

Tarea SSF10

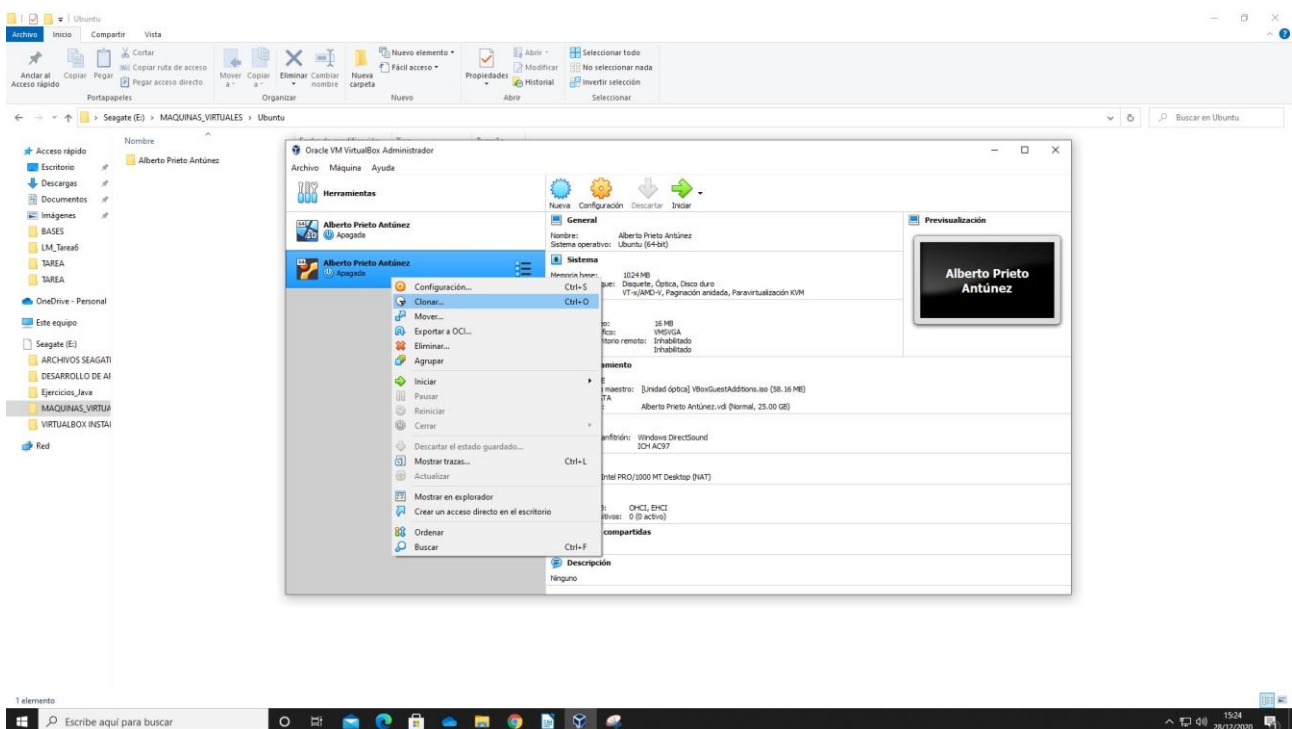
Curso: 1º CFGS de Desarrollo de Aplicaciones Web a distancia

Asignatura: Sistemas Informáticos

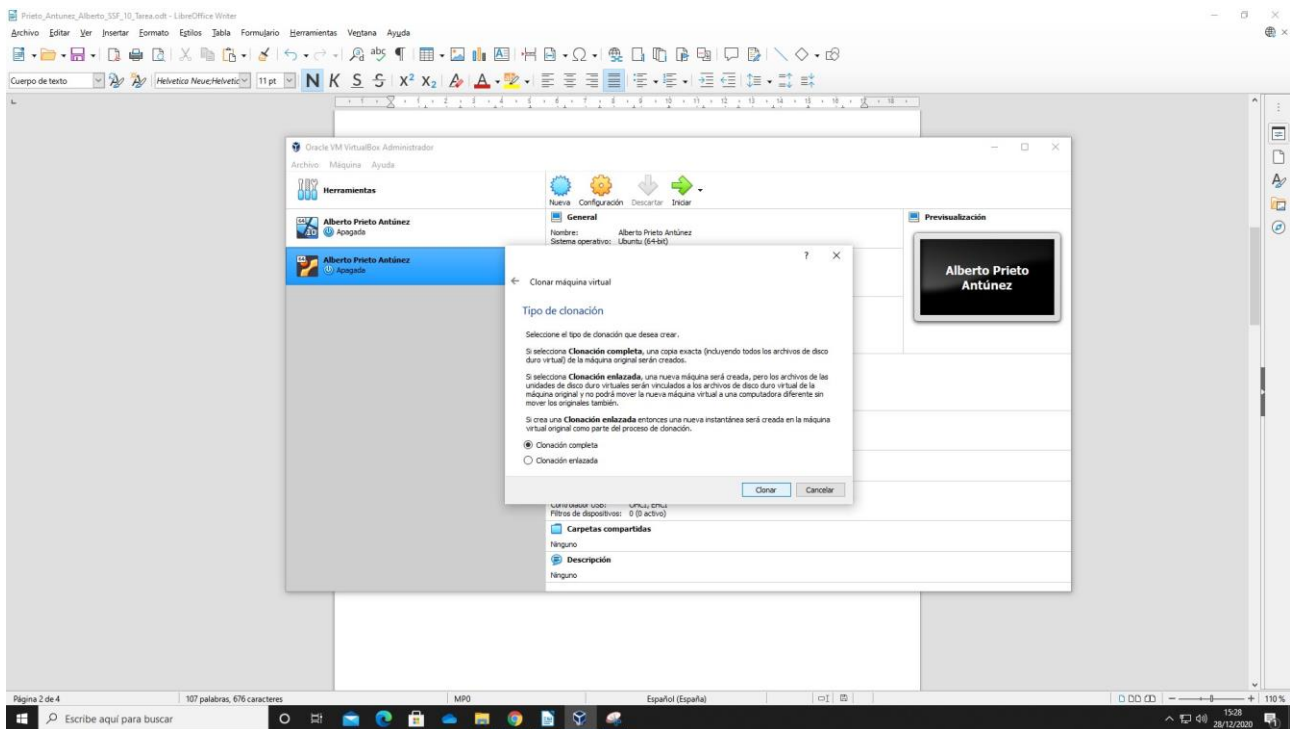
Unidad 10. Administración de la red (Linux III).

Ejercicio 1: Seguir los pasos del Punto 1.1 de los contenidos de la unidad, para configurar 2 máquinas en Linux. (ver vídeo).

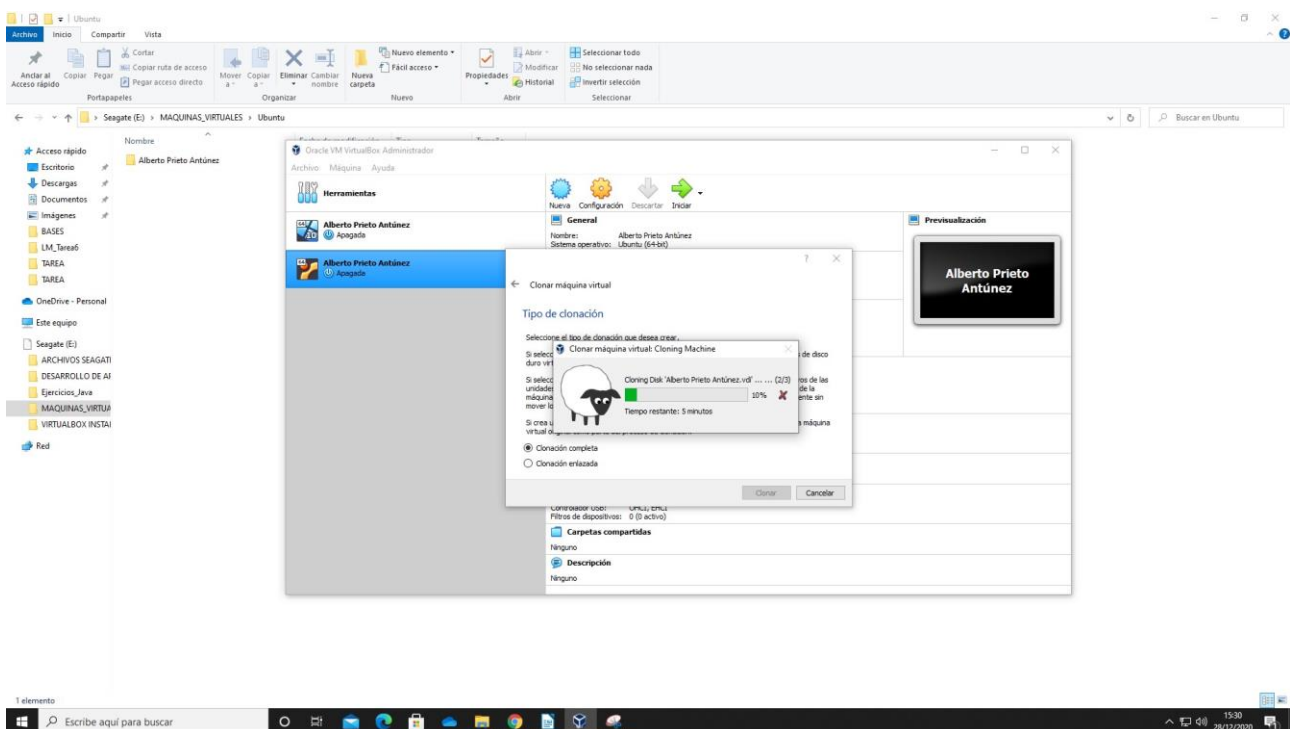
Para configurar 2 máquinas en Linux, lo primero que se realizará será clonar la máquina Linux, con VirtualBox, utilizada en unidades anteriores. La clonación se hará completa y reiniciando la dirección MAC. Para ello, haremos click, derecho en la máquina virtual Linux y daremos en la opción clonar.



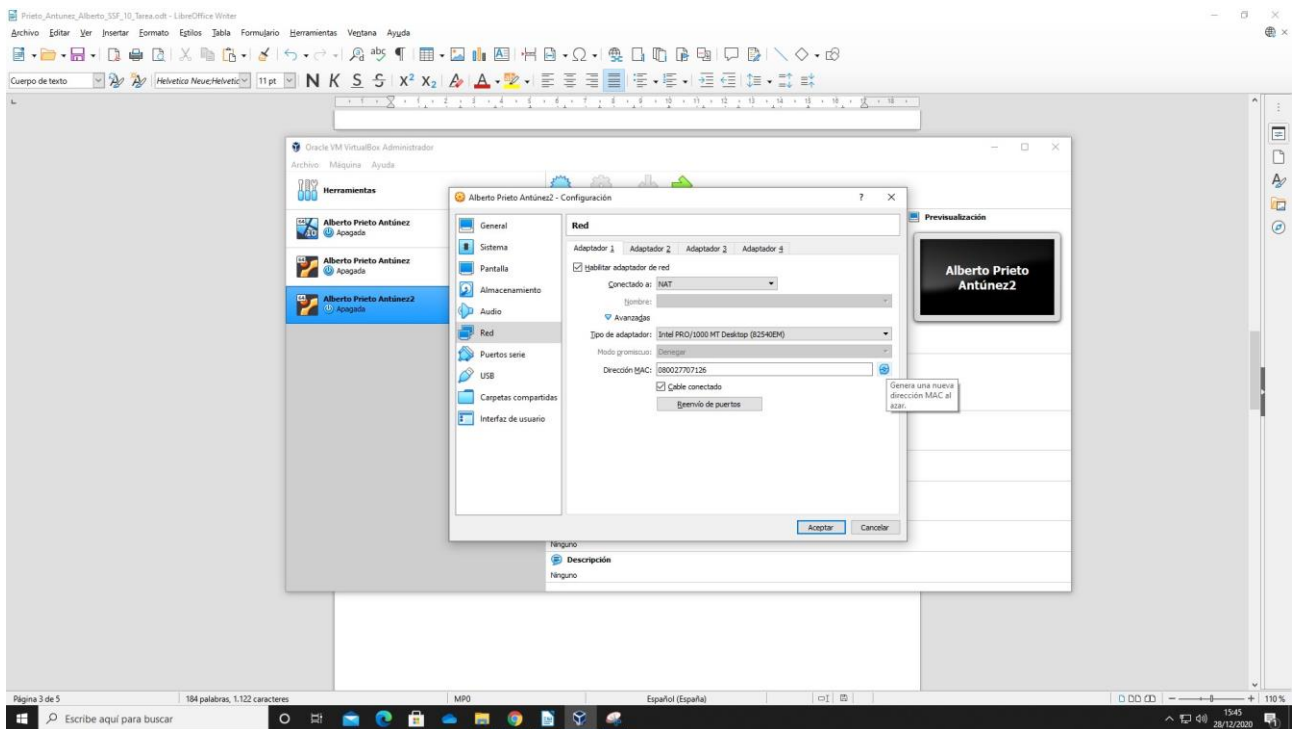
Se establecerá el nuevo nombre de la máquina virtual y se seleccionará la clonación completa. Una vez hecho esto, haremos click en clonar y se procederá a la clonación.



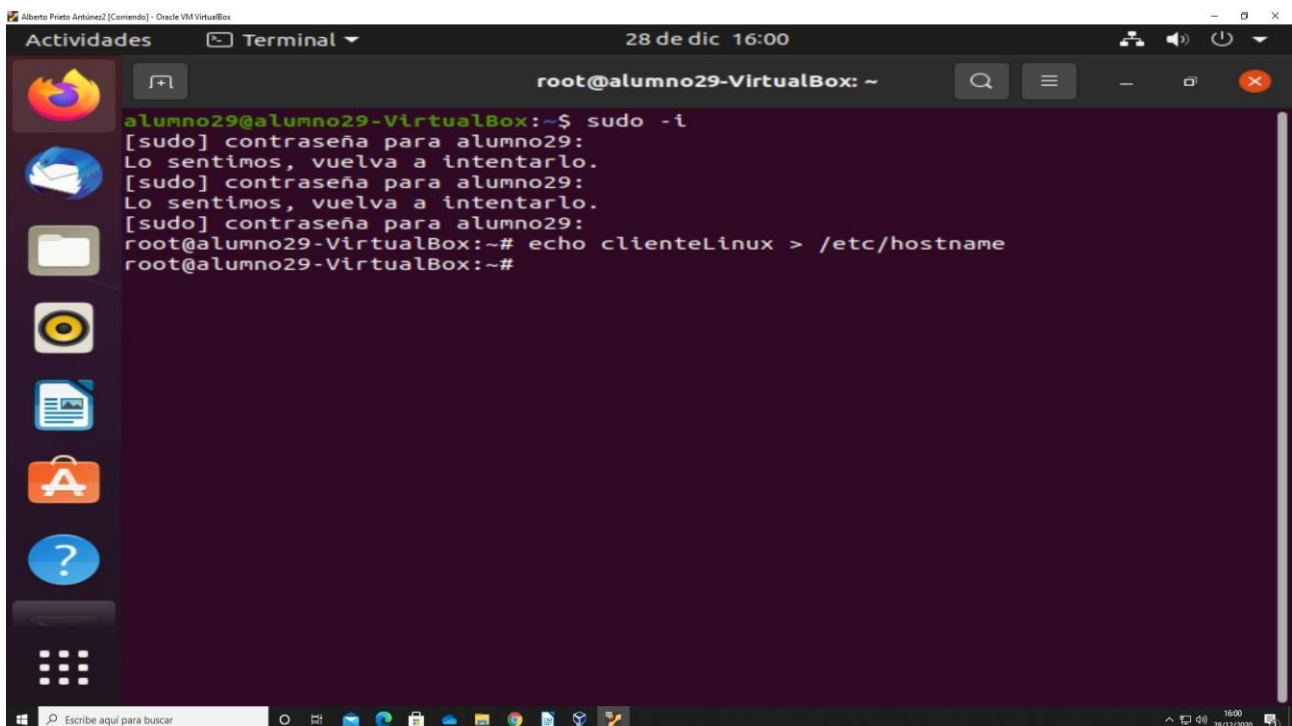
El proceso tardará unos minutos.



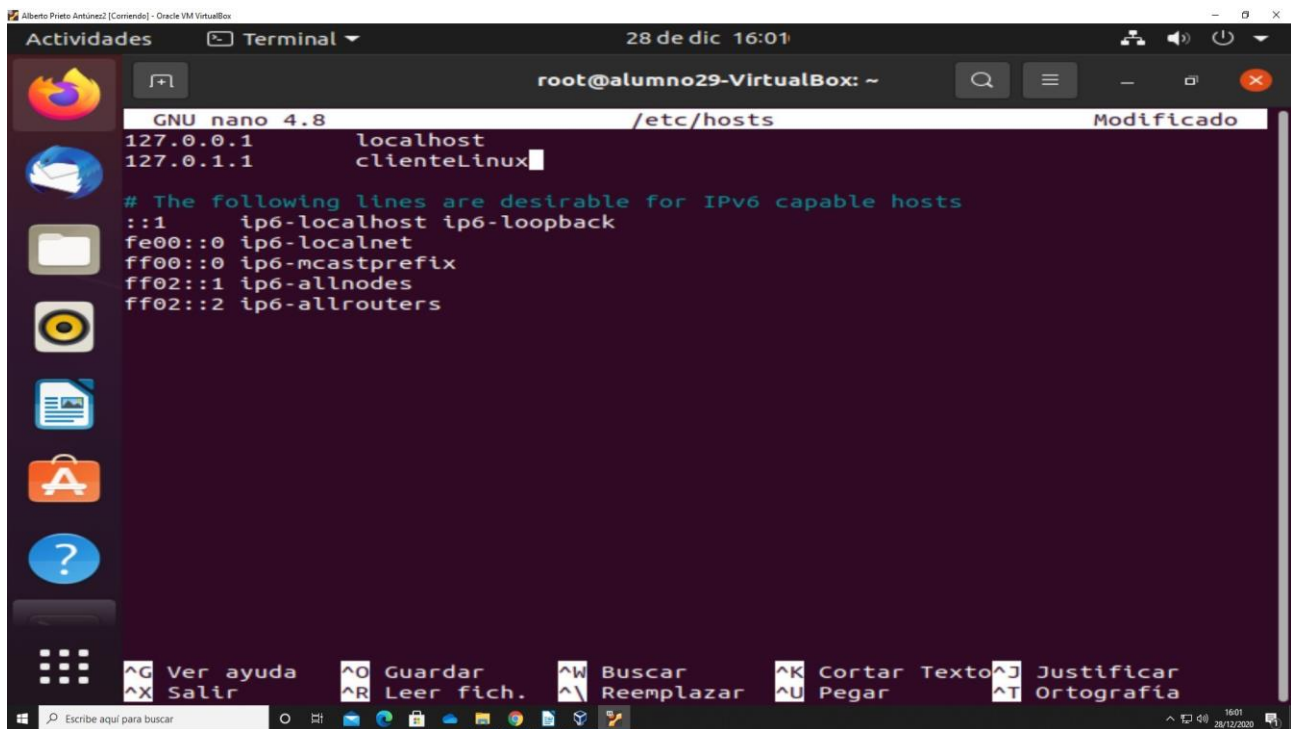
Una vez completada la clonación, lo que haremos será reiniciar la dirección MAC de la segunda máquina virtual Linux, para ello, iremos a la configuración de red y en la casilla **Dirección MAC** pulsamos el botón de generar una dirección MAC al azar.



Lo siguiente será cambiar el nombre de la máquina clonada a **clienteLinux**, para ello, abrimos un terminal y en modo administrador ejecutamos `echo clienteLinux > /etc/hostname`.



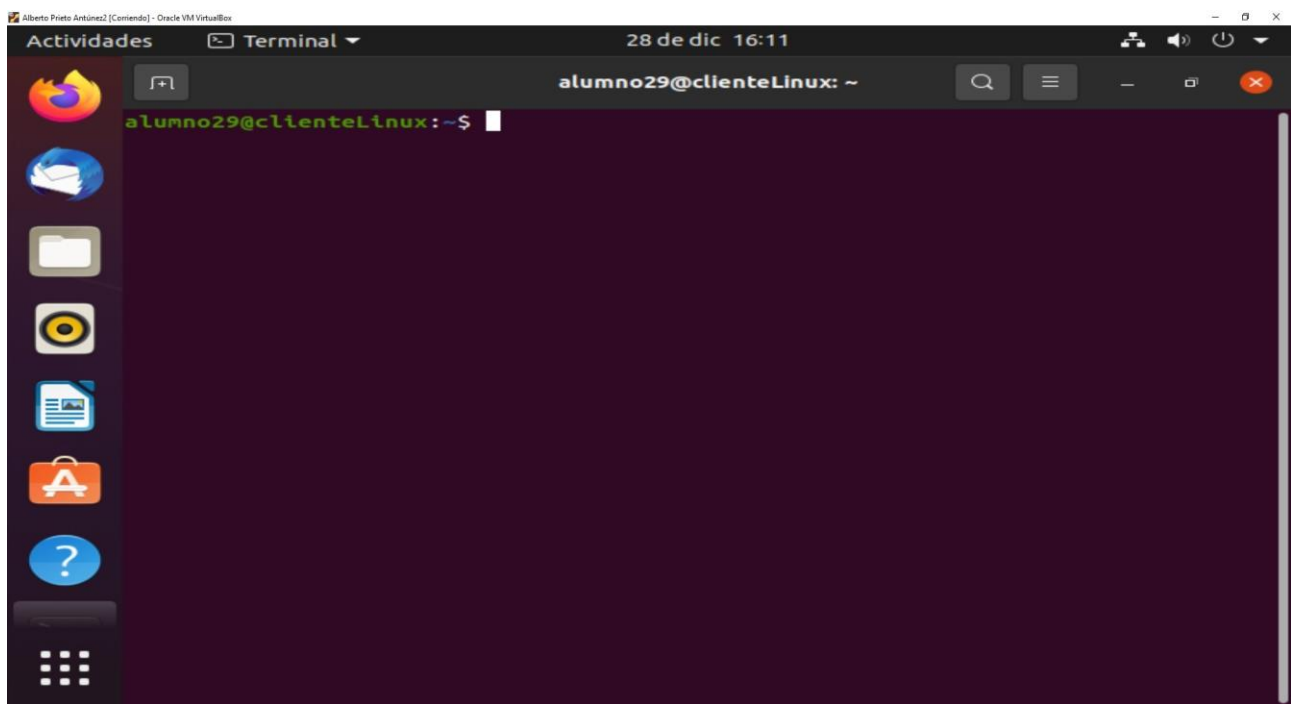
A continuación, editamos el archivo `/etc/hosts` y cambiamos el nombre de la máquina, para ello ejecutamos `nano /etc/hosts` y cambiamos el nombre a **clienteLinux**.



```
GNU nano 4.8 /etc/hosts Modificado
127.0.0.1 localhost
127.0.0.1 clienteLinux

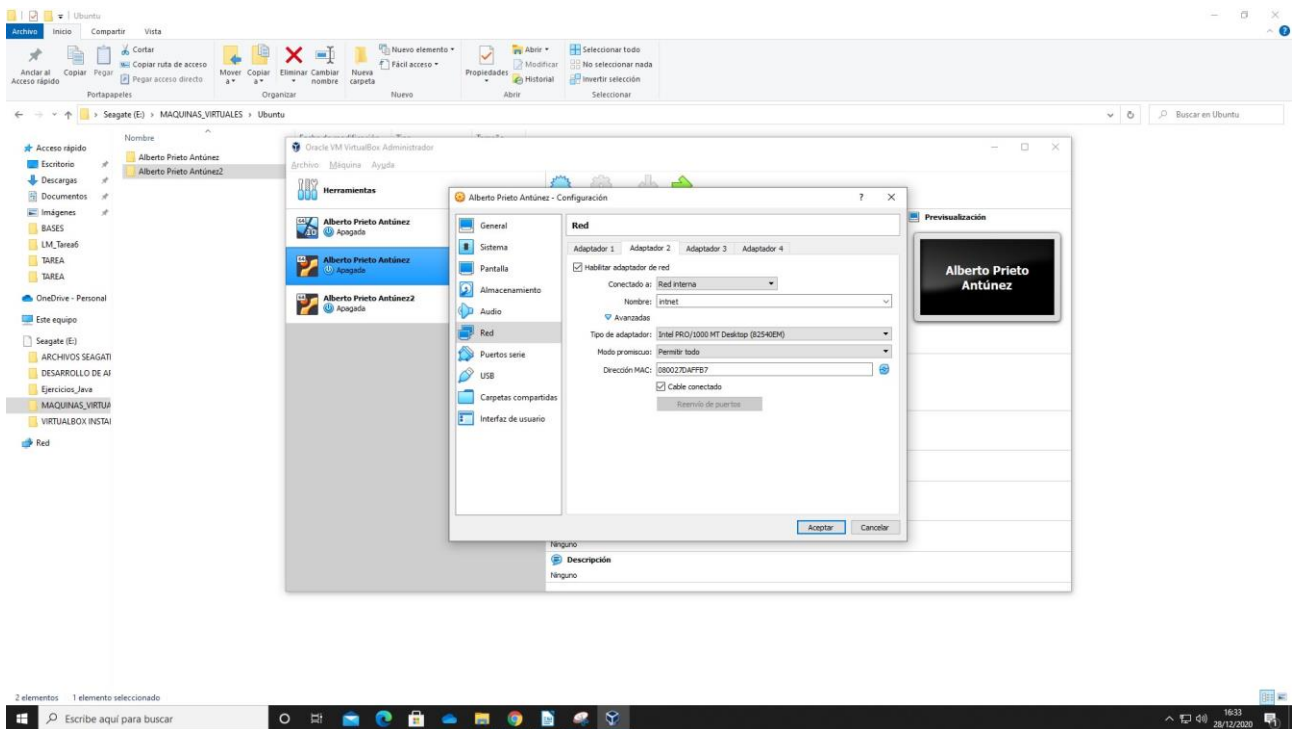
# The following lines are desirable for IPv6 capable hosts
::1 ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

Para finalizar hemos tenido que ejecutar el comando `sudo hostname clienteLinux` y reiniciar el terminal para que se efectúen los cambios.



```
alumno29@clienteLinux: ~$
```

Lo siguiente será configurar la red en la máquina virtual Linux servidor. Para ello, dejaremos el primer adaptador de red como hasta ahora, en NAT con la máquina anfitrión. Lo segundo será habilitar el adaptador 2 para ponerlo en la red interna.



Después iniciamos la máquina virtual Linux servidor y accederemos a los ficheros de configuración de interfaces. Para ello, veremos sus nombres en `/sys/class/net` y editaremos el fichero **01-network-manager-all.yaml** dentro de `/etc/netplan`.

```

alumno29@alumno29-VirtualBox:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
    inet6 fe80::31cc:57bacb:a1d6:60b prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:2a:34:89 txqueuelen 1000 (Ethernet)
    RX packets 1714 bytes 1768368 (1.7 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 693 bytes 67771 (67.7 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

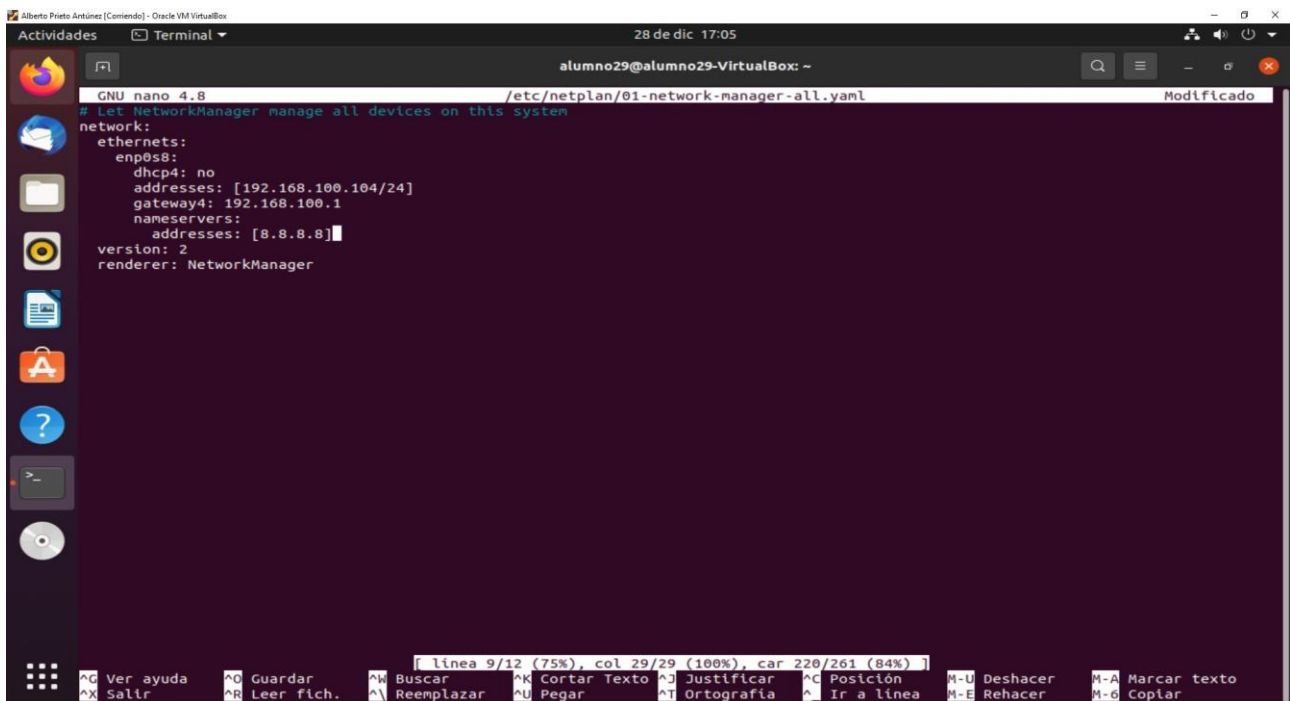
enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    ether 08:00:27:da:ff:b7 txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 101 bytes 15949 (15.9 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Bucle local)
    RX packets 313 bytes 27955 (27.9 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 313 bytes 27955 (27.9 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

alumno29@alumno29-VirtualBox:~$ ls /sys/class/net
enp0s3 enp0s8 lo
alumno29@alumno29-VirtualBox:~$ ls -l /etc/netplan/
total 4
-rw-r--r-- 1 root root 104 jul 31 17:35 01-network-manager-all.yaml
alumno29@alumno29-VirtualBox:~$ cat /etc/netplan/01-network-manager-all.yaml
# Let NetworkManager manage all devices on this system
network:
  version: 2
  renderer: NetworkManager
alumno29@alumno29-VirtualBox:~$

```

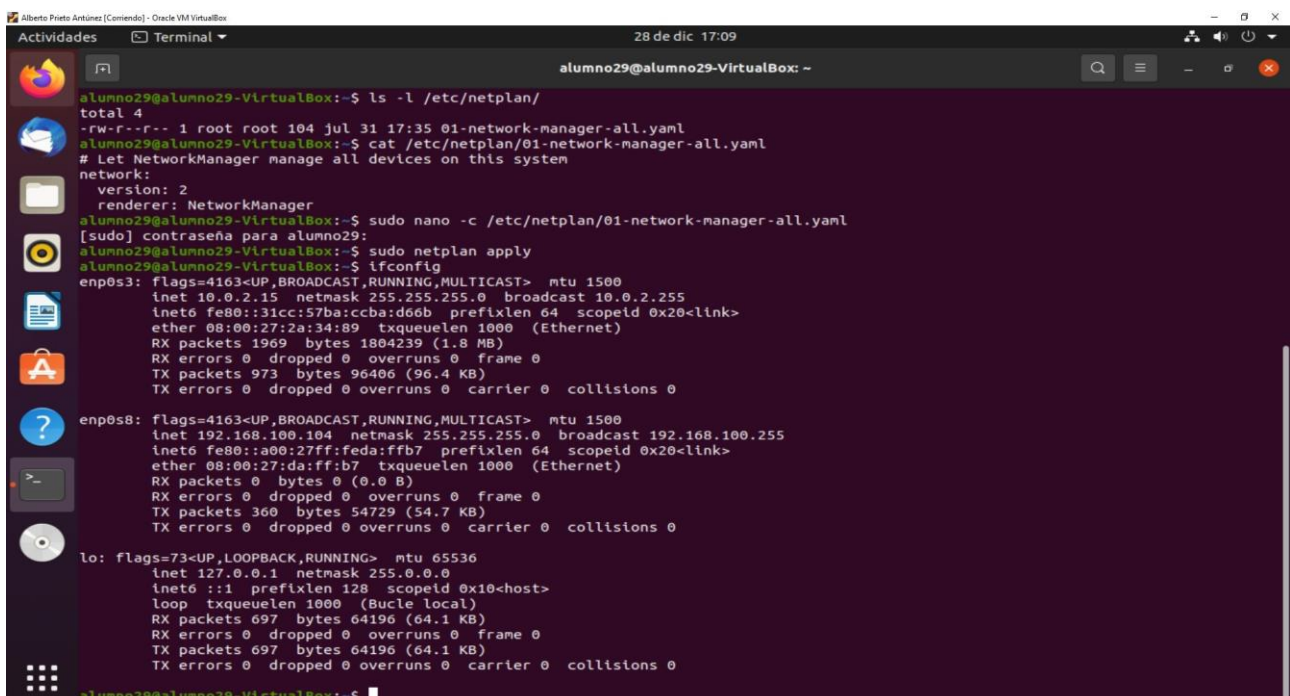
Para modificar el fichero, haremos, como administrador, `sudo nano -c /etc/netplan/01-networkmanager-all.yaml`.



```
alumno29@alumno29-VirtualBox: ~  
GNU nano 4.8 /etc/netplan/01-network-manager-all.yaml  
# Let NetworkManager manage all devices on this system  
network:  
  ethernets:  
    enp0s8:  
      dhcp4: no  
      addresses: [192.168.100.104/24]  
      gateway4: 192.168.100.1  
      nameservers:  
        addresses: [8.8.8.8]  
      version: 2  
      renderer: NetworkManager
```

Una vez realizada la modificación, se pulsa **Ctrl + O** para guardar (pulsar **Enter**) y **Ctrl + X** para salir. Para finalizar, se ejecuta el comando `sudo netplan apply` para aplicar los cambios.

Lo siguiente, es ejecutar el comando `ifconfig` para ver la configuración de red y verificar que hemos realizado la configuración adecuadamente.



```
alumno29@alumno29-VirtualBox: ~  
alumno29@alumno29-VirtualBox:~$ ls -l /etc/netplan/  
total 4  
-rw-r--r-- 1 root root 104 jul 31 17:35 01-network-manager-all.yaml  
alumno29@alumno29-VirtualBox:~$ cat /etc/netplan/01-network-manager-all.yaml  
# Let NetworkManager manage all devices on this system  
network:  
  version: 2  
  renderer: NetworkManager  
alumno29@alumno29-VirtualBox:~$ sudo nano -c /etc/netplan/01-network-manager-all.yaml  
[sudo] contraseña para alumno29:  
alumno29@alumno29-VirtualBox:~$ sudo netplan apply  
alumno29@alumno29-VirtualBox:~$ ifconfig  
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255  
    inet6 fe80::31cc:57ba:ccb8:d60b prefixlen 64 scopeid 0x20<link>  
    ether 08:00:27:2a:34:89 txqueuelen 1000 (Ethernet)  
    RX packets 1969 bytes 1804239 (1.8 MB)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 973 bytes 96406 (96.4 KB)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
    inet 192.168.100.104 netmask 255.255.255.0 broadcast 192.168.100.255  
    inet6 fe80::a00:27ff:feda:ffb7 prefixlen 64 scopeid 0x20<link>  
    ether 08:00:27:da:ff:b7 txqueuelen 1000 (Ethernet)  
    RX packets 0 bytes 0 (0.0 B)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 360 bytes 54729 (54.7 KB)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536  
    inet 127.0.0.1 netmask 255.0.0.0  
    inet6 ::1 prefixlen 128 scopeid 0x10<host>  
    loop txqueuelen 1000 (Bucle local)  
    RX packets 697 bytes 64196 (64.1 KB)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 697 bytes 64196 (64.1 KB)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
alumno29@alumno29-VirtualBox:~$
```

Para finalizar la configuración en el servidor Linux, comprobaremos que hay conectividad de red en el propio equipo ejecutando `ping 192.168.100.104` y que además, hay conexión a Internet haciendo `ping 8.8.8.8` (DNS de Google).

```
Albeto Prieto Ant6nez [Conenido] - Oracle VM VirtualBox
Actividades Terminal 28 de dic 17:11
alumno29@alumno29-VirtualBox: ~

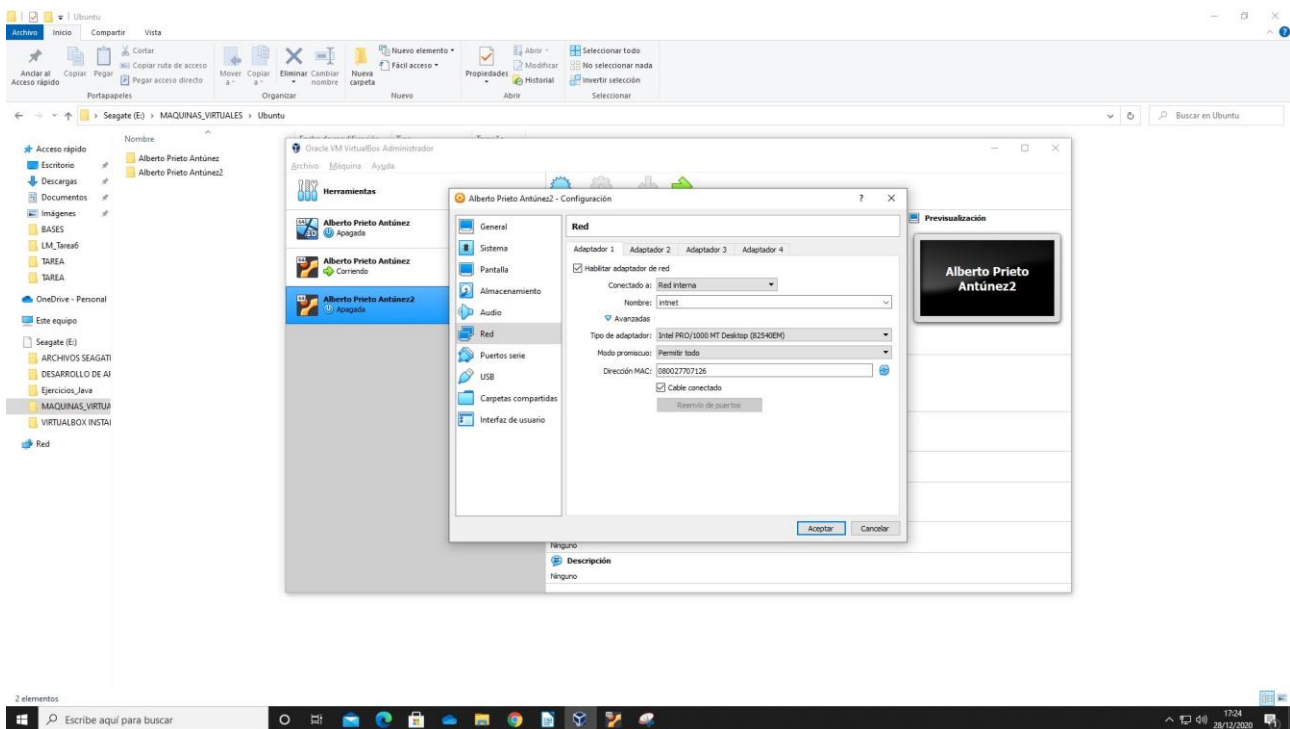
ether 08:00:27:2a:34:89 txqueuelen 1000 (Ethernet)
RX packets 1969 bytes 1804239 (1.8 MB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 973 bytes 96406 (96.4 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
inet 192.168.100.104 netmask 255.255.255.0 broadcast 192.168.100.255
inet6 fe80::a00:27ff:feda:ffb7 prefixlen 64 scopeid 0x20<link>
ether 08:00:27:da:ff:b7 txqueuelen 1000 (Ethernet)
RX packets 0 bytes 0 (0.0 B)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 360 bytes 54729 (54.7 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

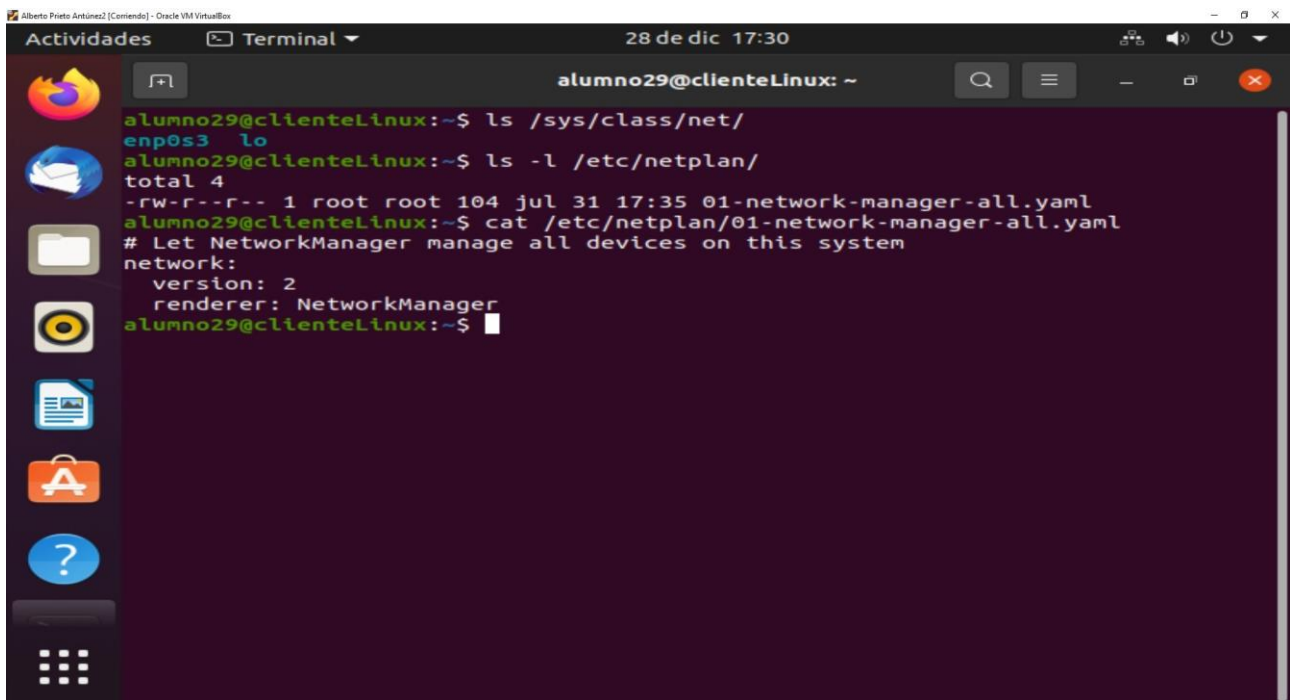
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
inet 127.0.0.1 netmask 255.0.0.0
inet6 ::1 prefixlen 128 scopeid 0x10<host>
loop txqueuelen 1000 (Bucle local)
RX packets 697 bytes 64196 (64.1 KB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 697 bytes 64196 (64.1 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

alumno29@alumno29-VirtualBox:~$ ping 192.168.100.104
PING 192.168.100.104 (192.168.100.104) 56(84) bytes of data.
64 bytes from 192.168.100.104: icmp_seq=1 ttl=64 time=0.017 ms
64 bytes from 192.168.100.104: icmp_seq=2 ttl=64 time=0.060 ms
64 bytes from 192.168.100.104: icmp_seq=3 ttl=64 time=0.059 ms
^Z
[1]+ Detenido ping 192.168.100.104
alumno29@alumno29-VirtualBox:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=118 time=29.8 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=118 time=31.4 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=118 time=30.8 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=118 time=30.5 ms
^Z
[2]+ Detenido ping 8.8.8.8
alumno29@alumno29-VirtualBox:~$
```

A continuaci3n, procedemos a configurar la m1quina **clienteLinux**. Esta m1quina solo tiene una tarjeta de red (interna en VirtualBox).



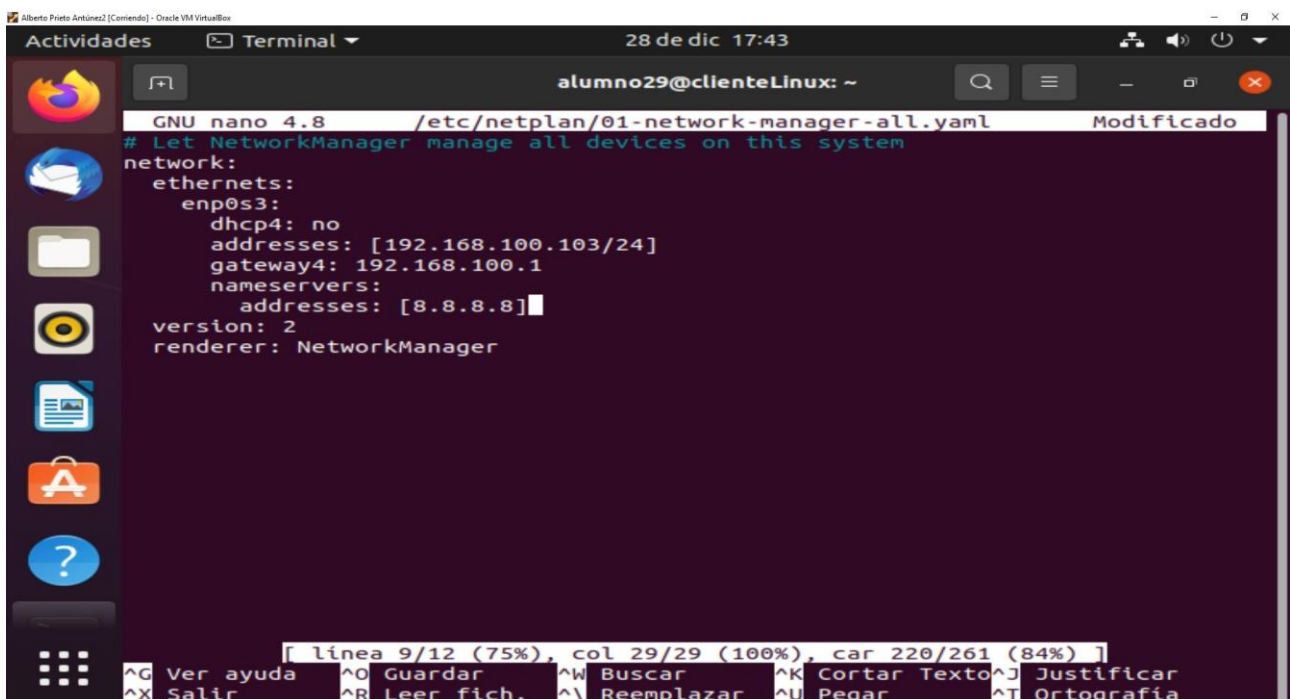
Iniciamos la m1quina virtual y realizamos los mismos pasos que en la m1quina servidor.



A terminal window titled 'alumno29@clienteLinux: ~' showing the following commands and output:

```
alumno29@clienteLinux:~$ ls /sys/class/net/
enp0s3  lo
alumno29@clienteLinux:~$ ls -l /etc/netplan/
total 4
-rw-r--r-- 1 root root 104 jul 31 17:35 01-network-manager-all.yaml
alumno29@clienteLinux:~$ cat /etc/netplan/01-network-manager-all.yaml
# Let NetworkManager manage all devices on this system
network:
  version: 2
  renderer: NetworkManager
alumno29@clienteLinux:~$
```

Volvemos a ejecutar `sudo nano -c /etc /netplan /01-network-manager-all.yaml` y realizamos la siguiente configuración. Volvemos a aplicar acambios.



A terminal window showing the nano editor editing the file `/etc/netplan/01-network-manager-all.yaml`. The editor title bar indicates 'GNU nano 4.8' and 'Modificado'. The content of the file is:

```
# Let NetworkManager manage all devices on this system
network:
  ethernet:
    enp0s3:
      dhcp4: no
      addresses: [192.168.100.103/24]
      gateway4: 192.168.100.1
      nameservers:
        addresses: [8.8.8.8]
  version: 2
  renderer: NetworkManager
```

The status bar at the bottom shows: `[linea 9/12 (75%), col 29/29 (100%), car 220/261 (84%)]`. Below the status bar is a menu with the following options:

Ver ayuda	Guardar	Buscar	Cortar Texto	Justificar
Salir	Leer fich.	Reemplazar	Pegar	Ortografia

Una vez hecho esto, ejecutamos `ifconfig` para ver la configuración de red y establecemos conexión con el servidor Linux a través de `ping 192.168.100.104` y vemos que tenemos conexión con la máquina Linux servidor.


```
Actividades Terminal 28 de dic 17:38
alumno29@clienteLinux: ~
alumno29@clienteLinux:~$ sudo nano -c /etc/netplan/01-network-manager-all.yaml
alumno29@clienteLinux:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
inet 192.168.100.103 netmask 255.255.255.0 broadcast 192.168.100.255
inet6 fe80::a00:27ff:fe70:7126 prefixlen 64 scopeid 0x20<link>
ether 08:00:27:70:71:26 txqueuelen 1000 (Ethernet)
RX packets 12 bytes 720 (720.0 B)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 208 bytes 32996 (32.9 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
inet 127.0.0.1 netmask 255.0.0.0
inet6 ::1 prefixlen 128 scopeid 0x10<host>
loop txqueuelen 1000 (Bucle local)
RX packets 1836 bytes 134376 (134.3 KB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 1836 bytes 134376 (134.3 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

alumno29@clienteLinux:~$ ping 192.168.100.104
PING 192.168.100.104 (192.168.100.104) 56(84) bytes of data.
64 bytes from 192.168.100.104: icmp_seq=1 ttl=64 time=0.789 ms
64 bytes from 192.168.100.104: icmp_seq=2 ttl=64 time=0.975 ms
64 bytes from 192.168.100.104: icmp_seq=3 ttl=64 time=0.905 ms
64 bytes from 192.168.100.104: icmp_seq=4 ttl=64 time=0.363 ms
^Z
[1]+ Detenido ping 192.168.100.104
alumno29@clienteLinux:~$
```

Sin embargo, si tratamos de hacer *ping 8.8.8.8* vemos que no tenemos comunicación ya que la máquina virtual servidor no está configurado como router, por lo que no comunica las dos redes.

```
Actividades Terminal 28 de dic 17:45
alumno29@clienteLinux: ~
RX packets 31 bytes 2200 (2.2 KB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 354 bytes 45362 (45.3 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
inet 127.0.0.1 netmask 255.0.0.0
inet6 ::1 prefixlen 128 scopeid 0x10<host>
loop txqueuelen 1000 (Bucle local)
RX packets 3049 bytes 223345 (223.3 KB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 3049 bytes 223345 (223.3 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

alumno29@clienteLinux:~$ ping 192.168.100.104
PING 192.168.100.104 (192.168.100.104) 56(84) bytes of data.
64 bytes from 192.168.100.104: icmp_seq=1 ttl=64 time=0.314 ms
64 bytes from 192.168.100.104: icmp_seq=2 ttl=64 time=0.842 ms
64 bytes from 192.168.100.104: icmp_seq=3 ttl=64 time=0.896 ms
^Z
[4]+ Detenido ping 192.168.100.104
alumno29@clienteLinux:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
From 192.168.100.103 icmp_seq=1 Destination Host Unreachable
From 192.168.100.103 icmp_seq=2 Destination Host Unreachable
From 192.168.100.103 icmp_seq=3 Destination Host Unreachable
^Z
[5]+ Detenido ping 8.8.8.8
alumno29@clienteLinux:~$
```

Ejercicio 2: Seguir los pasos del Punto 2 de los contenidos de la unidad. Instalar Samba, configurar y compartir los mismos recursos que hay en los apuntes. Realizar la conexión desde máquina Windows o desde otra máquina cliente Linux.

En primer lugar, vamos a instalar SAMBA en la máquina Linux servidor, para ello, en un terminal ejecutamos *sudo apt install samba samba-common-bin* e introducimos la contraseña.

```
Alberto Prieto Antón [Comando] - Oracle VM VirtualBox
Actividades Terminal 28 de dic 21:15
alumno29@alumno29-VirtualBox: ~

Configurando attr (1:2.4.48-5) ...
Configurando samba-vfs-modules:amd64 (2:4.11.6+dfsg-0ubuntu1.6) ...
Configurando samba-common (2:4.11.6+dfsg-0ubuntu1.6) ...

Creating config file /etc/samba/smb.conf with new version
Configurando python3-tdb (1.4.2-3build1) ...
Configurando python3-pygments (2.3.1+dfsg-1ubuntu2) ...
Configurando python3-pyparsing (2.4.6-1) ...
Configurando python3-gpg (1.13.1-7ubuntu2) ...
Configurando tdb-tools (1.4.2-3build1) ...
update-alternatives: utilizando /usr/bin/tdbbackup.tdbtools para proveer /usr/bin/tdbbackup (tdbbackup) en modo automático
Configurando python3-ldb (2:2.0.10-0ubuntu0.20.04.2) ...
Configurando python3-dnspython (1.16.0-1build1) ...
Configurando python3-crypto (2.6.1-13ubuntu2) ...
Configurando samba-dsdb-modules:amd64 (2:4.11.6+dfsg-0ubuntu1.6) ...
Configurando python3-samba (2:4.11.6+dfsg-0ubuntu1.6) ...
Configurando librdmacm1:amd64 (28.0-1ubuntu1) ...
Configurando librados2 (15.2.7-0ubuntu0.20.04.1) ...
Configurando libcephfs2 (15.2.7-0ubuntu0.20.04.1) ...
Configurando python3-packaging (20.3-1) ...
Configurando python3-markdown (3.1.1-3) ...
Configurando samba-common-bin (2:4.11.6+dfsg-0ubuntu1.6) ...
checking smb.conf with testparm
Load smb config files from /etc/samba/smb.conf
Loaded services file OK.
WARNING: The 'netbios name' is too long (max. 15 chars).
Server role: ROLE_STANDALONE

Done
Configurando samba (2:4.11.6+dfsg-0ubuntu1.6) ...
Samba is not being run as an AD Domain Controller: Masking samba-ad-dc.service
Please ignore the following error about deb-systemd-helper not finding those services.
(samba-ad-dc.service masked)
Created symlink /etc/systemd/system/multi-user.target.wants/nmbd.service → /lib/systemd/system/nmbd.service.
Failed to preset unit: Unit file /etc/systemd/system/samba-ad-dc.service is masked.
/usr/bin/deb-systemd-helper: error: systemctl preset failed on samba-ad-dc.service: No such file or directory
Created symlink /etc/systemd/system/multi-user.target.wants/smbd.service → /lib/systemd/system/smbd.service.

Progreso: 98% [#####...]
```

Una vez instalado, hay que comprobar si está activado, para ello, ejecutamos *systemctl status smbd* y *systemctl status nmbd*.

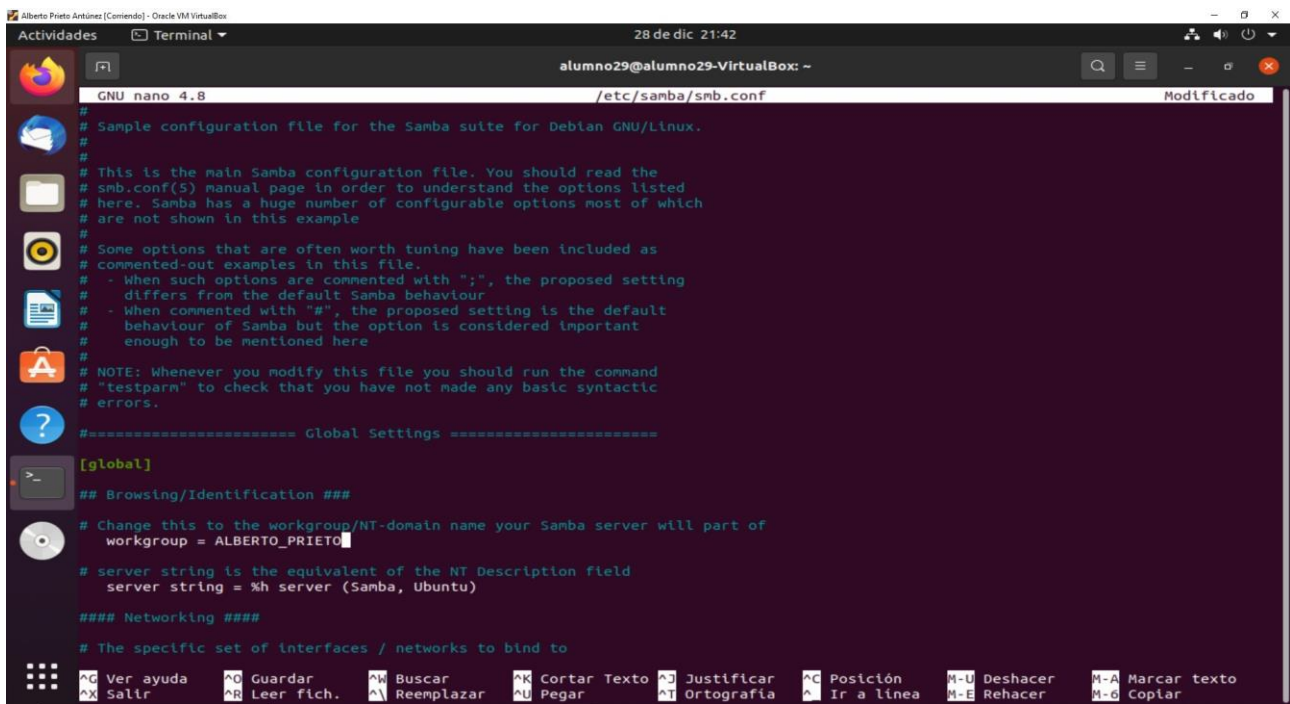
```
Alberto Prieto Antón [Comando] - Oracle VM VirtualBox
Actividades Terminal 28 de dic 21:22
alumno29@alumno29-VirtualBox: ~

alumno29@alumno29-VirtualBox:~$ systemctl smbd status
Unknown operation smbd.
alumno29@alumno29-VirtualBox:~$ systemctl status smbd
● smbd.service - Samba SMB Daemon
   Loaded: loaded (/lib/systemd/system/smbd.service; enabled; vendor preset: enabled)
   Active: active (running) since Mon 2020-12-28 21:15:44 WET; 3min 43s ago
     Docs: man:smbd(8)
           man:samba(7)
           man:smb.conf(5)
    Main PID: 4628 (smbd)
      Status: "smbd: ready to serve connections..."
        Tasks: 4 (limit: 1110)
       Memory: 12.3M
    CGroup: /system.slice/smbd.service
            └─4628 /usr/sbin/smbd --foreground --no-process-group
              4632 /usr/sbin/smbd --foreground --no-process-group
              4633 /usr/sbin/smbd --foreground --no-process-group
              4634 /usr/sbin/smbd --foreground --no-process-group

dic 28 21:15:43 alumno29-VirtualBox systemd[1]: Starting Samba SMB Daemon...
dic 28 21:15:44 alumno29-VirtualBox update-apparmor-samba-profile[4621]: grep: /etc/apparmor.d/samba/smbd-shares: No existe el arch
dic 28 21:15:44 alumno29-VirtualBox update-apparmor-samba-profile[4625]: diff: /etc/apparmor.d/samba/smbd-shares: No existe el arch
dic 28 21:15:44 alumno29-VirtualBox systemd[1]: Started Samba SMB Daemon.
alumno29@alumno29-VirtualBox:~$ systemctl status nmbd
● nmbd.service - Samba NMB Daemon
   Loaded: loaded (/lib/systemd/system/nmbd.service; enabled; vendor preset: enabled)
   Active: active (running) since Mon 2020-12-28 21:15:43 WET; 6min ago
     Docs: man:nmbd(8)
           man:samba(7)
           man:smb.conf(5)
    Main PID: 4617 (nmbd)
      Status: "nmbd: ready to serve connections..."
        Tasks: 1 (limit: 1110)
       Memory: 3.3M
    CGroup: /system.slice/nmbd.service
            └─4617 /usr/sbin/nmbd --foreground --no-process-group

dic 28 21:15:43 alumno29-VirtualBox systemd[1]: Starting Samba NMB Daemon...
dic 28 21:15:43 alumno29-VirtualBox systemd[1]: Started Samba NMB Daemon.
alumno29@alumno29-VirtualBox:~$
```

Lo siguiente será modificar el archivo */etc/samba/smb.conf*. Añadiremos un grupo de trabajo que llamaremos de la misma forma que llamamos al grupo de trabajo de Windows (Nombre_Apellido): ALBERTO_PRIETO. Para ello, ejecutamos *sudo nano /etc/samba/smb.conf*.



```
GNU nano 4.8 /etc/samba/smb.conf Modificado
# Sample configuration file for the Samba suite for Debian GNU/Linux.
#
# This is the main Samba configuration file. You should read the
# smb.conf(5) manual page in order to understand the options listed
# here. Samba has a huge number of configurable options most of which
# are not shown in this example
#
# Some options that are often worth tuning have been included as
# commented-out examples in this file.
# - When such options are commented with ";", the proposed setting
# differs from the default Samba behaviour
# - When commented with "#", the proposed setting is the default
# behaviour of Samba but the option is considered important
# enough to be mentioned here
#
# NOTE: Whenever you modify this file you should run the command
# "testparm" to check that you have not made any basic syntactic
# errors.
# ===== Global Settings =====
[global]
## Browsing/Identification ###
# Change this to the workgroup/NT-domain name your Samba server will part of
workgroup = ALBERTO_PRIETO
# server string is the equivalent of the NT Description field
server string = %h server (Samba, Ubuntu)

#### Networking ####
# The specific set of interfaces / networks to bind to
```

En el final del fichero hay que añadir las siguientes líneas:

[publico] path =

/samba/lectura

browseable = yes

guest ok = yes read

only = yes

[escritura]

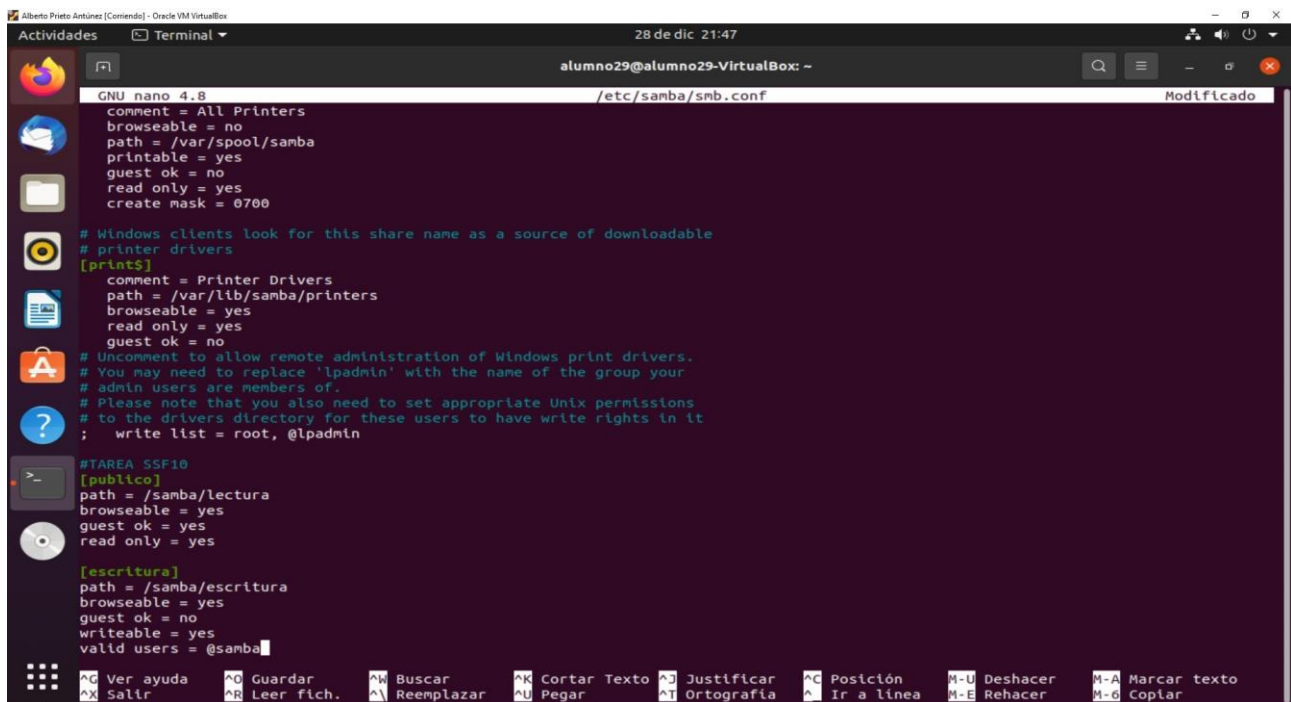
path = /samba/escritura

browseable = yes guest

ok = no writeable = yes

valid users = @samba

Una vez hecho esto, se guardan lo cambios y se sale de la edición del archivo.



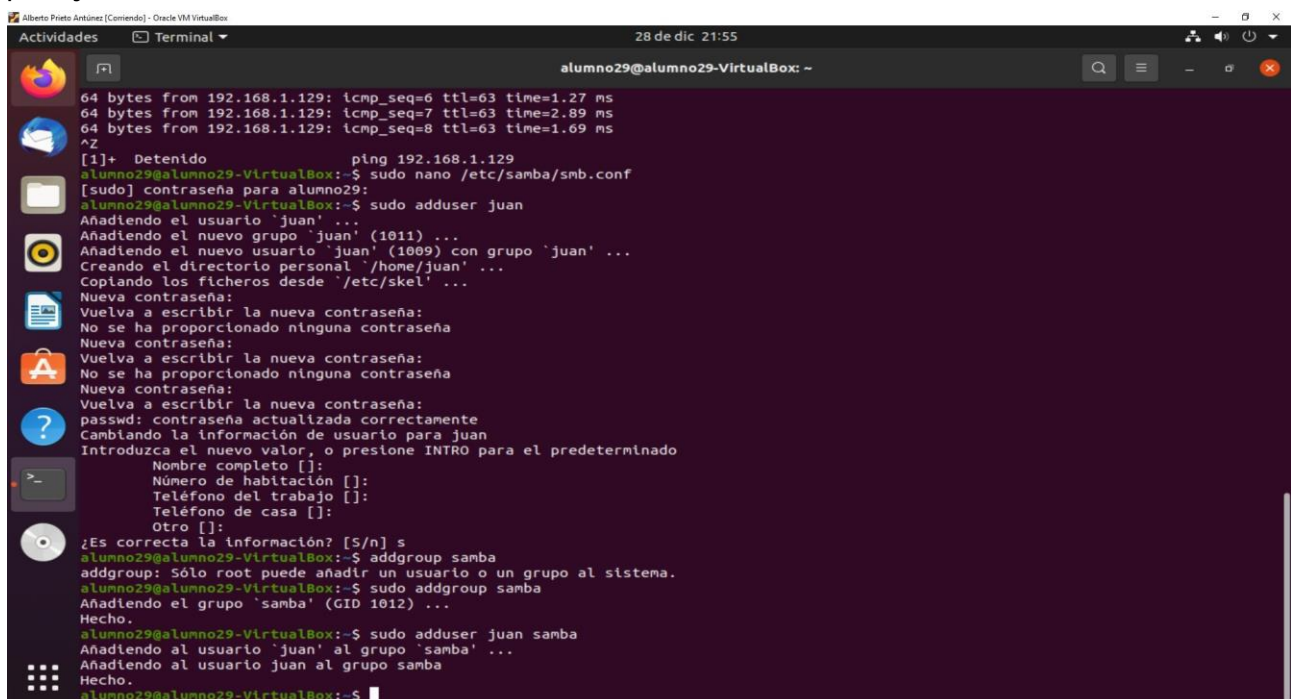
```
GNU nano 4.8 /etc/samba/smb.conf Modificado
comment = All Printers
browseable = no
path = /var/spool/samba
printable = yes
guest ok = no
read only = yes
create mask = 0700

# Windows clients look for this share name as a source of downloadable
# printer drivers
[print$]
comment = Printer Drivers
path = /var/lib/samba/printers
browseable = yes
read only = yes
guest ok = no
# Uncomment to allow remote administration of Windows print drivers.
# You may need to replace 'lpadmin' with the name of the group your
# admin users are members of.
# Please note that you also need to set appropriate Unix permissions
# to the drivers directory for these users to have write rights in it
; write list = root, @lpadmin

#TAREA SSF10
[publico]
path = /samba/lectura
browseable = yes
guest ok = yes
read only = yes

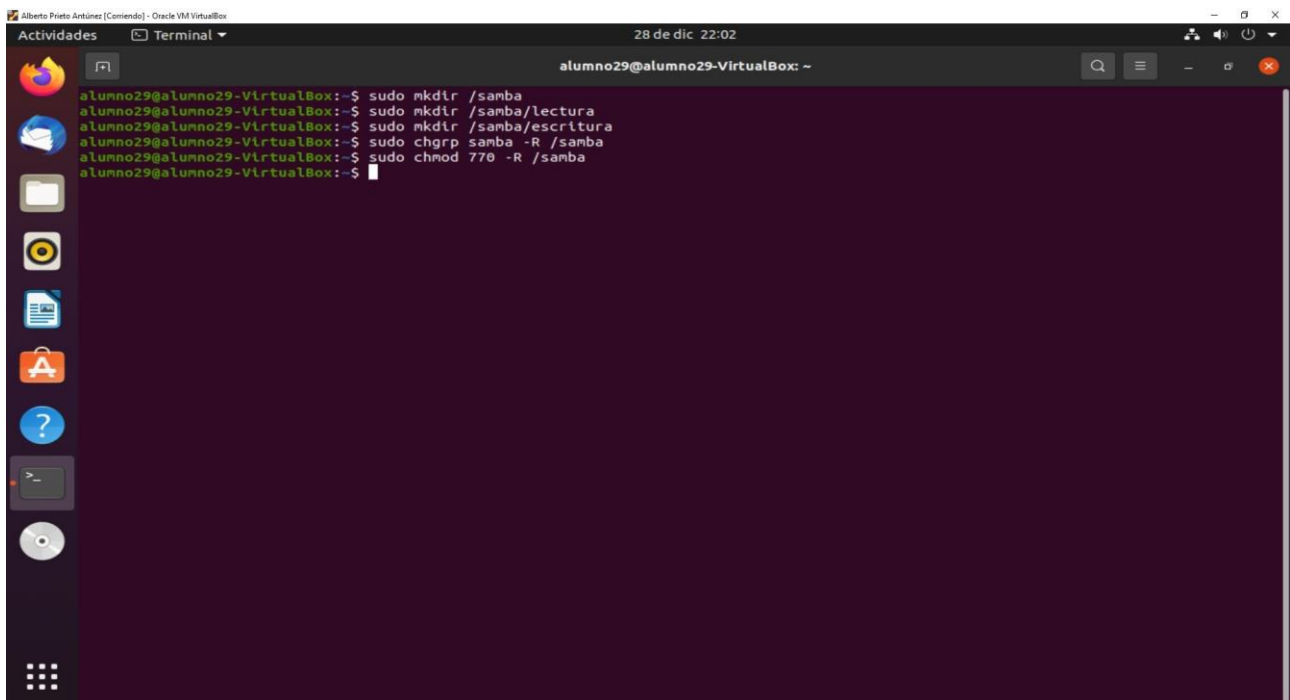
[escritura]
path = /samba/escritura
browseable = yes
guest ok = no
writeable = yes
valid users = @samba
```

Lo siguiente será crear usuarios, grupos, carpetas y permisos en la máquina servidor. En primer lugar, los usuarios **juan** y **juana** y el grupo **samba**. El procedimiento para **juan** es el mismo que para **juana**.



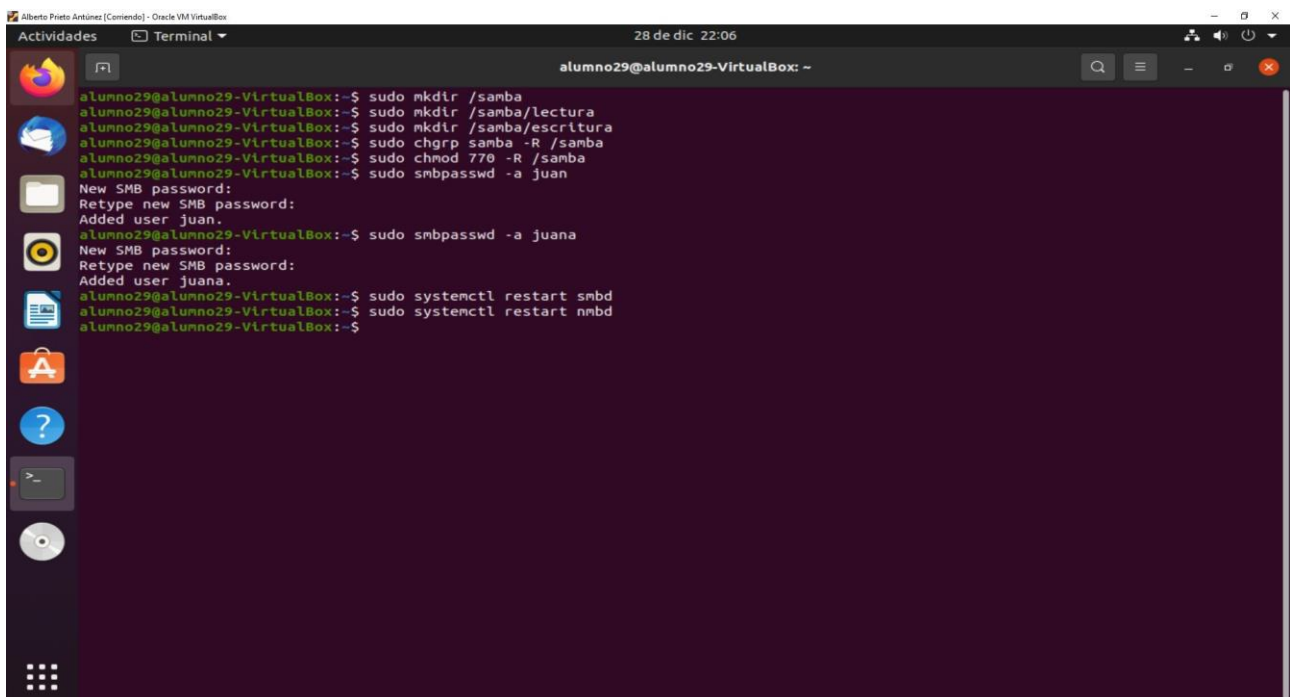
```
64 bytes from 192.168.1.129: icmp_seq=6 ttl=63 time=1.27 ms
64 bytes from 192.168.1.129: icmp_seq=7 ttl=63 time=2.89 ms
64 bytes from 192.168.1.129: icmp_seq=8 ttl=63 time=1.69 ms
^Z
[1]+  Detenido                  ping 192.168.1.129
alumno29@alumno29-VirtualBox:~$ sudo nano /etc/samba/smb.conf
[sudo] contraseña para alumno29:
alumno29@alumno29-VirtualBox:~$ sudo adduser juan
Añadiendo el usuario 'juan' ...
Añadiendo el nuevo grupo 'juan' (1011) ...
Añadiendo el nuevo usuario 'juan' (1009) con grupo 'juan' ...
Creando el directorio personal '/home/juan' ...
Copiando los ficheros desde '/etc/skel' ...
Nueva contraseña:
Vuelva a escribir la nueva contraseña:
No se ha proporcionado ninguna contraseña
Nueva contraseña:
Vuelva a escribir la nueva contraseña:
No se ha proporcionado ninguna contraseña
Nueva contraseña:
Vuelva a escribir la nueva contraseña:
passwd: contraseña actualizada correctamente
Cambiando la información de usuario para juan
Introduzca el nuevo valor, o presione INTRO para el predeterminado
Nombre completo []:
Número de habitación []:
Teléfono del trabajo []:
Teléfono de casa []:
Otro []:
¿Es correcta la información? [S/n] s
alumno29@alumno29-VirtualBox:~$ addgroup samba
addgroup: Sólo root puede añadir un usuario o un grupo al sistema.
alumno29@alumno29-VirtualBox:~$ sudo addgroup samba
Añadiendo el grupo 'samba' (GID 1012) ...
Hecho.
alumno29@alumno29-VirtualBox:~$ sudo adduser juan samba
Añadiendo al usuario 'juan' al grupo 'samba' ...
Añadiendo al usuario juan al grupo samba
Hecho.
alumno29@alumno29-VirtualBox:~$
```

Lo siguiente será crear la carpeta **samba** y las subcarpetas de **lectura** y **escritura**. Cambiaremos la propiedad de la carpeta al grupo **samba** y sus permisos, de forma que puedan guardar los cambios a todos los usuarios del grupo.



```
alumno29@alumno29-VirtualBox: ~  
alumno29@alumno29-VirtualBox:~$ sudo mkdir /samba  
alumno29@alumno29-VirtualBox:~$ sudo mkdir /samba/lectura  
alumno29@alumno29-VirtualBox:~$ sudo mkdir /samba/escritura  
alumno29@alumno29-VirtualBox:~$ sudo chgrp samba -R /samba  
alumno29@alumno29-VirtualBox:~$ sudo chmod 770 -R /samba  
alumno29@alumno29-VirtualBox:~$
```

Por último, se añadirán los usuarios **juan** y **juana** como usuarios del servicio **samba** y se reiniciarán los **smbd** y **nmbd** para que los cambios en el archivo de configuración surtan efecto.

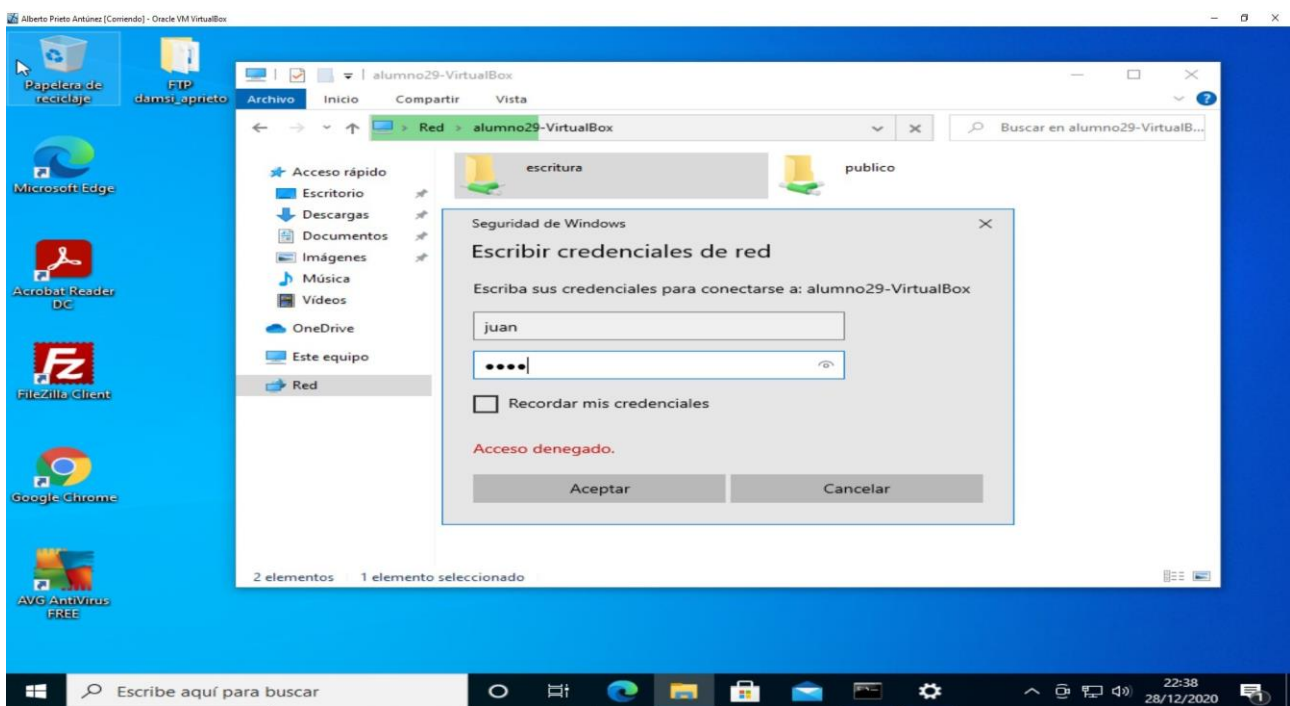
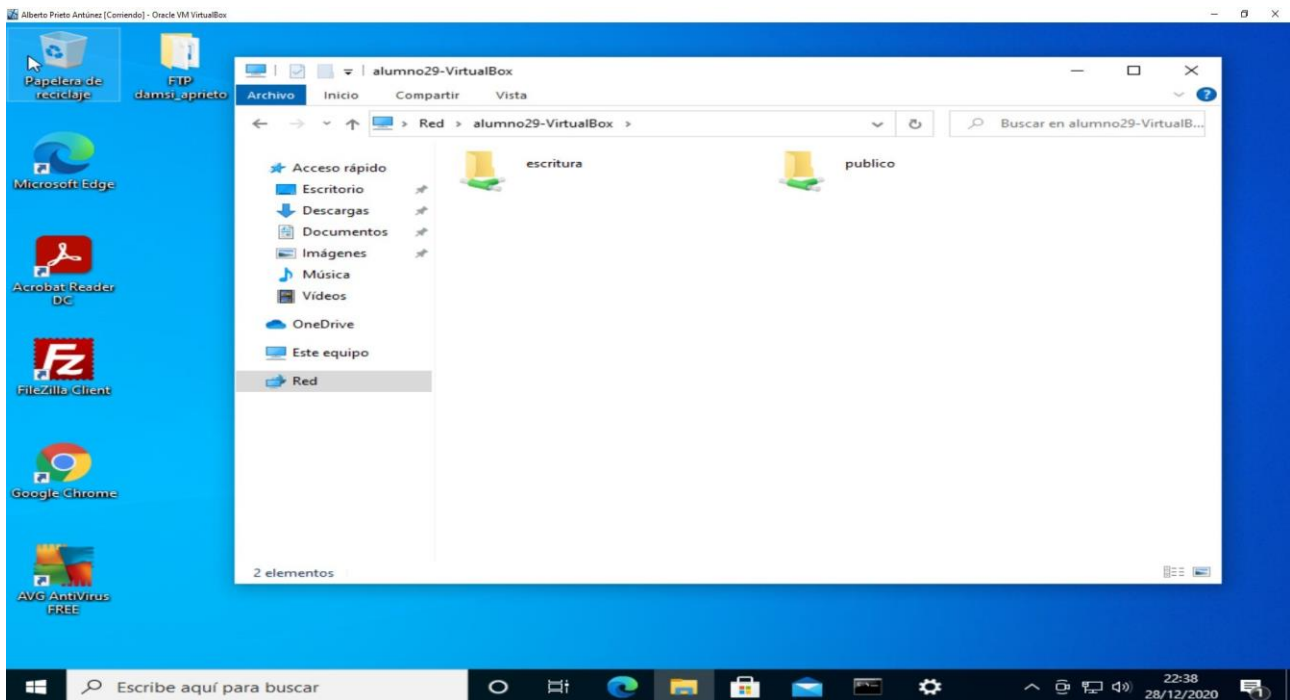


```
alumno29@alumno29-VirtualBox: ~  
alumno29@alumno29-VirtualBox:~$ sudo mkdir /samba  
alumno29@alumno29-VirtualBox:~$ sudo mkdir /samba/lectura  
alumno29@alumno29-VirtualBox:~$ sudo mkdir /samba/escritura  
alumno29@alumno29-VirtualBox:~$ sudo chgrp samba -R /samba  
alumno29@alumno29-VirtualBox:~$ sudo chmod 770 -R /samba  
alumno29@alumno29-VirtualBox:~$ sudo smbpasswd -a juan  
New SMB password:  
Retype new SMB password:  
Added user juan.  
alumno29@alumno29-VirtualBox:~$ sudo smbpasswd -a juana  
New SMB password:  
Retype new SMB password:  
Added user juana.  
alumno29@alumno29-VirtualBox:~$ sudo systemctl restart smbd  
alumno29@alumno29-VirtualBox:~$ sudo systemctl restart nmbd  
alumno29@alumno29-VirtualBox:~$
```

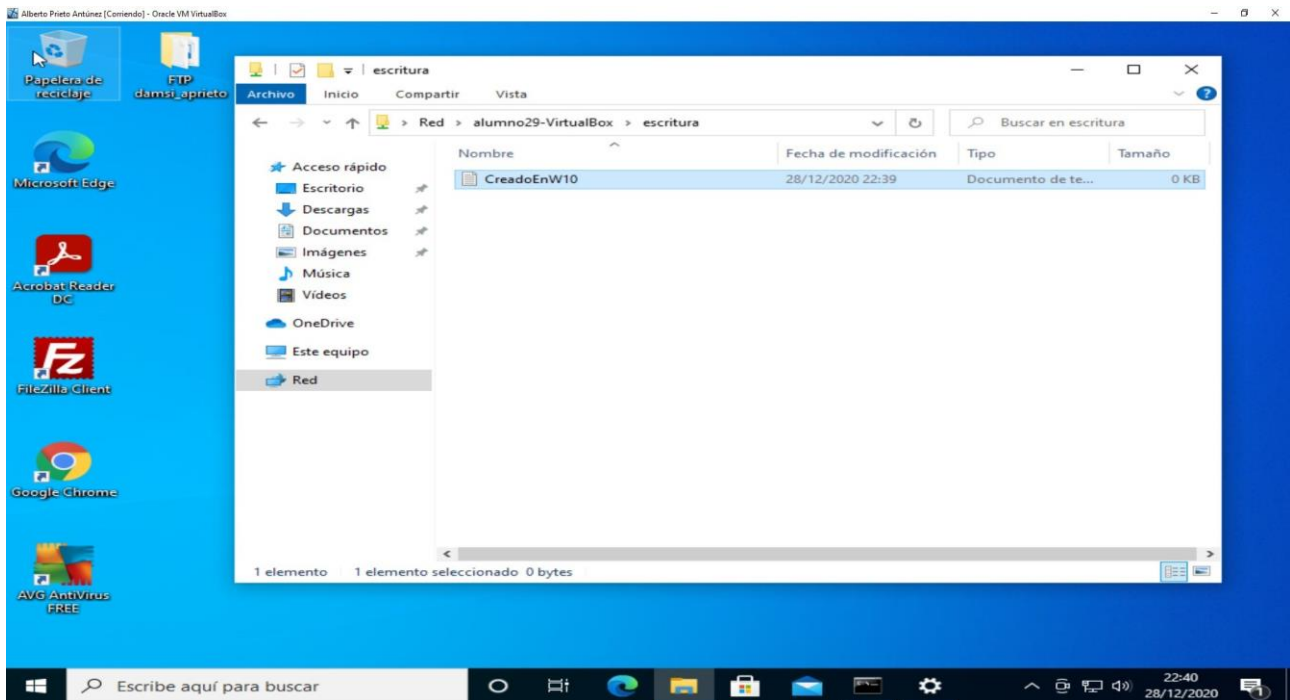
Lo siguiente que haremos es establecer conexión con las carpetas compartidas desde el sistema operativo Windows 10. De forma previa, hemos añadido un adaptador de red interna adicional y establecido una ip fija **192.168.100.102** para nuestro sistema Windows.

Una vez hemos hecho esto, hemos hecho *ping 192.168.100.102* desde la máquina servidor Linux e igualmente *ping 192.168.100.104* desde la máquina cliente Windows y ambas tienen comunicación. Para ver las carpetas compartidas, hemos tenido que solucionar un problema con el puerto 445 de la máquina Linux servidor, para ello, hemos tenido que abrirlo utilizando el comando *sudo ufw allow*

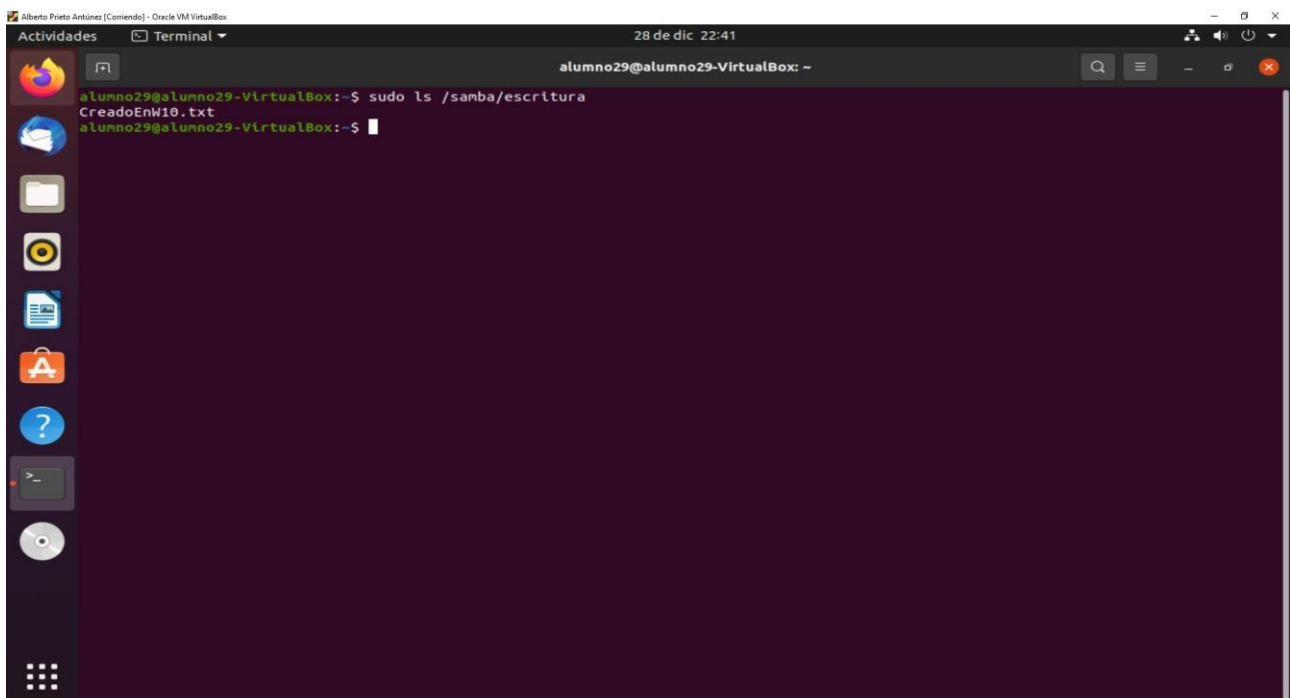
445/tcp y el problema se ha solucionado. Por último, en Windows 10, accedemos a Red y escribimos el nombre de la máquina servidor **\\alumno29-VirtualBox**.



A continuación, accedemos a la carpeta **escritura** con las credenciales del usuario **juan** creados anteriormente. Creamos un fichero de texto que se llame **CreadoEnW10**.



Por lo tanto, se puede ver que si accedemos desde la máquina servidora Linux a dicha carpeta con `sudo ls /samba/escritura` la carpeta que hemos creado desde el sistema Windows aparece correctamente.



Ejercicio 3: Seguir los pasos del Punto 4 de los contenidos de la unidad, para realizar las siguientes acciones:

1. Instalar servicio ssh

Para instalar el servicio SSH, ejecutamos el comando `sudo apt install ssh`

```
Alberto Prieto Antón [Comando] - Oracle VM VirtualBox
Actividades Terminal 28 de dic 22:54
alumno29@alumno29-VirtualBox: ~

alumno29@alumno29-VirtualBox:~$ sudo apt install ssh
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
El paquete indicado a continuación se instaló de forma automática y ya no es necesario.
  libfprint-2-tod1
Utilice «sudo apt autoremove» para eliminarlo.
Se instalarán los siguientes paquetes adicionales:
  ncurses-term openssh-server openssh-sftp-server ssh-import-id
Paquetes sugeridos:
  molly-guard monkeysphere ssh-askpass
Se instalarán los siguientes paquetes NUEVOS:
  ncurses-term openssh-server openssh-sftp-server ssh ssh-import-id
0 actualizados, 5 nuevos se instalarán, 0 para eliminar y 21 no actualizados.
Se necesita descargar 693 kB de archivos.
Se utilizarán 6.130 kB de espacio de disco después de esta operación.
¿Desea continuar? [S/n] S
Des:1 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 openssh-sftp-server amd64 1:8.2p1-4ubuntu0.1 [51,5 kB]
Des:2 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 openssh-server amd64 1:8.2p1-4ubuntu0.1 [377 kB]
Des:3 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 ssh all 1:8.2p1-4ubuntu0.1 [5.080 B]
Des:4 http://es.archive.ubuntu.com/ubuntu focal/main amd64 ncurses-term all 6.2-0ubuntu2 [249 kB]
Des:5 http://es.archive.ubuntu.com/ubuntu focal/main amd64 ssh-import-id all 5.10-0ubuntu1 [10,0 kB]
Descargados 693 kB en 1s (608 kB/s)
Preconfigurando paquetes ...
Seleccionando el paquete openssh-sftp-server previamente no seleccionado.
(Leyendo la base de datos ... 245341 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar .../openssh-sftp-server_1%3a8.2p1-4ubuntu0.1_amd64.deb ...
Desempaquetando openssh-sftp-server (1:8.2p1-4ubuntu0.1) ...
Seleccionando el paquete openssh-server previamente no seleccionado.
Preparando para desempaquetar .../openssh-server_1%3a8.2p1-4ubuntu0.1_amd64.deb ...
Desempaquetando openssh-server (1:8.2p1-4ubuntu0.1) ...
Seleccionando el paquete ssh previamente no seleccionado.
Preparando para desempaquetar .../ssh_1%3a8.2p1-4ubuntu0.1_all.deb ...
Desempaquetando ssh (1:8.2p1-4ubuntu0.1) ...
Seleccionando el paquete ncurses-term previamente no seleccionado.
Preparando para desempaquetar .../ncurses-term_6.2-0ubuntu2_all.deb ...
Desempaquetando ncurses-term (6.2-0ubuntu2) ...
Seleccionando el paquete ssh-import-id previamente no seleccionado.
Preparando para desempaquetar .../ssh-import-id_5.10-0ubuntu1_all.deb ...
Desempaquetando ssh-import-id (5.10-0ubuntu1) ...
```

Con el comando `sudo systemctl status ssh` se comprueba si está activado.

```
Alberto Prieto Antón [Comando] - Oracle VM VirtualBox
Actividades Terminal 28 de dic 22:55
alumno29@alumno29-VirtualBox: ~

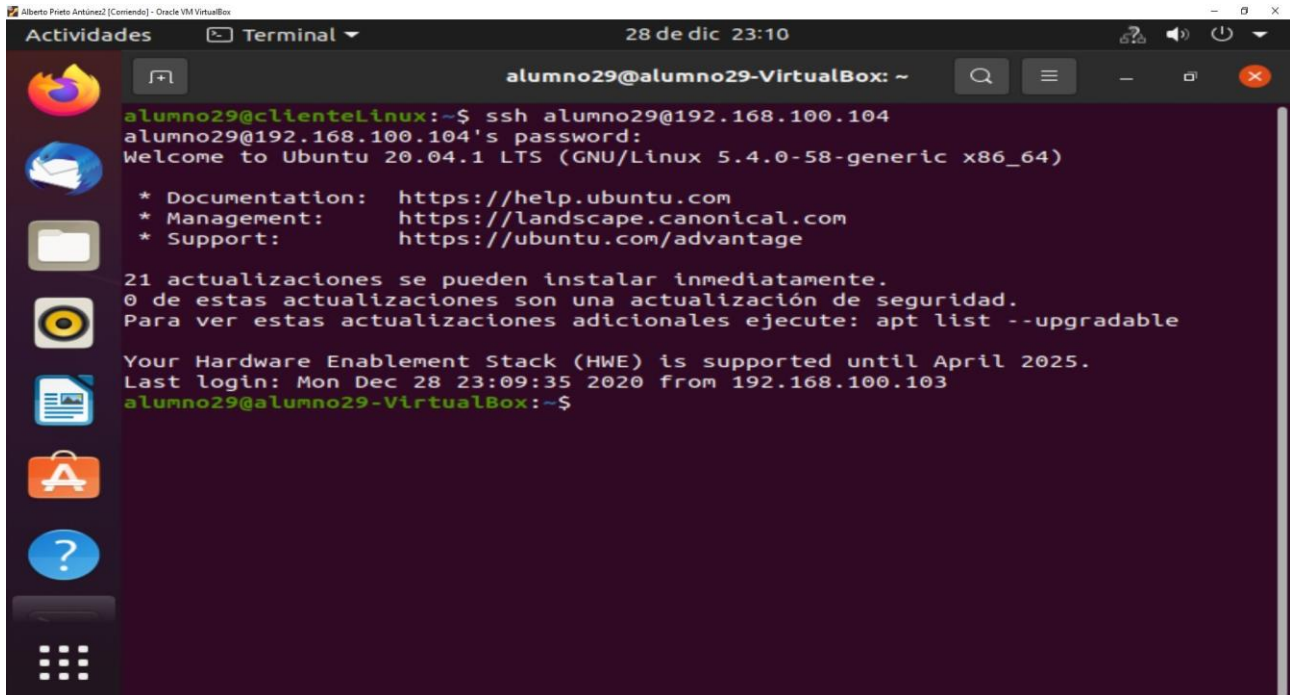
alumno29@alumno29-VirtualBox:~$ sudo systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enabled)
   Active: active (running) since Mon 2020-12-28 22:53:57 WET; 1min 57s ago
     Docs: man:sshd(8)
           man:sshd_config(5)
   Main PID: 7110 (sshd)
      Tasks: 1 (limit: 1110)
     Memory: 1.6M
    CGroup: /system.slice/ssh.service
            └─7110 sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups

dic 28 22:53:56 alumno29-VirtualBox systemd[1]: Starting OpenBSD Secure Shell server...
dic 28 22:53:57 alumno29-VirtualBox sshd[7110]: Server listening on 0.0.0.0 port 22.
dic 28 22:53:57 alumno29-VirtualBox sshd[7110]: Server listening on :: port 22.
dic 28 22:53:57 alumno29-VirtualBox systemd[1]: Started OpenBSD Secure Shell server.
alumno29@alumno29-VirtualBox:~$
```

2.

Conectar desde clienteLinux y ejecutar algunos comandos

En la máquina cliente no es necesario instalar nada, simplemente hay que conectarse utilizando `ssh user@IP` o `user@Nombre_equipo`. Donde User es un usuario de la máquina servidor, en nuestro caso: `alumno29 @192.168.100.104`



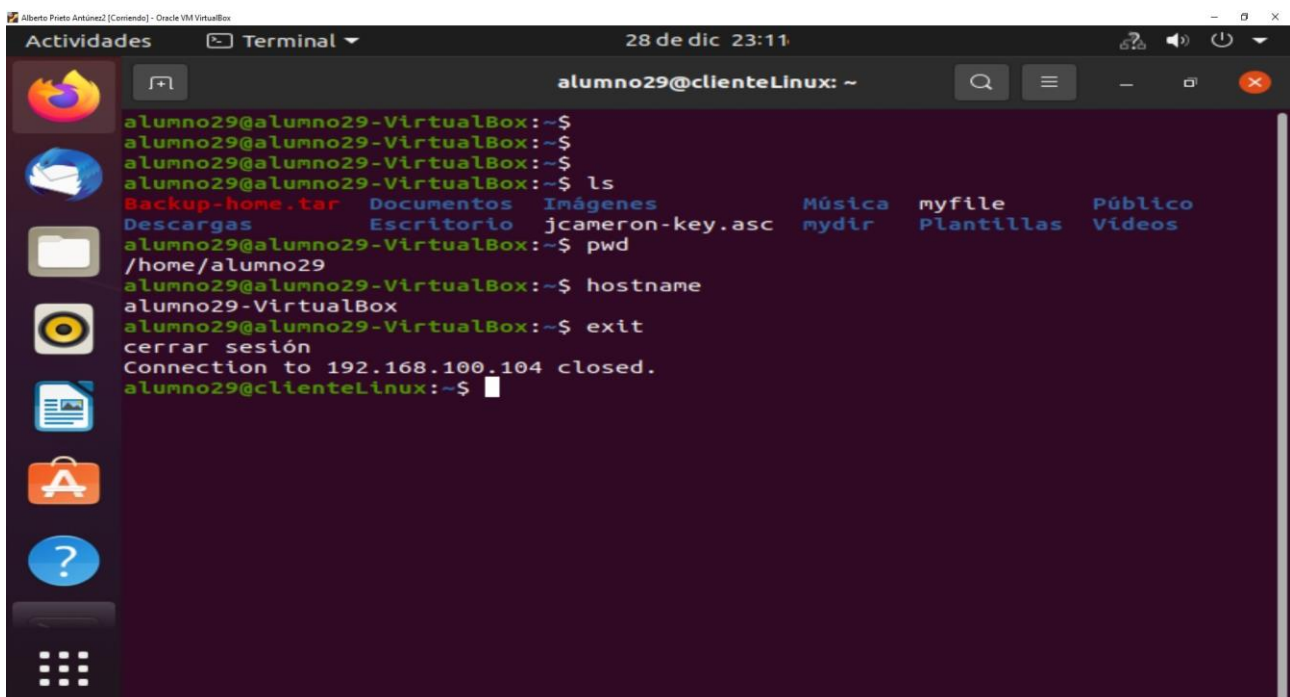
The screenshot shows a terminal window titled 'Actividades' with a 'Terminal' tab. The prompt is 'alumno29@alumno29-VirtualBox: ~'. The user enters the command 'ssh alumno29@192.168.100.104'. The terminal shows the password prompt, the login message 'Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-58-generic x86_64)', and system updates. The user then enters 'ls' and 'pwd' commands. The terminal output shows the directory listing and the current path '/home/alumno29'. The user then enters 'hostname' and 'exit' commands. The terminal output shows the hostname 'alumno29-VirtualBox' and the message 'cerrar sesión'.

```
alumno29@clienteLinux:~$ ssh alumno29@192.168.100.104
alumno29@192.168.100.104's password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-58-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

21 actualizaciones se pueden instalar inmediatamente.
0 de estas actualizaciones son una actualización de seguridad.
Para ver estas actualizaciones adicionales ejecute: apt list --upgradable

Your Hardware Enablement Stack (HWE) is supported until April 2025.
Last login: Mon Dec 28 23:09:35 2020 from 192.168.100.103
alumno29@alumno29-VirtualBox:~$
```



The screenshot shows the same terminal window as the previous one. The user enters 'ls' and 'pwd' commands. The terminal output shows the directory listing and the current path '/home/alumno29'. The user then enters 'hostname' and 'exit' commands. The terminal output shows the hostname 'alumno29-VirtualBox' and the message 'cerrar sesión'. The user then enters 'exit' and the terminal shows 'Connection to 192.168.100.104 closed.' and the prompt 'alumno29@clienteLinux:~\$'.

```
alumno29@alumno29-VirtualBox:~$
alumno29@alumno29-VirtualBox:~$
alumno29@alumno29-VirtualBox:~$ ls
Backup-home.tar  Documentos  Imágenes    Música      myfile      Público
Descargas       Escritorio  jcameron-key.asc  mydir      Plantillas  Videos
alumno29@alumno29-VirtualBox:~$ pwd
/home/alumno29
alumno29@alumno29-VirtualBox:~$ hostname
alumno29-VirtualBox
alumno29@alumno29-VirtualBox:~$ exit
cerrar sesión
Connection to 192.168.100.104 closed.
alumno29@clienteLinux:~$
```

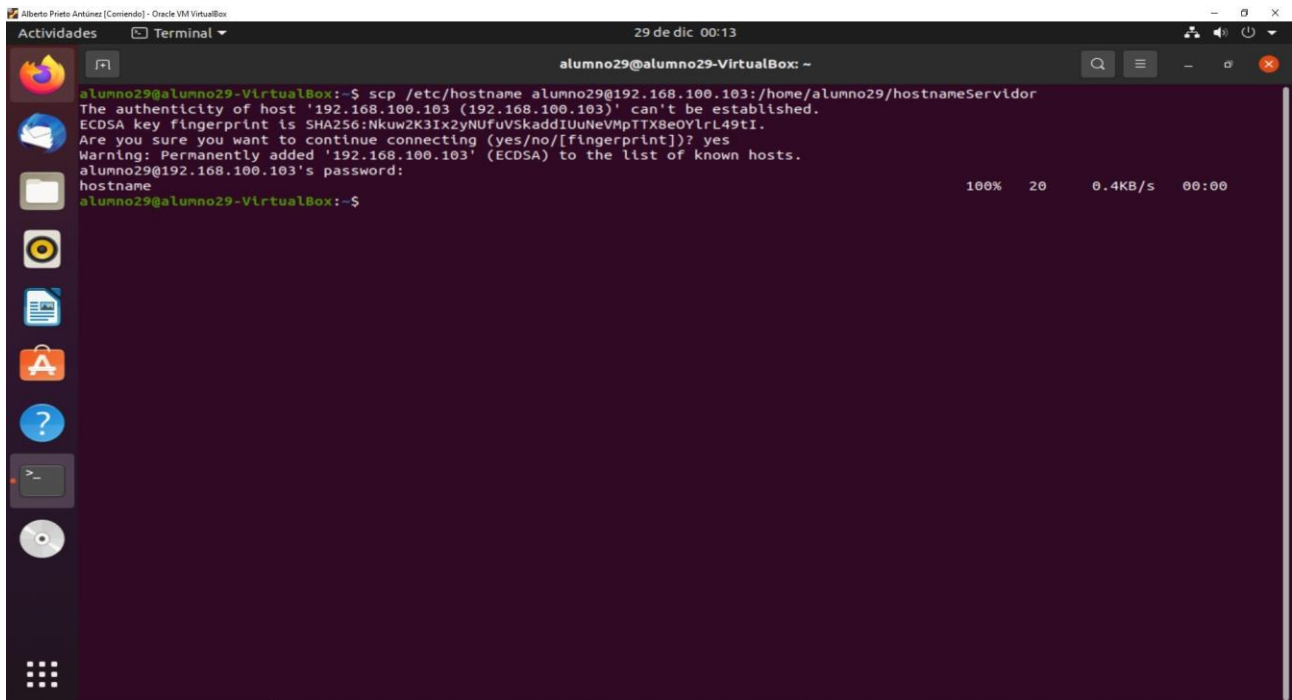
3.

Dentro de la conexión al servidor por SSH, hemos ejecutado algunos comandos como *ls*, *pwd*, *hostname* y *exit*. Este último para cerrar la conexión ssh.

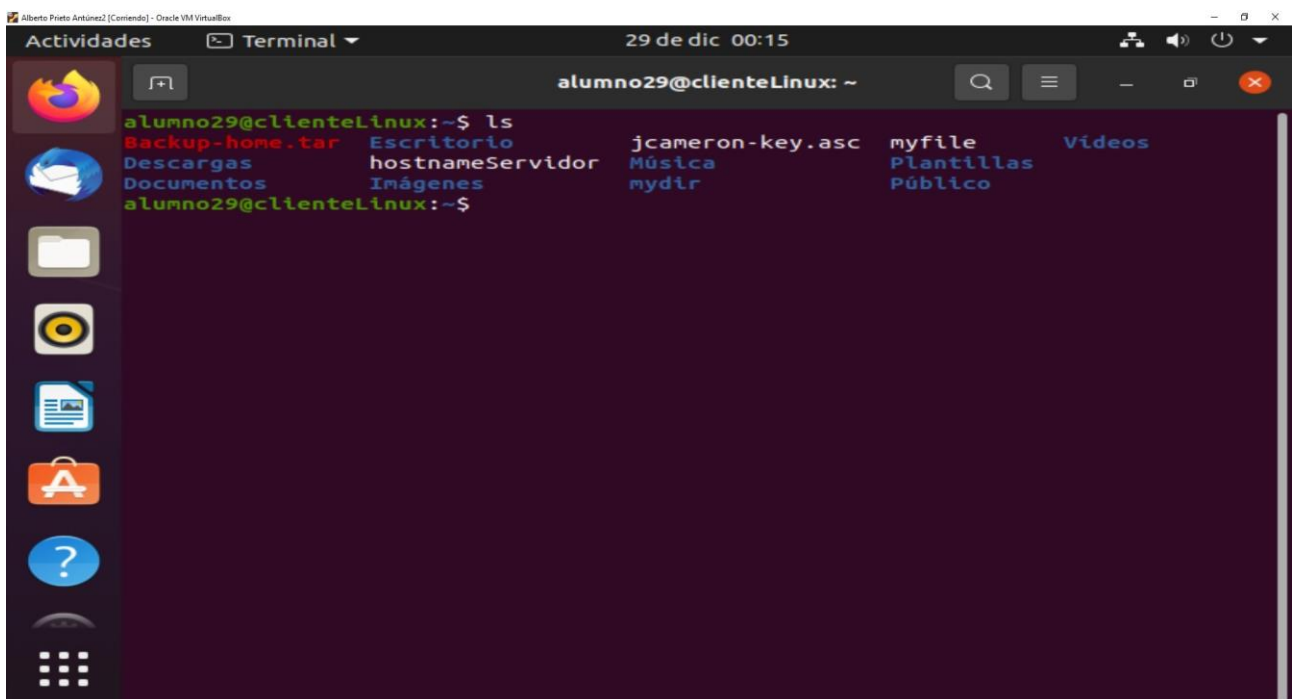
Copiar el archivo `/etc/hostname` del servidor a la máquina cliente con el nombre `hostnameServidor`

Para ello, desde la máquina servidor, ejecutaremos : *scp /etc/hostname*

alumno29@192.168.100.103:/home/alumno29/hostnameServidor



```
alumno29@alumno29-VirtualBox: ~  
alumno29@alumno29-VirtualBox:~$ scp /etc/hostname alumno29@192.168.100.103:/home/alumno29/hostnameServidor  
The authenticity of host '192.168.100.103 (192.168.100.103)' can't be established.  
ECDSA key fingerprint is SHA256:Nkuw2K3Ix2yNUFvVSkaddIUuNeVmPTTX8eOYrL49tI.  
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes  
Warning: Permanently added '192.168.100.103' (ECDSA) to the list of known hosts.  
alumno29@192.168.100.103's password:  
hostname                                                                    100% 20    0.4KB/s   00:00  
alumno29@alumno29-VirtualBox:~$
```



```
alumno29@clienteLinux: ~  
alumno29@clienteLinux:~$ ls  
Backup-home.tar  Escritorio  jcameron-key.asc  myfile  Videos  
Descargas        hostnameServidor  Música           Plantillas  
Documentos       Imágenes       mydir            Público
```

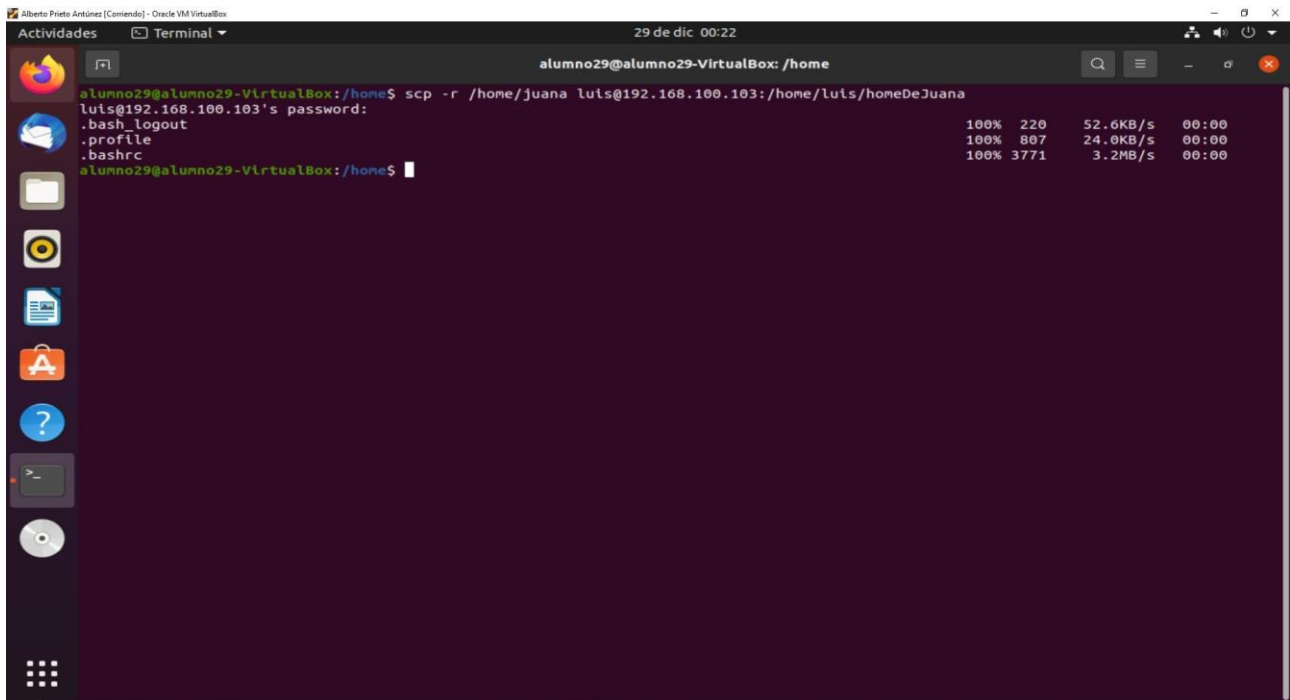

4.

Si desde la máquina cliente hacemos `ls` podemos comprobar que el archivo **hostnameServidor** se ha transferido correctamente desde la máquina servidor a la máquina cliente.

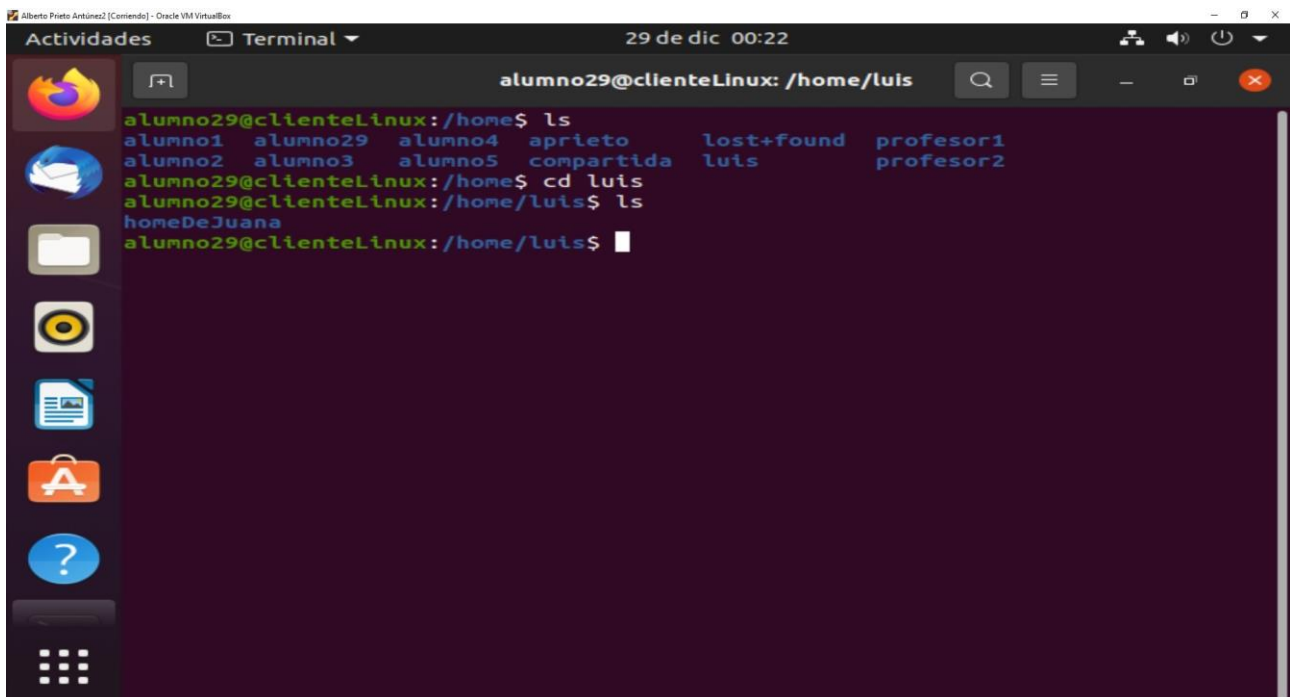
Copiar el directorio `/home/juana` del cliente Linux al servidor como el usuario

luis Para ello desde la máquina donde se encuentra el `/home` de **juana** ejecutamos: `scp`

`-r /home/juana luis@192.168.100.103:/home/luis/homeDeJuana`



```
alumno29@alumno29-VirtualBox: /home
alumno29@alumno29-VirtualBox: /home$ scp -r /home/juana luis@192.168.100.103:/home/luis/homeDeJuana
luis@192.168.100.103's password:
.bash_logout 100% 220 52.6KB/s 00:00
.profile 100% 807 24.0KB/s 00:00
.bashrc 100% 3771 3.2MB/s 00:00
alumno29@alumno29-VirtualBox: /home$
```



```
alumno29@clienteLinux: /home/luis
alumno29@clienteLinux: /home$ ls
alumno1  alumno29  alumno4  aprieto  lost+found  profesor1
alumno2  alumno3  alumno5  compartida  luis  profesor2
alumno29@clienteLinux: /home$ cd luis
alumno29@clienteLinux: /home/luis$ ls
homeDeJuana
alumno29@clienteLinux: /home/luis$
```

5.

Una vez transferido correctamente todo el contenido, podemos comprobar desde la máquina cliente, al cual le hemos creado el usuario **luis**, que su carpeta contiene **homeDeJuana**.

Ejercicio 4: Realizar un mapa conceptual en el que aparezcan los comandos vistos y usados en esta actividad, agrupados por sus funcionalidades. Debe aparecer el nombre del comando, indicar para qué sirve y un ejemplo.

Comando	Descripción	Ejemplo
sudo	Permite al usuario ejecutar programas y comandos como usuario "root".	\$sudo apt-get update
sudo -i	Permite entrar en el entorno del terminal en modo "root" de forma permanente.	\$sudo -i
echo	Es un comando de impresión de texto en pantalla o en archivo.	\$echo clienteLinux > /etc/hostname
nano	Sirve para editar archivos de texto.	\$sudo nano /etc/hosts
nano -c	Sirve para editar archivos de texto conociendo la línea y columna en la que se encuentra el cursor.	\$sudo nano -c /etc/netplan/01network-manager-all.yaml
hostname	Sirve para conocer el nombre de la máquina.	\$hostname
hostname [nuevo_nombre]	Sirve para cambiar el nombre de la máquina.	\$sudo hostname clienteLinux
ifconfig	Sirve para conocer o configurar los parámetros de las interfaces de red de la máquina.	\$ifconfig
ls	Permite listar los archivos y directorios de un directorio.	\$ls /sys/class/net
ls -l	Muestra el listado de archivos y directorios en formato largo y mostrando los permisos e información adicional.	\$ls -l /etc/netplan
ping	Envía y recibe paquetes a una dirección ip dada.	\$ping 8.8.8.8
apt install	Sirve para instalar un determinado programa con apt (Advanced Package Tool).	\$sudo apt install samba sambacommon-bin
systemctl	Sirve para controlar el sistema y sus servicios	\$systemctl status smbd
adduser	Sirve para crear un usuario nuevo.	\$sudo adduser juan
adduser [usuario] [grupo]	Permite incorporar un usuario a un grupo.	\$sudo adduser juan samba
addgroup	Sirve para crear un grupo nuevo.	\$sudo addgroup samba
mkdir	Permite crear un subdirectorio o carpeta del sistema de archivos.	\$sudo mkdir /samba/lectura

chgrp	Permite cambiar el grupo de usuarios propietario de un archivo o directorio. Se le añade -R para que se aplique en el directorio, subdirectorio y archivos contenidos.	\$sudo chgrp samba -R /samba
chmod	Permite cambiar los permisos de un fichero o directorio. Se añade -R para que se aplique también al contenido de dicho directorio.	\$sudo chmod 770 -R /samba

smbpasswd	Se utiliza para crear un nuevo usuario en un controlador de dominio en samba. Se usa -a para agregar un usuario existente.	\$sudo smbpasswd -a juan
systemctl restart	Reinicia un servicio para aplicar cambios.	\$systemctl restart smbd
ufw allow	Permite abrir un puerto determinado.	\$sudo ufw allow 22/tcp
ssh	Permite conectarse de manera remota a través de un terminal con user@ip o user@nombre_máquina.	\$ssh alumno29@192.168.100.104
exit	Permite salir del modo "root" o de la conexión ssh.	\$exit
pwd	Permite conocer el directorio en el que nos encontramos.	\$pwd
scp	Es un servicio de SSH que permite copiar ficheros de una máquina a otra. \$scp [fichero o directorio origen] [dirección ssh user@(ip o máquina)]:[directorio destino]. Se le añade un -r para copiar los subdirectorios o o archivos contenidos.	\$scp /etc/hostname alumno29@192.168.100.103:/home
cd	Sirve para movernos por el árbol de directorios.	\$cd .. \$cd /home/alumno29