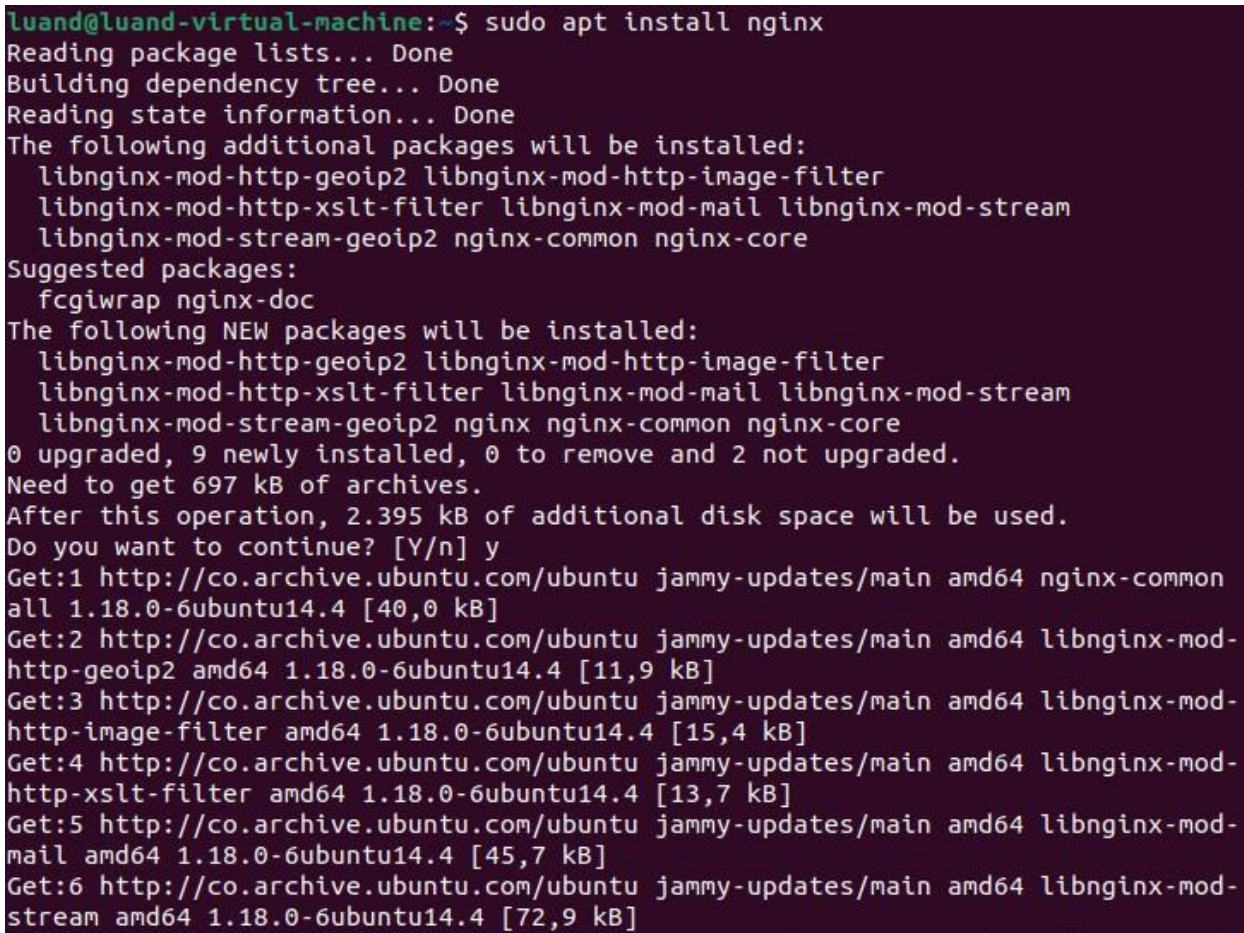


```

luand@luand-virtual-machine: ~
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

luand@luand-virtual-machine:~$ sudo apt update
[sudo] password for luand:
Hit:1 http://co.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://co.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:3 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:4 http://co.archive.ubuntu.com/ubuntu jammy-backports InRelease [109 kB]
Get:5 http://security.ubuntu.com/ubuntu jammy-security/main amd64 DEP-11 Metad
a [43,0 kB]
Get:6 http://co.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [894
kB]
Get:7 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 DEP-11 Met
adata [40,1 kB]
Get:8 http://co.archive.ubuntu.com/ubuntu jammy-updates/main amd64 DEP-11 Metada
ta [101 kB]
Get:9 http://co.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages
[973 kB]
Get:10 http://co.archive.ubuntu.com/ubuntu jammy-updates/universe i386 Packages
[651 kB]
Get:11 http://co.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 DEP-11 M
etadadata [289 kB]
Get:12 http://co.archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 DEP-11
Metadata [940 B]
Get:13 http://co.archive.ubuntu.com/ubuntu jammy-backports/main amd64 DEP-11 Met
adata [4,904 B]
Get:14 http://co.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 DEP-11
Metadata [15,5 kB]
Fetched 3.350 kB in 3s (969 kB/s)
S
  
```

Luego instalamos nginx



```

luand@luand-virtual-machine:~$ sudo 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 2 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? [Y/n] y
Get:1 http://co.archive.ubuntu.com/ubuntu jammy-updates/main amd64 nginx-common
all 1.18.0-6ubuntu14.4 [40,0 kB]
Get:2 http://co.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libnginx-mod-
http-geoip2 amd64 1.18.0-6ubuntu14.4 [11,9 kB]
Get:3 http://co.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libnginx-mod-
http-image-filter amd64 1.18.0-6ubuntu14.4 [15,4 kB]
Get:4 http://co.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libnginx-mod-
http-xslt-filter amd64 1.18.0-6ubuntu14.4 [13,7 kB]
Get:5 http://co.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libnginx-mod-
mail amd64 1.18.0-6ubuntu14.4 [45,7 kB]
Get:6 http://co.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libnginx-mod-
stream amd64 1.18.0-6ubuntu14.4 [72,9 kB]
  
```

En este momento, Si miramos el estado del nginx, nos dice que se encuentra activo

```
luand@luand-virtual-machine:~$ 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: ena
   Active: active (running) since Sat 2023-08-26 09:32:58 -05; 59s ago
     Docs: man:nginx(8)
   Process: 2698 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_proce
   Process: 2699 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (c
 Main PID: 2790 (nginx)
    Tasks: 3 (limit: 2210)
   Memory: 4.5M
      CPU: 68ms
   CGroup: /system.slice/nginx.service
           └─2790 "nginx: master process /usr/sbin/nginx -g daemon on; master
             └─2792 "nginx: worker process" "" "" "" "" "" "" "" "" "" "" ""
             └─2793 "nginx: worker process" "" "" "" "" "" "" "" "" "" "" ""

ago 26 09:32:58 luand-virtual-machine systemd[1]: Starting A high performance w
ago 26 09:32:58 luand-virtual-machine systemd[1]: Started A high performance we
lines 1-17/17 (END)
```

Si entramos a la carpeta de nginx logramos ver los archivos de configuración de nginx

```
luand@luand-virtual-machine:~$ cd /etc/nginx
luand@luand-virtual-machine:/etc/nginx$ ls
conf.d          koi-win          nginx.conf       sites-enabled
fastcgi.conf    mime.types       proxy_params     snippets
fastcgi_params  modules-available scgi_params      uwsgi_params
koi-utf         modules-enabled  sites-available  win-utf
luand@luand-virtual-machine:/etc/nginx$ S
```

Luego , si entramos a la carpeta sites-available podemos ver el sitio por defecto

```
luand@luand-virtual-machine:/etc/nginx$ cd sites-available
luand@luand-virtual-machine:/etc/nginx/sites-available$ ls
default
```

Luego, ya que queremos abrir diferentes sitios web, podemos copiar dicho archivo default, ponerle un nombre para cada html que deseemos mostrar y cambiarle algunas secciones.

```
luand@luand-virtual-machine:/etc/nginx/sites-available$ sudo cp default sitioper
sonal
luand@luand-virtual-machine:/etc/nginx/sites-available$ ls
default  sitioper
luand@luand-virtual-machine:/etc/nginx/sites-available$ nano sitioper
```


Cambiamos algunos elementos de la configuración como el default en el puerto 80, la ruta y asignamos un `server_name`

```
GNU nano 6.2          sitiopersonal *
# Default server configuration
#
server {
    listen 80;
    listen [::]:80;

    # SSL configuration
    #
    # listen 443 ssl default_server;
    # listen [::]:443 ssl default_server;
    #
    # Note: You should disable gzip for SSL traffic.
    # See: https://bugs.debian.org/773332
    #
    # Read up on ssl_ciphers to ensure a secure configuration.
    # See: https://bugs.debian.org/765782
    #
    # Self signed certs generated by the ssl-cert package
    # Don't use them in a production server!
    #
    # include snippets/snakeoil.conf;

    root /var/www/sitiopersonal;

    # Add index.php to the list if you are using PHP
    index index.html index.htm index.nginx-debian.html;

    server_name andresM.com www.andresM.com;
s
```

Con esto, ya queda configurado el archivo

Vamos a la carpeta de sites-enabled y hacemos el proceso para habilitar nuestro sitio web apuntándole al archivo de configuración

```
luand@luand-virtual-machine:/etc/nginx/sites-enabled$ sudo ln -s /etc/nginx/sites-available/sitiopersonal .
luand@luand-virtual-machine:/etc/nginx/sites-enabled$ ls
default  sitiopersonal
```

Creamos una carpeta en la ruta `/var/www/_____` para almacenar el archivo html que queremos mostrar

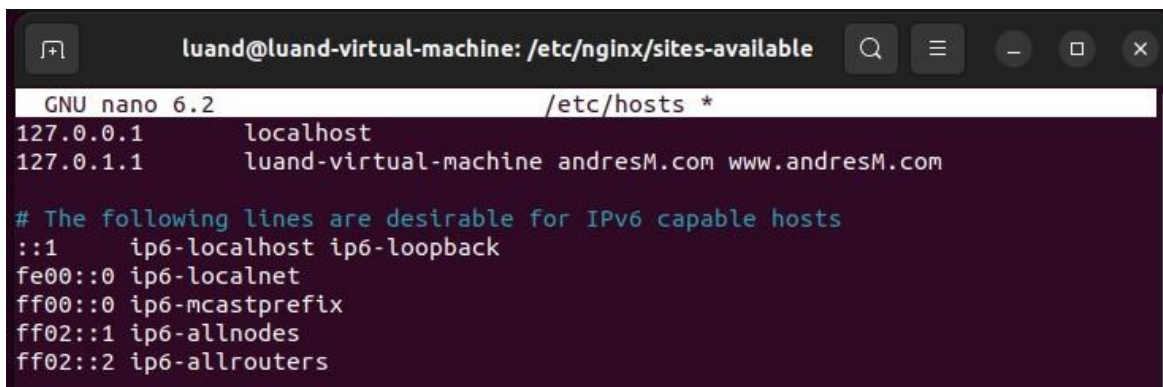
```
luand@luand-virtual-machine:~$ sudo mkdir -p /var/www/sitiopersonal
luand@luand-virtual-machine:~$ cd /var/www/sitiopersonal
luand@luand-virtual-machine:/var/www/sitiopersonal$ cd ..
luand@luand-virtual-machine:/var/www$ ls
html  sitiopersonal
luand@luand-virtual-machine:/var/www$
```

Hacemos un systemctl restart nginx o un ngx -s reload para reiniciar el servicio y que se efectúen los cambios

```
luand@luand-virtual-machine:/etc/nginx/sites-available$ sudo nginx -s reload
```

Luego editamos el archivo hosts

```
luand@luand-virtual-machine:/etc/nginx/sites-available$ sudo nano /etc/hosts
```



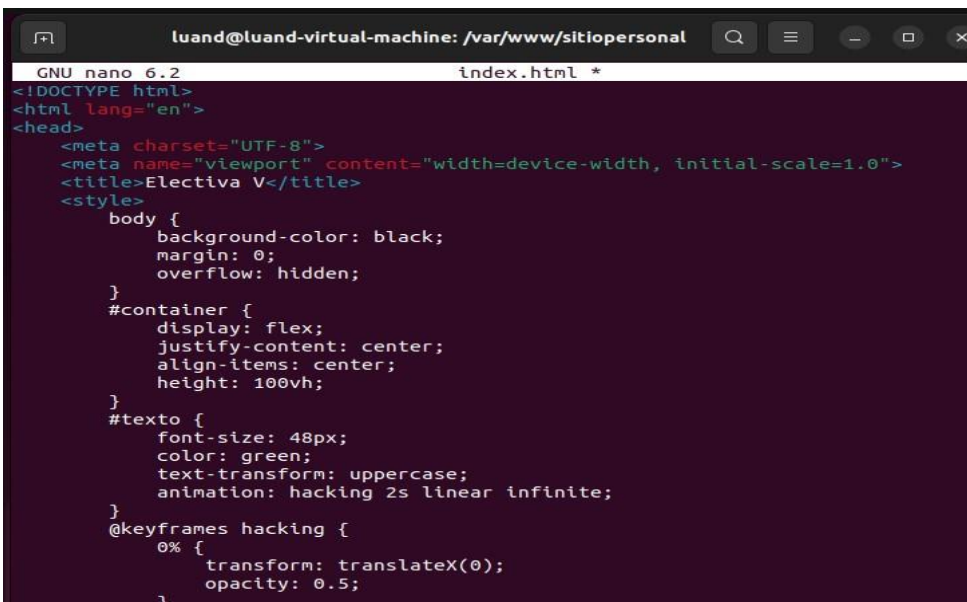
```
GNU nano 6.2 /etc/hosts *
127.0.0.1    localhost
127.0.1.1    luand-virtual-machine andresM.com www.andresM.com

# The following lines are desirable for IPv6 capable hosts
::1         ip6-localhost ip6-loopback
fe00::0     ip6-localnet
ff00::0     ip6-mcastprefix
ff02::1     ip6-allnodes
ff02::2     ip6-allrouters
```

Ahora bien, podemos ir a la carpeta donde alojaremos nuestro html.index y lo crearemos

```
luand@luand-virtual-machine:/etc/nginx/sites-available$ cd /var/www/sitiopersonal
luand@luand-virtual-machine:/var/www/sitiopersonal$ sudo nano index.html
```

A nuestro gusto



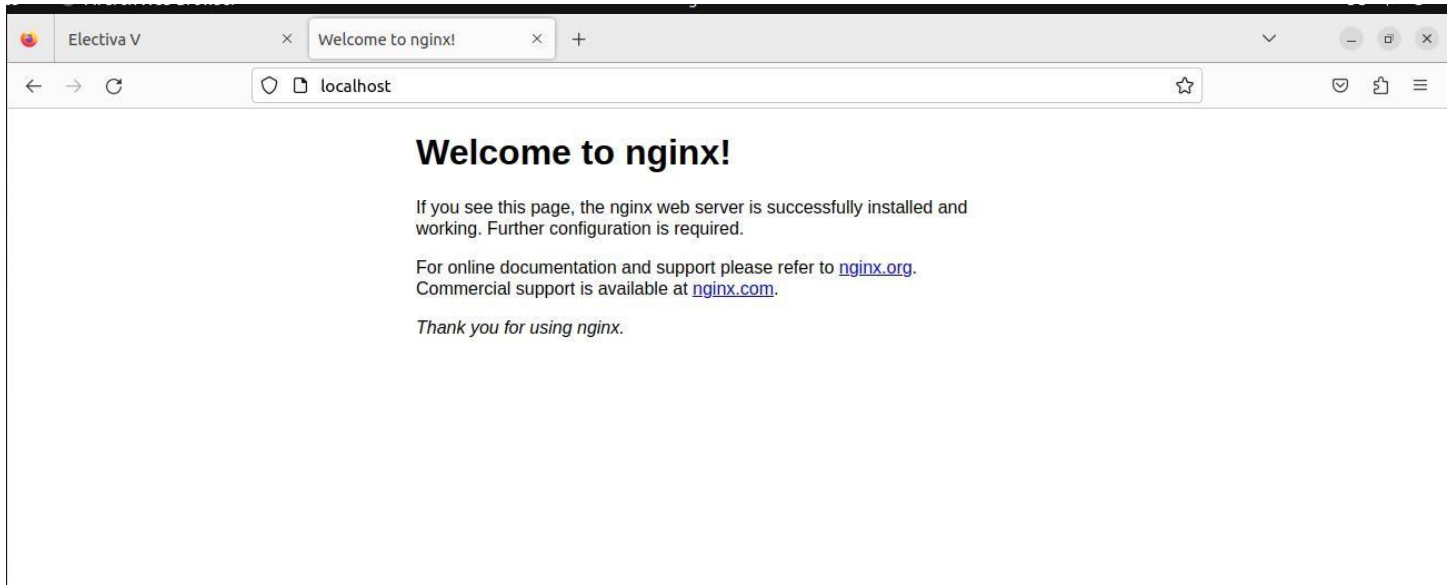
```
GNU nano 6.2 index.html *
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Electiva V</title>
  <style>
    body {
      background-color: black;
      margin: 0;
      overflow: hidden;
    }
    #container {
      display: flex;
      justify-content: center;
      align-items: center;
      height: 100vh;
    }
    #texto {
      font-size: 48px;
      color: green;
      text-transform: uppercase;
      animation: hacking 2s linear infinite;
    }
    @keyframes hacking {
      0% {
        transform: translateX(0);
        opacity: 0.5;
      }
    }
  </style>

```

Y listo, si abrimos un navegador y ponemos la dirección url, veremos nuestro archivo html

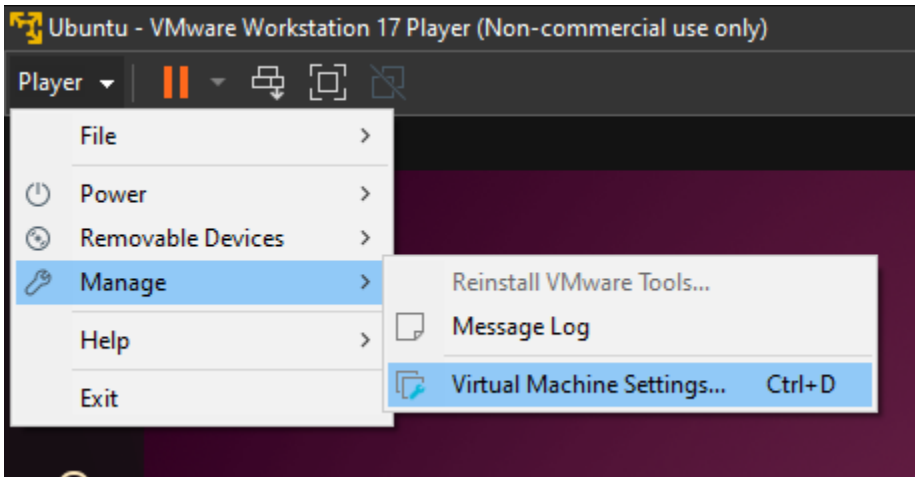


Nota: el localhost lo he dejado por defecto, por ende, aparecerá el html por default que viene con la instalación de nginx

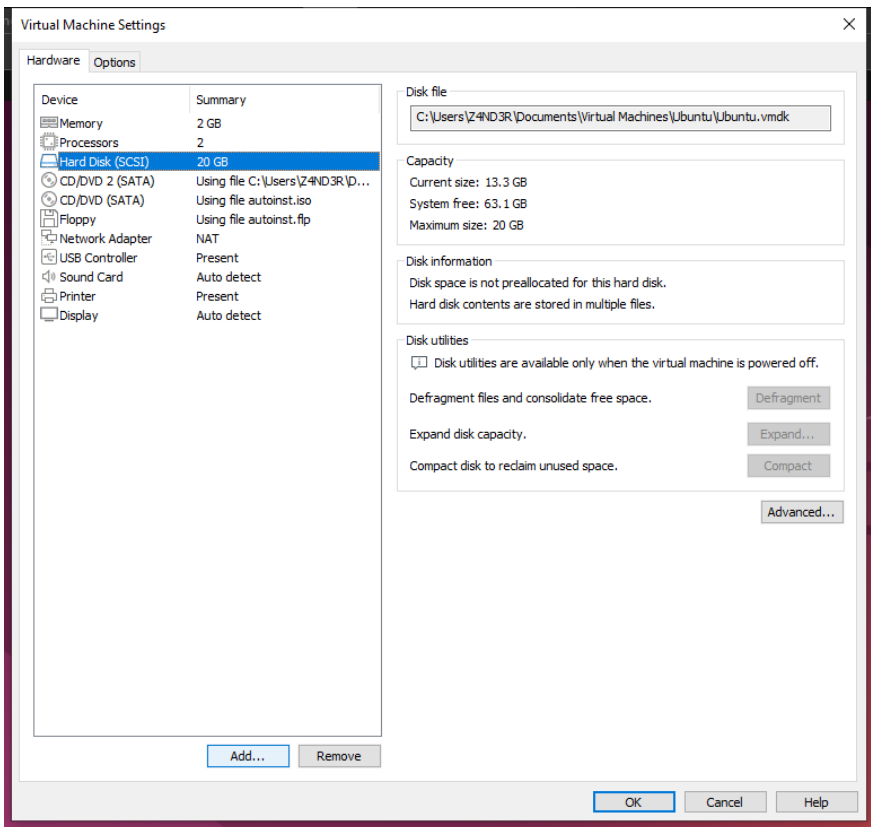


GUÍA PARTICIONES DE DISCO (VMWare)

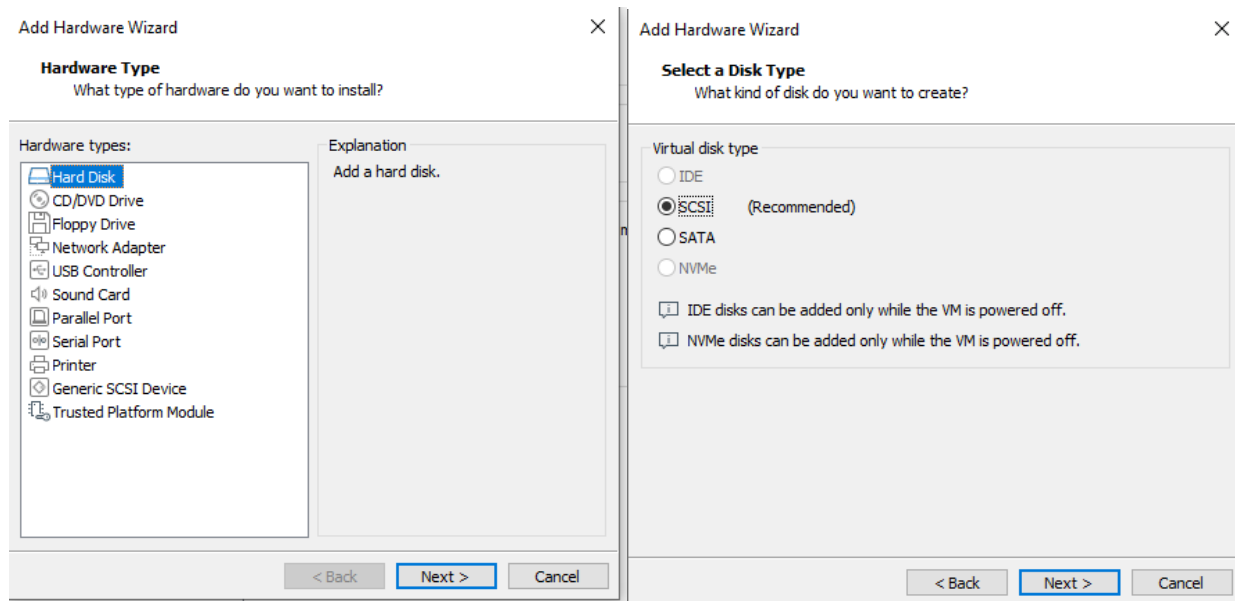
Como primer paso, damos click en player, manage y luego virtual machine settings



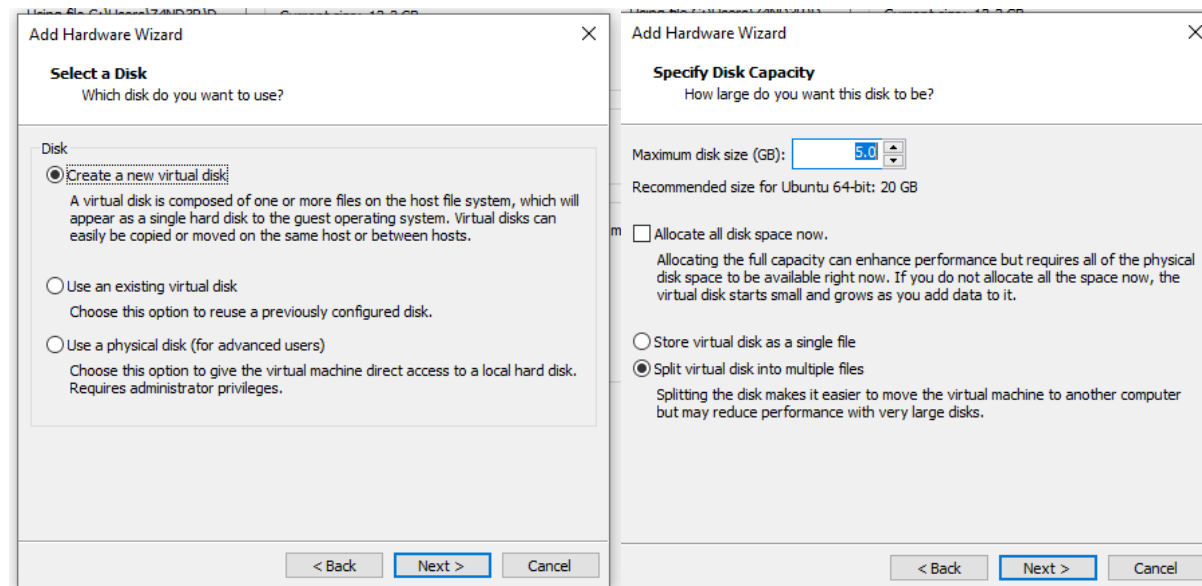
En la ventana que nos abre, seleccionamos hard disks y añadimos uno nuevo



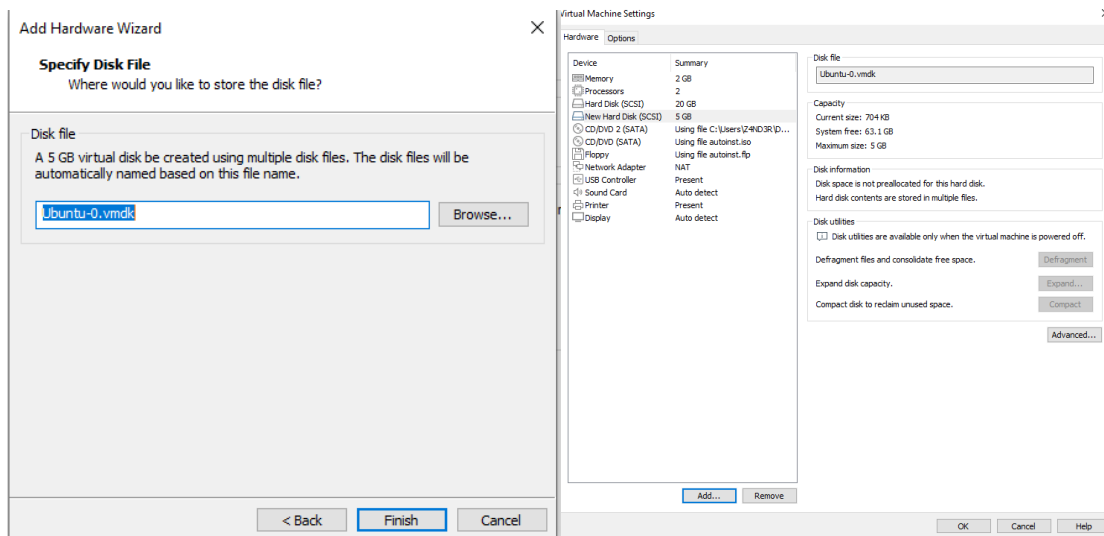
Damos click en Add y seleccionamos Hard disk, damos next y elegimos la opción SCSI, next



Creamos un disco virtual nuevo y le asignamos 5GB para este ejercicio, damos click en next



Le asignamos un nombre o lo dejamos por defecto, damos click en finish y ahora logramos ver que el disco se ha creado



Cerramos, damos click en Ok, volvemos a nuestra máquina virtual y accedemos a la terminal, entramos como usuario root y ejecutamos el comando lsblk, buscamos el disco que acabamos de crear

```

root@luand-virtual-machine: ~
luand@luand-virtual-machine:~$ sudo su -
[sudo] password for luand:
root@luand-virtual-machine:~# lsblk
NAME        MAJ:MIN RM   SIZE RO TYPE MOUNTPOINTS
fd0          2:0    1     4K  0 disk
loop0        7:0     0     4K  1 loop /snap/bare/5
loop1        7:1     0   63,4M  1 loop /snap/core20/1974
loop2        7:2     0   73,9M  1 loop /snap/core22/858
loop3        7:3     0  237,2M  1 loop /snap/firefox/2987
loop4        7:4     0  349,7M  1 loop /snap/gnome-3-38-2004/143
loop5        7:5     0  485,5M  1 loop /snap/gnome-42-2204/120
loop6        7:6     0  485,5M  1 loop /snap/gnome-42-2204/126
loop7        7:7     0   91,7M  1 loop /snap/gtk-common-themes/1535
loop8        7:8     0   12,3M  1 loop /snap/snap-store/959
loop9        7:9     0   53,3M  1 loop /snap/snapd/19457
loop10       7:10    0    452K  1 loop /snap/snapd-desktop-integration/83
sda          8:0     0    20G  0 disk
├─sda1       8:1     0     1M  0 part
├─sda2       8:2     0   513M  0 part /boot/efi
└─sda3       8:3     0   19,5G  0 part /var/snap/firefox/common/host-hunspell
sdb          8:16    0     5G  0 disk
sr0         11:0     1  155,4M  0 rom  /media/luand/CDROM
sr1         11:1     1    4,7G  0 rom  /media/luand/Ubuntu 22.04.3 LTS amd64
root@luand-virtual-machine:~#
    
```

Para este caso, es el sdb así que ejecutamos el comando `fdisk /dev/sdb`, luego de esto escribimos `n`, damos enter, luego en tipo de partición escribimos `p`, damos enter, en número de partición para este caso será `1`, first sector lo dejamos por defecto, y last sector añadimos para este ejemplo `+2GB`

```
root@luand-virtual-machine:~# fdisk /dev/sdb

Welcome to fdisk (util-linux 2.37.2).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Device does not contain a recognized partition table.
Created a new DOS disklabel with disk identifier 0x7e5feb50.

Command (m for help): n
Partition type
   p   primary (0 primary, 0 extended, 4 free)
   e   extended (container for logical partitions)
Select (default p): p
Partition number (1-4, default 1): 1
First sector (2048-10485759, default 2048):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (2048-10485759, default 10485759):
+2GB

Created a new partition 1 of type 'Linux' and of size 1,9 GiB.

Command (m for help):
```

Escribimos `w`, damos enter y escribimos el comando `partprobe -s` que nos detecta la partición

```
Created a new partition 1 of type 'Linux' and of size 1,9 GiB.

Command (m for help): w
The partition table has been altered.
Calling ioctl() to re-read partition table.
Syncing disks.

root@luand-virtual-machine:~# partprobe -s
/dev/sda: gpt partitions 1 2 3
/dev/sdb: msdos partitions 1
Warning: Unable to open /dev/sr0 read-write (Read-only file system). /dev/sr0 has been opened read-only.
/dev/sr0: msdos partitions
Warning: Unable to open /dev/sr1 read-write (Read-only file system). /dev/sr1 has been opened read-only.
/dev/sr1: msdos partitions
root@luand-virtual-machine:~#
```

Ahora, procederemos a darle la extensión a la partición usando el comando `mkfs.ext4 /dev/sdb1`

```
root@luand-virtual-machine:~# mkfs.ext4 /dev/sdb1
mke2fs 1.46.5 (30-Dec-2021)
Creating filesystem with 488192 4k blocks and 122160 inodes
Filesystem UUID: de90c242-fca9-4382-b0b3-4940ea2a50a4
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912

Allocating group tables: done
Writing inode tables: done
Creating journal (8192 blocks): done
Writing superblocks and filesystem accounting information: done

root@luand-virtual-machine:~#
```

Seguidamente, creamos el directorio para montar la partición en él

```
root@luand-virtual-machine:~# mkfs.ext4 /dev/sdb1
mke2fs 1.46.5 (30-Dec-2021)
Creating filesystem with 488192 4k blocks and 122160 inodes
Filesystem UUID: de90c242-fca9-4382-b0b3-4940ea2a50a4
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912

Allocating group tables: done
Writing inode tables: done
Creating journal (8192 blocks): done
Writing superblocks and filesystem accounting information: done

root@luand-virtual-machine:~#
```

Procedemos a montar la partición en ese directorio con el comando `mount rutaParticion rutaDirectorio`

```
root@luand-virtual-machine:~# mkdir /mnt/ext4
root@luand-virtual-machine:~# mount /dev/sdb1 /mnt/ext4
root@luand-virtual-machine:~# df -Th
```

Filesystem	Type	Size	Used	Avail	Use%	Mounted on
tmpfs	tmpfs	193M	1,7M	191M	1%	/run
/dev/sda3	ext4	20G	11G	7,5G	59%	/
tmpfs	tmpfs	962M	0	962M	0%	/dev/shm
tmpfs	tmpfs	5,0M	4,0K	5,0M	1%	/run/lock
/dev/sda2	vfat	512M	6,1M	506M	2%	/boot/efi
tmpfs	tmpfs	193M	100K	193M	1%	/run/user/1000
/dev/sr0	iso9660	156M	156M	0	100%	/media/luand/CDROM
/dev/sr1	iso9660	4,7G	4,7G	0	100%	/media/luand/Ubuntu 22.04.3 LTS amd64
/dev/sdb1	ext4	1,8G	24K	1,7G	1%	/mnt/ext4

```
root@luand-virtual-machine:~#
```


Posteriormente, podemos ingresar a nuestra partición y crear un archivo de prueba para ver el funcionamiento correcto al montar y desmontar la partición

```

root@luand-virtual-machine: /mnt/ext4
root@luand-virtual-machine:/mnt/ext4# sudo nano pruebaMount.txt
root@luand-virtual-machine:/mnt/ext4# ls
lost+found  pruebaMount.txt
root@luand-virtual-machine:/mnt/ext4#

```

Desmontamos la partición, ingresamos al directorio y listamos los elementos para ver si están nuestros archivos

```

root@luand-virtual-machine: /mnt/ext4
root@luand-virtual-machine:~# umount /dev/sdb1 /mnt/ext4
umount: /mnt/ext4: not mounted.
root@luand-virtual-machine:~# cd /mnt/ext4
root@luand-virtual-machine:/mnt/ext4# ls
root@luand-virtual-machine:/mnt/ext4#

```

Como se puede evidenciar, no hay nada dentro del directorio, pero si montamos la partición y realizamos el proceso nuevamente, *Oooh sorpresa*, allí están nuestros archivos

```

root@luand-virtual-machine: /mnt/ext4
root@luand-virtual-machine:~# umount /dev/sdb1 /mnt/ext4
umount: /mnt/ext4: not mounted.
root@luand-virtual-machine:~# cd /mnt/ext4
root@luand-virtual-machine:/mnt/ext4# ls
root@luand-virtual-machine:/mnt/ext4# cd
root@luand-virtual-machine:~# mount /dev/sdb1 /mnt/ext4
root@luand-virtual-machine:~# cd /mnt/ext4
root@luand-virtual-machine:/mnt/ext4# ls
lost+found  pruebaMount.txt
root@luand-virtual-machine:/mnt/ext4#

```