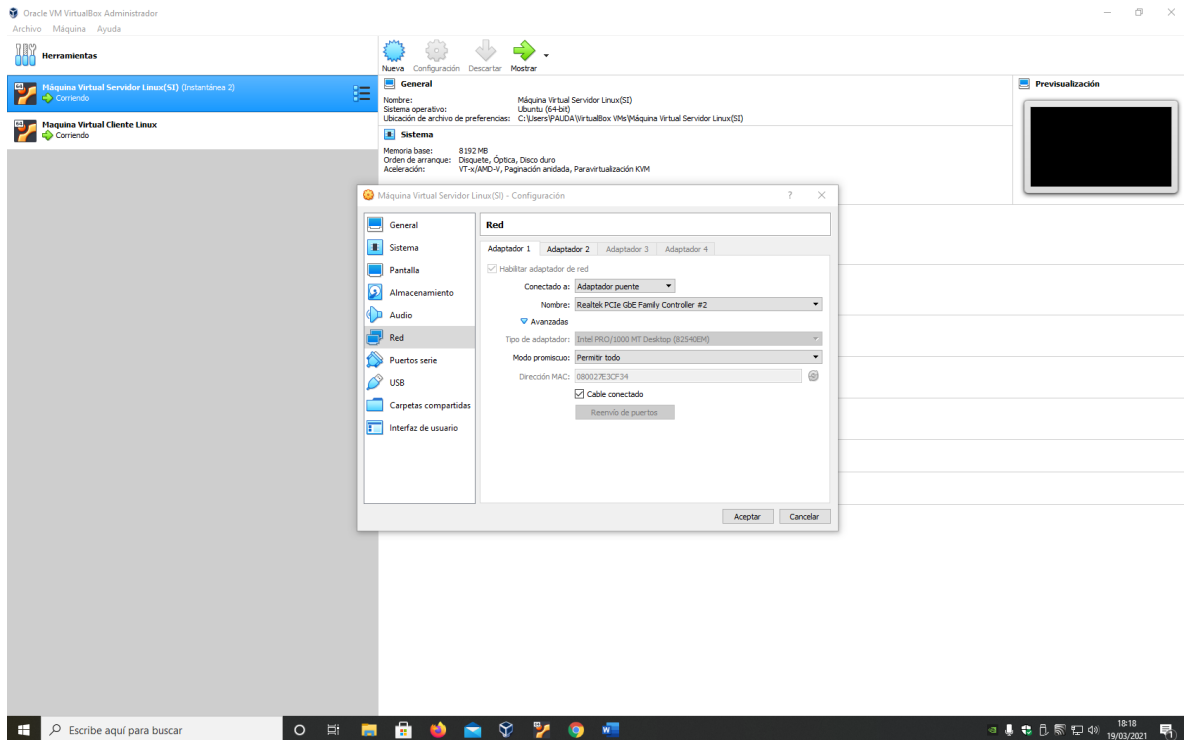
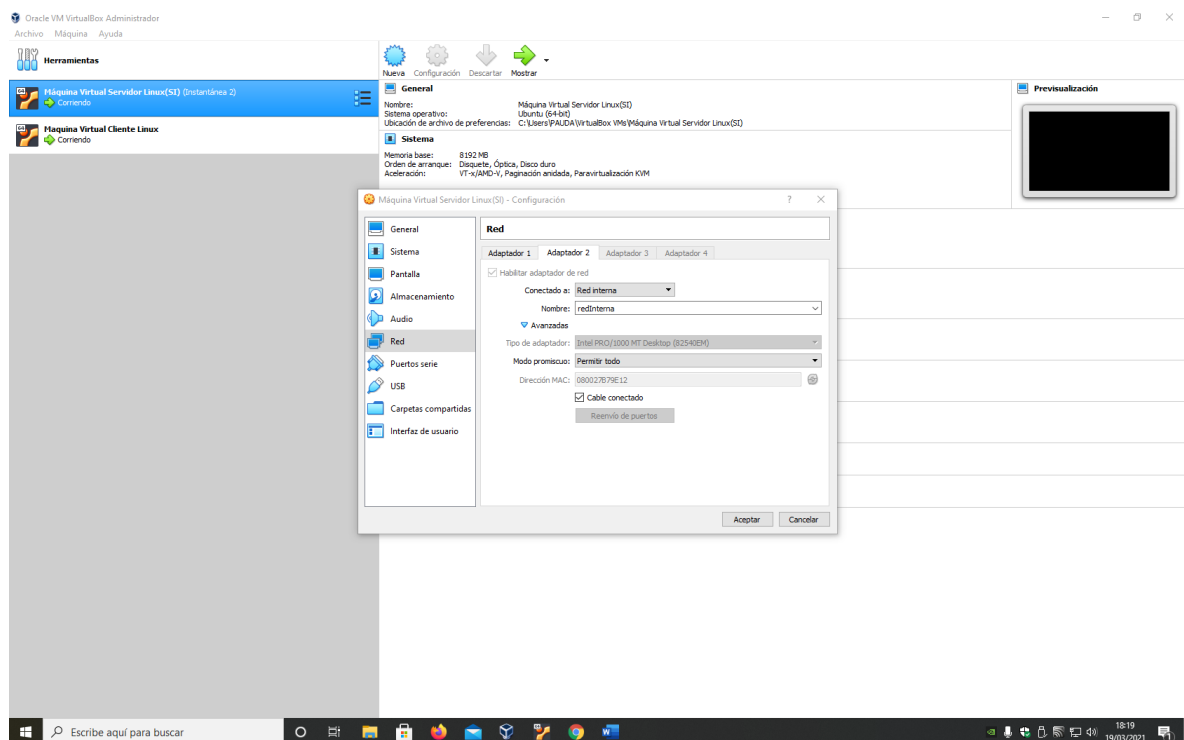


ACTIVIDAD 10.1

Para realizar esta tarea nos vamos al apartado de configuración de la máquina virtual instalada en nuestro equipo anfitrión y en el apartado red configuramos los 2 adaptadores para las 2 interfaces de nuestra red:

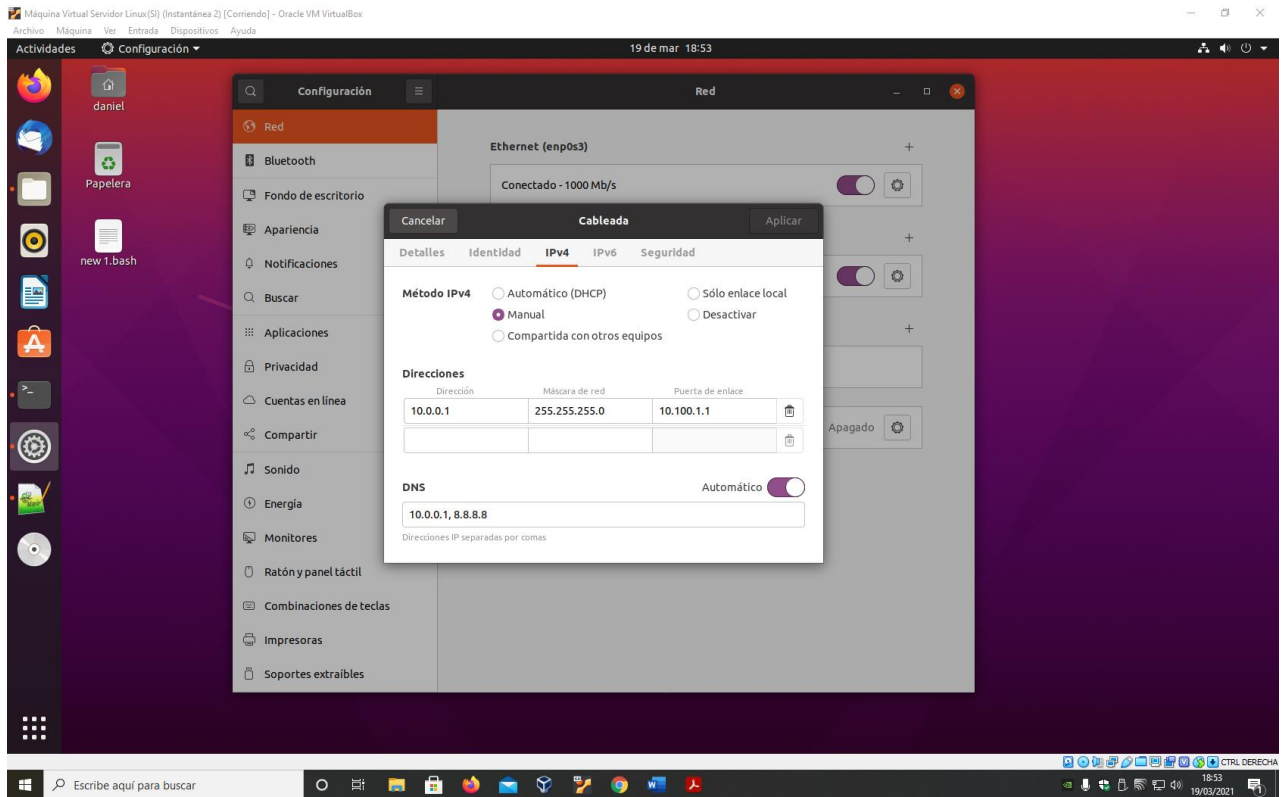


El adaptador 1 le decimos que es un adaptador puente entre nuestra red física de casa y la red virtual del s.o.. Le asignamos a una tarjeta de red ethernet en este caso la del ordenador anfitrión y le decimos que por defecto permita todas las conexiones.

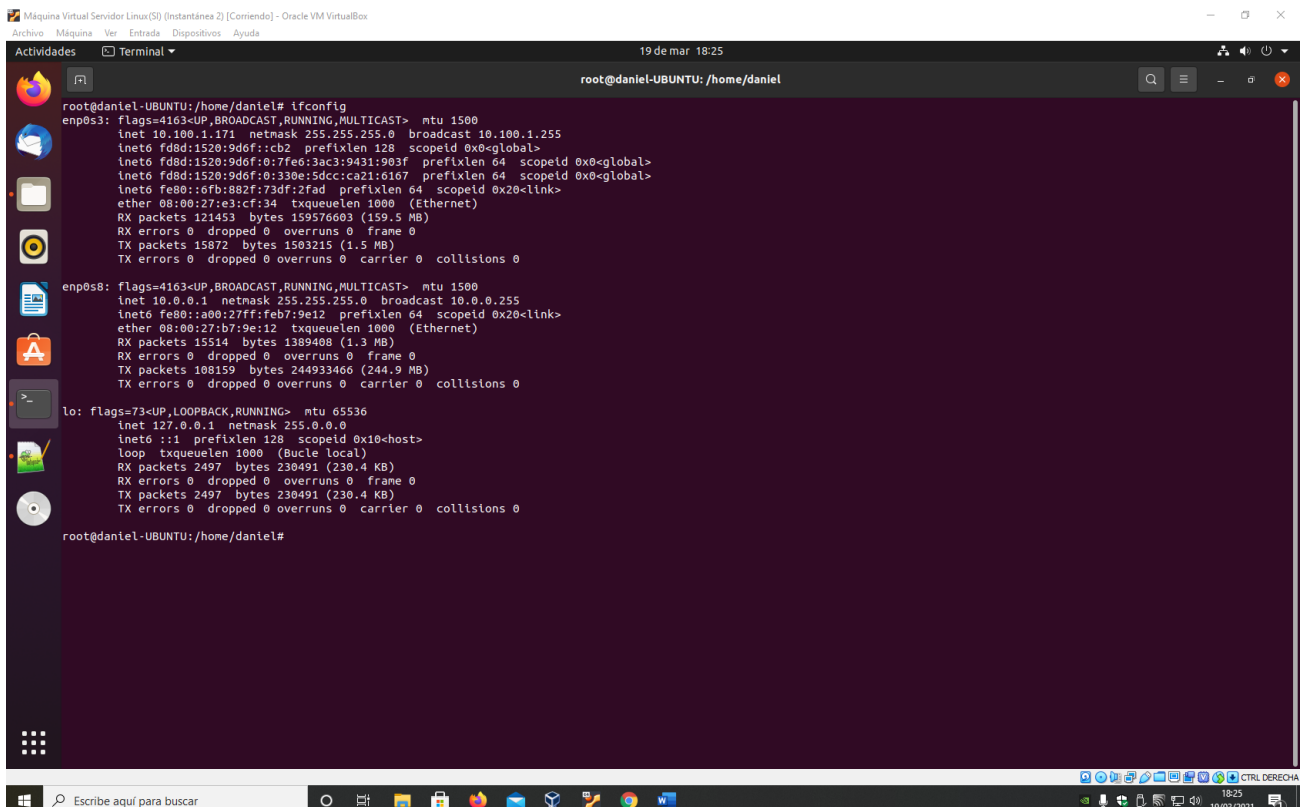


El adaptador 2 vamos a configurarlo como red interna y le vamos a dar el nombre de "redInterna", también permitimos todas las conexiones

Una vez configurados los adaptadores arrancaremos nuestro s.o. servidor en la máquina virtual y vamos a configurar las 2 interfaces de red que hemos creado, una sería la “enp0s3” configurada por dhcp para que el router le asigne direcciones de manera automática que permitirá la conexión a internet y la otra interfaz “enp0s8” será donde vamos a crear nuestra red interna, asignándole las siguientes direcciones a través de la interfaz gráfica de Ubuntu:

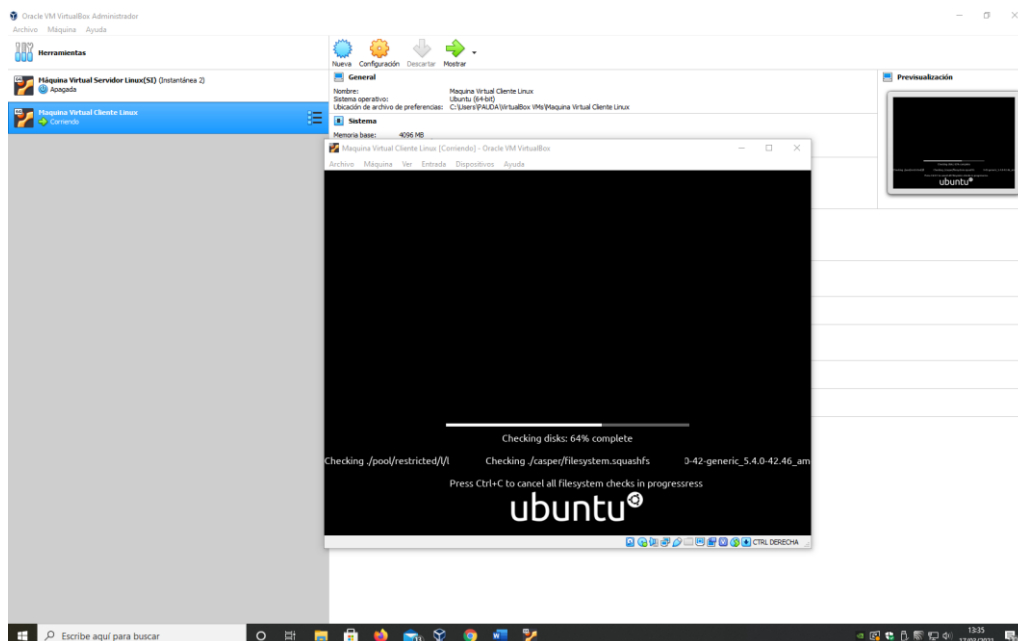


Una vez tenemos todo configurado nos vamos al terminal y comprobamos utilizando el comando “ifconfig” que la red ha quedado bien configurada.

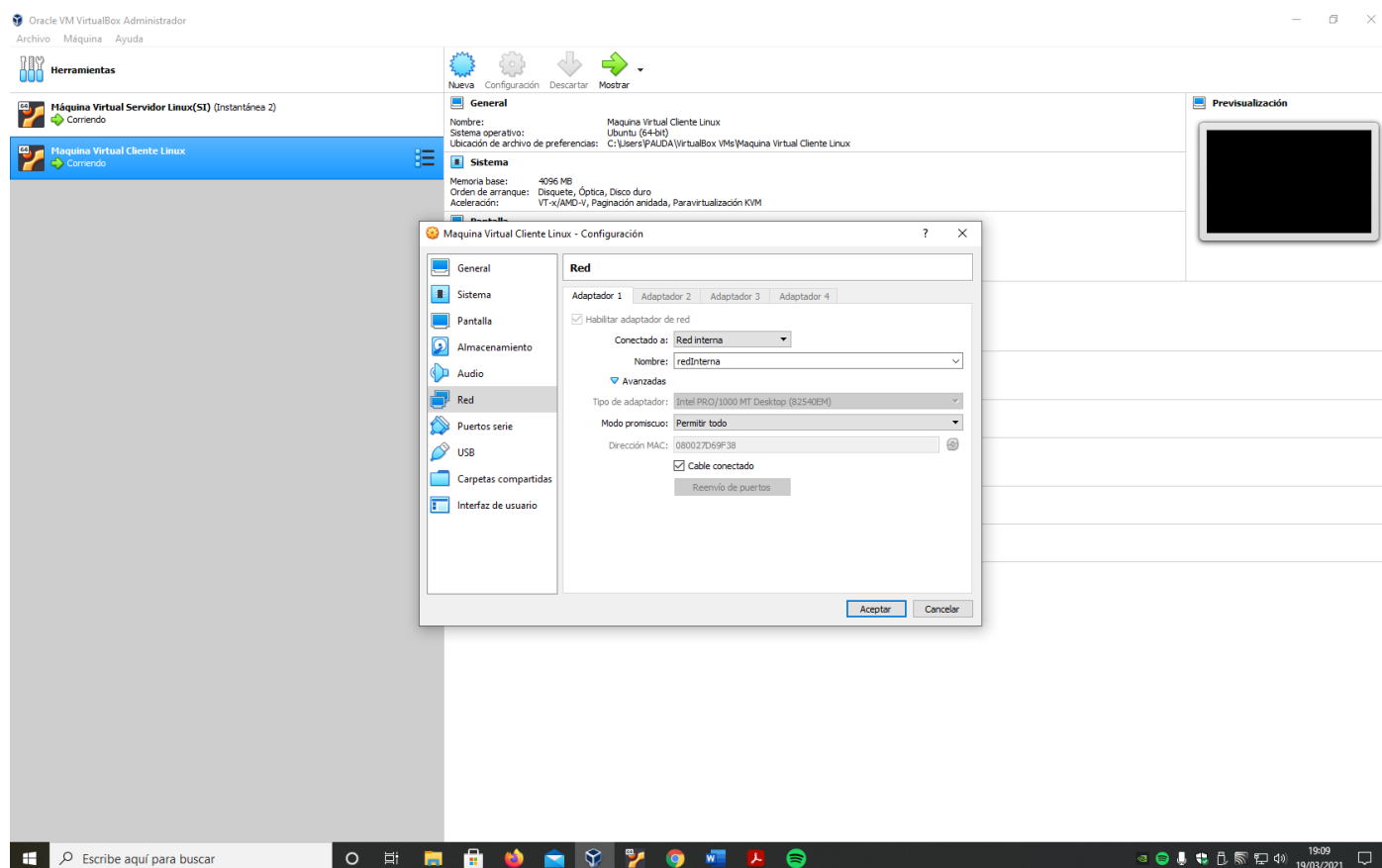


ACTIVIDAD 10.2

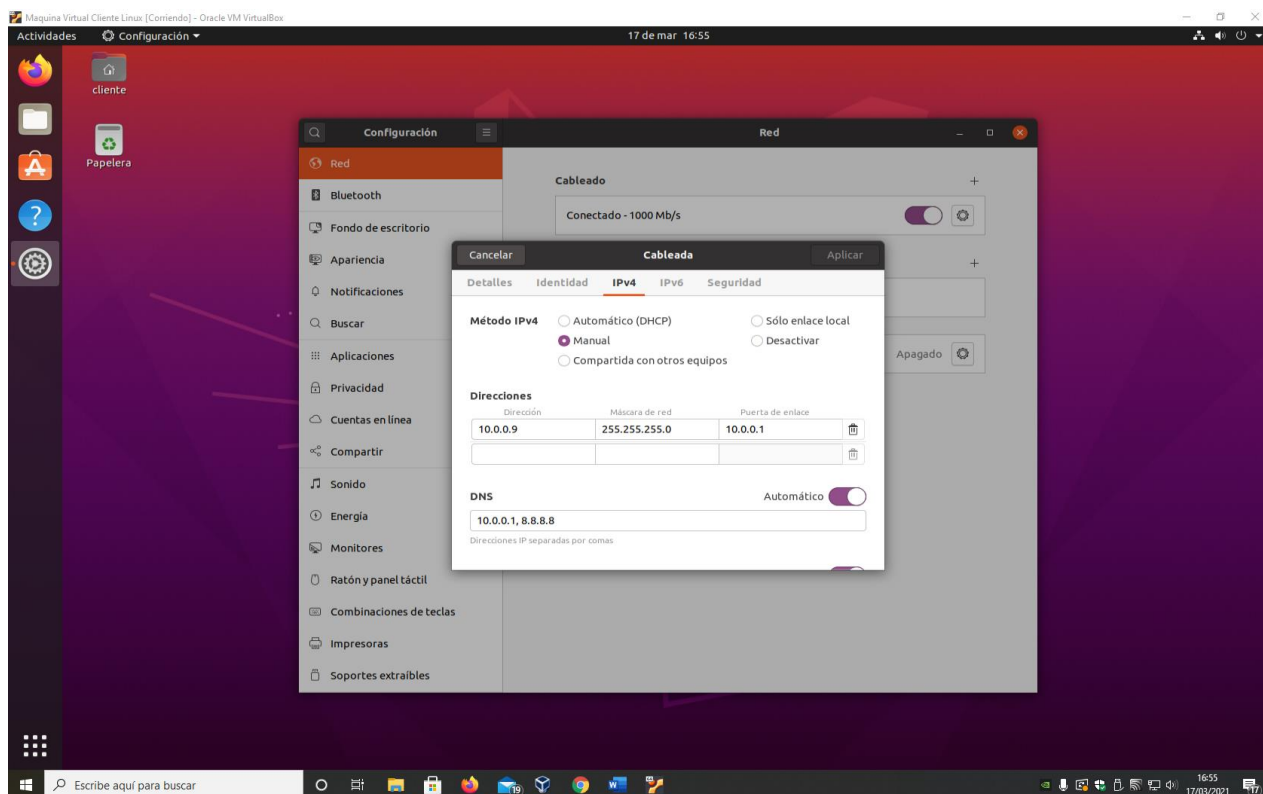
Creamos una segunda máquina virtual con el nombre “cliente” y en el que instalamos el s.o. Ubuntu 20.04:



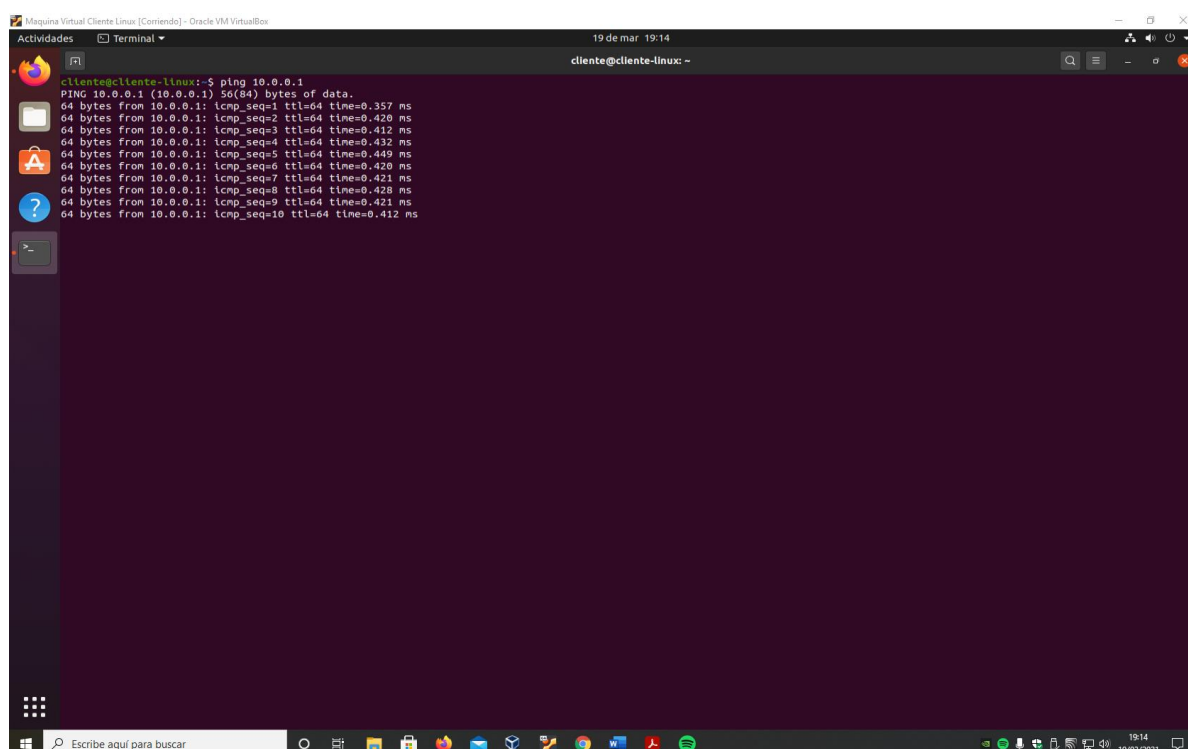
Una vez hemos instalado el s.o. de Ubuntu en la máquina cliente nos disponemos a configurar el adaptador de red como hicimos antes, nos vamos a configuración apartado de red y en opciones de adaptador 1 le decimos que estará conectado a una red interna con el mismo nombre con el que hemos creado la red interna del servidor, es decir “redInterna” permitiendo conexión a todos:



Arrancamos nuestra máquina virtual con el sistema cliente y teniendo ya arrancado el sistema servidor para poder comprobar que nuestra red interna se ha configurado de manera correcta. A través del menú de configuración del sistema vamos a configurar la interfaz de red del adaptador de manera manual con los datos que nos pide el ejercicio:



Una vez configurada la interfaz vamos a comprobar que estamos conectados a la red interna haciendo *ping* a la dirección del servidor 10.0.0.1 nos saldrá lo siguiente, como obtenemos respuesta del mismo quiere decir que ya estamos conectados



ACTIVIDAD 10.3

Vamos a configurar el iptables en sistema operativo de nuestro “servidor” para que la red interna tenga conexión a internet:

1. Establecemos que el sistema actúe de router:

```
root@daniel-UBUNTU:/home/daniel# echo "1" >/proc/sys/net/ipv4/ip_forward
```

2. Limpiamos la configuración del cortafuegos:

```
root@daniel-UBUNTU:/home/daniel# iptables -t nat -F
```

3. Indicamos que la “redInterna” tiene salida al exterior por la nat:

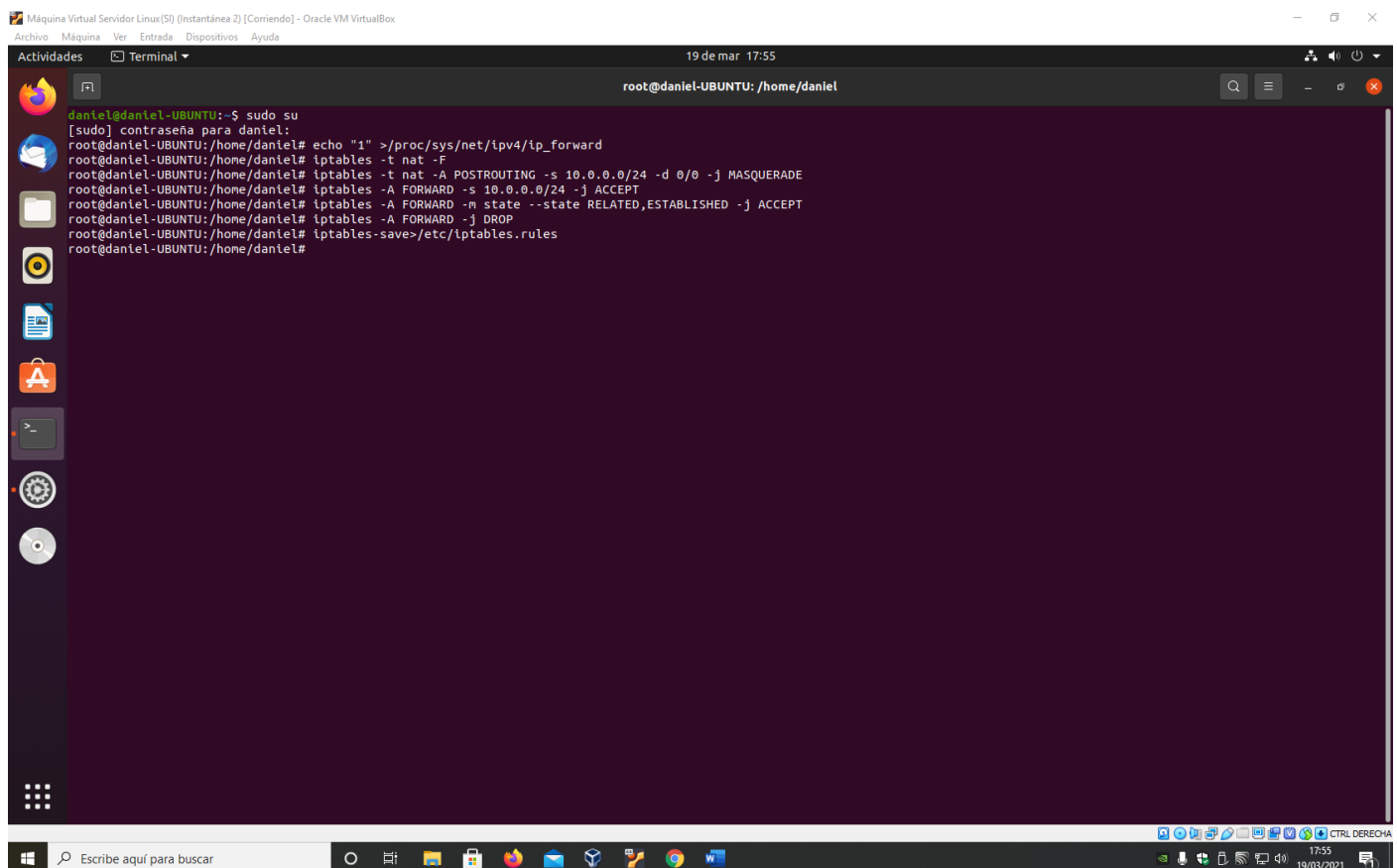
```
root@daniel-UBUNTU:/home/daniel# iptables -t nat -A POSTROUTING -s 10.0.0.0/24 -d 0/0 -j MASQUERADE
```

4. Permitimos todo el tráfico de “redInterna” y el resto se deniega:

```
root@daniel-UBUNTU:/home/daniel# iptables -A FORWARD -s 10.0.0.0/24 -j ACCEPT
root@daniel-UBUNTU:/home/daniel# iptables -A FORWARD -m state --state
RELATED,ESTABLISHED -j ACCEPT
root@daniel-UBUNTU:/home/daniel# iptables -A FORWARD -j DROP
```

5. Guardamos la configuración del cortafuegos:

```
root@daniel-UBUNTU:/home/daniel# iptables-save>/etc/iptables.rules
```



The screenshot shows a terminal window titled "Máquina Virtual Servidor Linux(S) (Instantánea 2) [Corriendo] - Oracle VM VirtualBox". The terminal output shows the following commands and their results:

```
daniel@daniel-UBUNTU:~$ sudo su
[sudo] contraseña para daniel:
root@daniel-UBUNTU:/home/daniel# echo "1" >/proc/sys/net/ipv4/ip_forward
root@daniel-UBUNTU:/home/daniel# iptables -t nat -F
root@daniel-UBUNTU:/home/daniel# iptables -t nat -A POSTROUTING -s 10.0.0.0/24 -d 0/0 -j MASQUERADE
root@daniel-UBUNTU:/home/daniel# iptables -A FORWARD -s 10.0.0.0/24 -j ACCEPT
root@daniel-UBUNTU:/home/daniel# iptables -A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT
root@daniel-UBUNTU:/home/daniel# iptables -A FORWARD -j DROP
root@daniel-UBUNTU:/home/daniel# iptables-save>/etc/iptables.rules
```

Modificamos el fichero `/etc/sysctl.conf` para activar la variable `net.ipv4.ip_forward=1`:

```

# Uncomment the next line to enable TCP/IP SYN cookies
# See http://lwn.net/Articles/277146/
# Note: This may impact IPv6 TCP sessions too
#net.ipv4.tcp_syncookies=1

# Uncomment the next line to enable packet forwarding for IPv4
net.ipv4.ip_forward=1

# Uncomment the next line to enable packet forwarding for IPv6
# Enabling this option disables Stateless Address Autoconfiguration
# based on Router Advertisements for this host
#net.ipv6.conf.all.forwarding=1

#####
# Additional settings - these settings can improve the network
# security of the host and prevent against some network attacks
# including spoofing attacks and man in the middle attacks through
# redirection. Some network environments, however, require that these
# settings are disabled so review and enable them as needed.
#
# Do not accept ICMP redirects (prevent MITM attacks)
#net.ipv4.conf.all.accept_redirects = 0
#net.ipv6.conf.all.accept_redirects = 0
# or
# Accept ICMP redirects only for gateways listed in our default
# gateway list (enabled by default)
#net.ipv4.conf.all.secure_redirects = 1
#
# Do not send ICMP redirects (we are not a router)
#net.ipv4.conf.all.send_redirects = 0
#
# Do not accept IP source route packets (we are not a router)
#net.ipv4.conf.all.accept_source_route = 0
#net.ipv6.conf.all.accept_source_route = 0
#
# Log Martian Packets
#net.ipv4.conf.all.log_martians = 1
#
#####
# Magic system request Key
# 0=disable, 1=enable all, >1 bitmask of sysrq functions

```

Para comprobar que tenemos conexión a internet hacemos un *ping* a la dirección www.google.es desde el equipo cliente y vemos que tenemos respuesta:

```

cliente@cliente-linux:~$ ping www.google.es
PING www.google.es (142.250.184.3) 56(84) bytes of data:
64 bytes from mad41s10-ln-f3.1e100.net (142.250.184.3): icmp_seq=1 ttl=116 time=6.70 ms
64 bytes from mad41s10-ln-f3.1e100.net (142.250.184.3): icmp_seq=2 ttl=116 time=6.19 ms
64 bytes from mad41s10-ln-f3.1e100.net (142.250.184.3): icmp_seq=3 ttl=116 time=6.24 ms
64 bytes from mad41s10-ln-f3.1e100.net (142.250.184.3): icmp_seq=4 ttl=116 time=6.24 ms
64 bytes from mad41s10-ln-f3.1e100.net (142.250.184.3): icmp_seq=5 ttl=116 time=6.24 ms
64 bytes from mad41s10-ln-f3.1e100.net (142.250.184.3): icmp_seq=6 ttl=116 time=6.24 ms
64 bytes from mad41s10-ln-f3.1e100.net (142.250.184.3): icmp_seq=7 ttl=116 time=6.24 ms
64 bytes from mad41s10-ln-f3.1e100.net (142.250.184.3): icmp_seq=8 ttl=116 time=6.24 ms
64 bytes from mad41s10-ln-f3.1e100.net (142.250.184.3): icmp_seq=9 ttl=116 time=6.24 ms
64 bytes from mad41s10-ln-f3.1e100.net (142.250.184.3): icmp_seq=10 ttl=116 time=6.24 ms
64 bytes from mad41s10-ln-f3.1e100.net (142.250.184.3): icmp_seq=11 ttl=116 time=6.24 ms
64 bytes from mad41s10-ln-f3.1e100.net (142.250.184.3): icmp_seq=12 ttl=116 time=6.24 ms
64 bytes from mad41s10-ln-f3.1e100.net (142.250.184.3): icmp_seq=13 ttl=116 time=6.24 ms
64 bytes from mad41s10-ln-f3.1e100.net (142.250.184.3): icmp_seq=14 ttl=116 time=6.24 ms
64 bytes from mad41s10-ln-f3.1e100.net (142.250.184.3): icmp_seq=15 ttl=116 time=6.24 ms
64 bytes from mad41s10-ln-f3.1e100.net (142.250.184.3): icmp_seq=16 ttl=116 time=6.24 ms
64 bytes from mad41s10-ln-f3.1e100.net (142.250.184.3): icmp_seq=17 ttl=116 time=6.24 ms
64 bytes from mad41s10-ln-f3.1e100.net (142.250.184.3): icmp_seq=18 ttl=116 time=6.24 ms
64 bytes from mad41s10-ln-f3.1e100.net (142.250.184.3): icmp_seq=19 ttl=116 time=6.24 ms
64 bytes from mad41s10-ln-f3.1e100.net (142.250.184.3): icmp_seq=20 ttl=116 time=6.24 ms

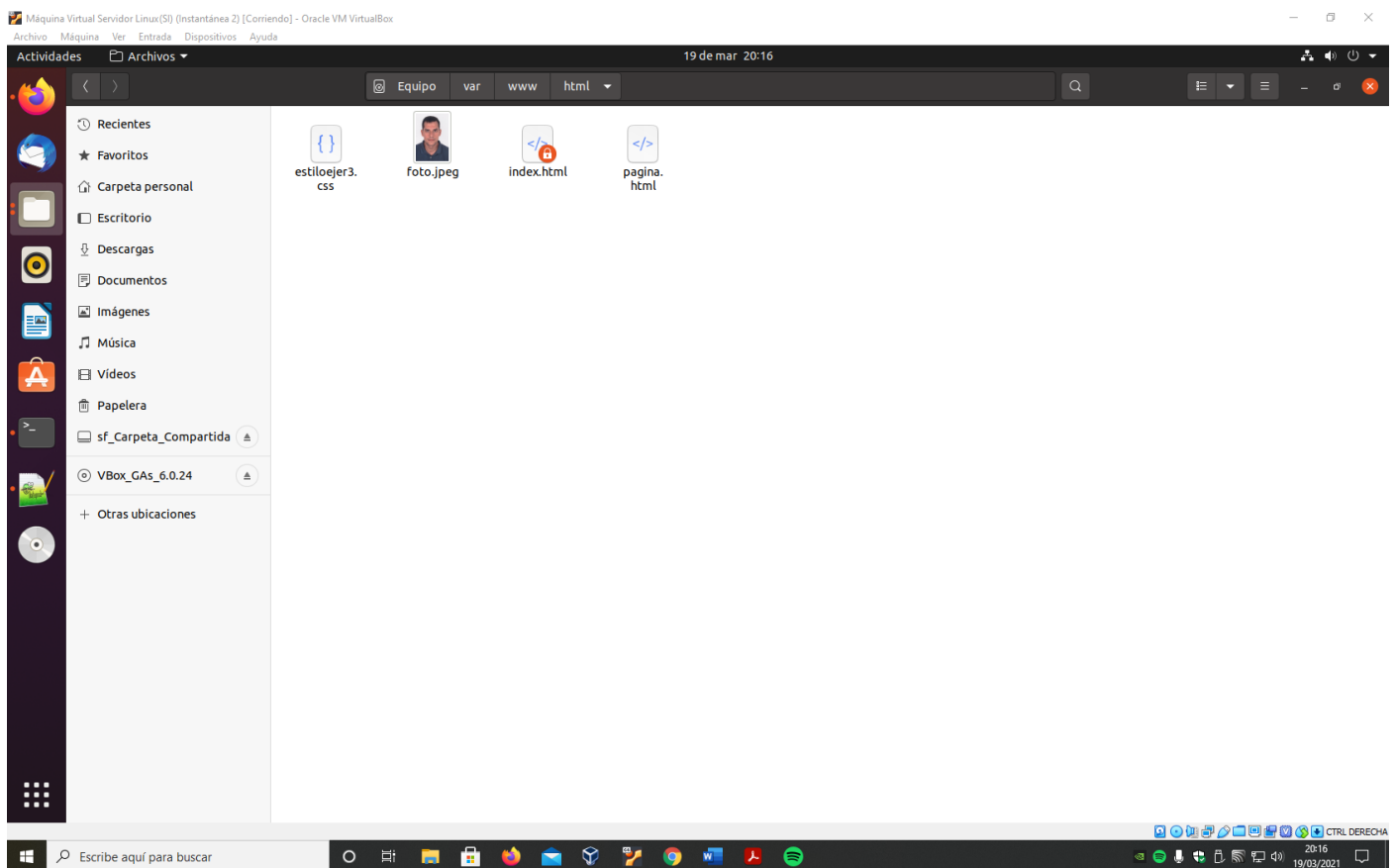
```

ACTIVIDAD 10.4

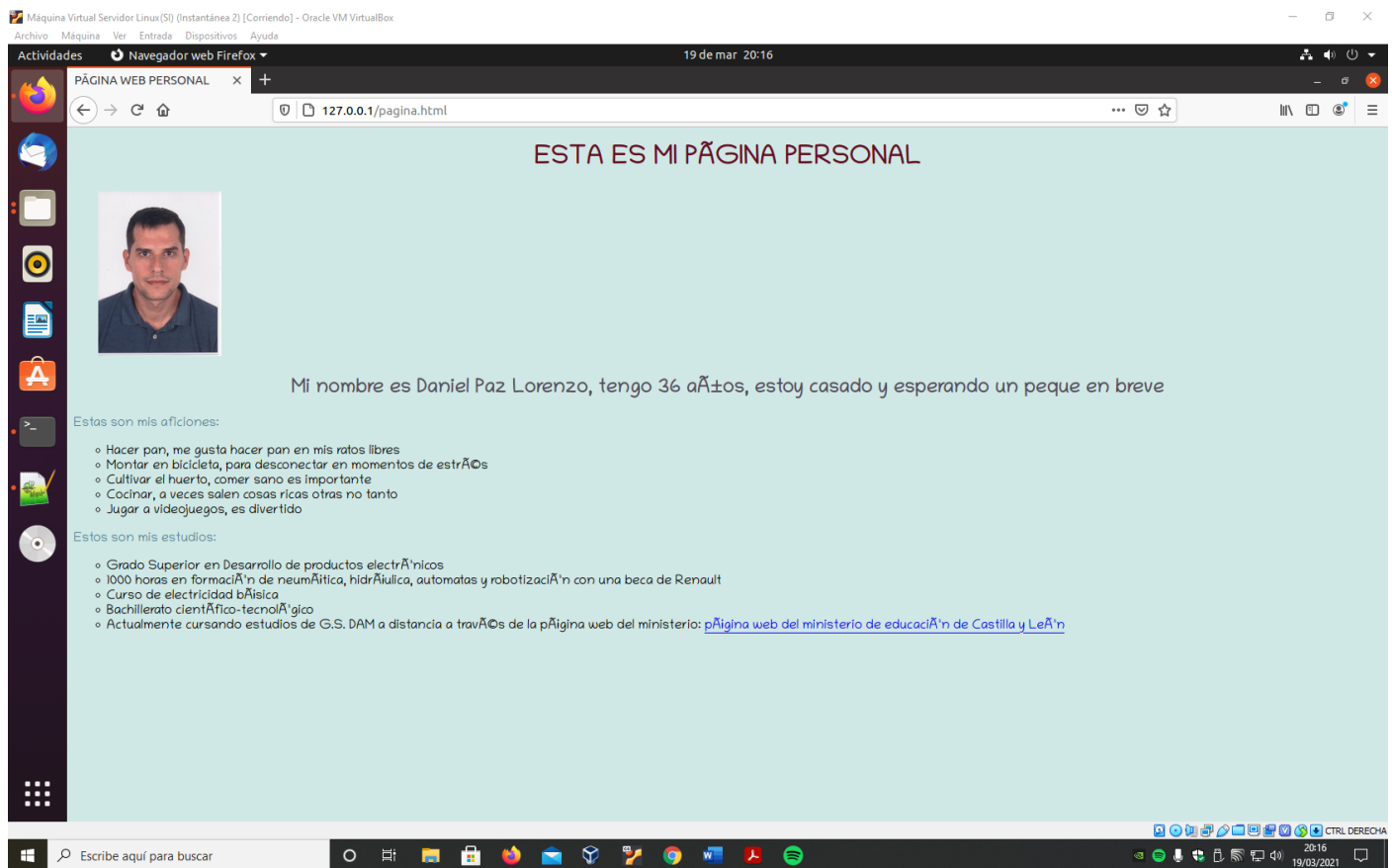
Instalamos el servidor “apache” a través de la línea de comandos del terminal en modo root con las siguientes instrucciones:

1. Desde el repositorio realizamos la instalación:
`root@daniel-UBUNTU:/home/daniel# apt-get install apache2`
2. Configuramos para que el servicio se inicie automáticamente:
`root@daniel-UBUNTU:/home/daniel# update-rc.d apache2 enable`
3. Iniciamos el servicio:
`root@daniel-UBUNTU:/home/daniel# service apache2 start`

Ahora vamos a la carpeta `/var/www/html` y allí creo los archivos de mi página web:



Entramos en el navegador y ponemos la dirección <http://127.0.0.1/pagina.html> (dirección localhost) nos mostrará la página web personal que he creado:



ACTIVIDAD 10.5

Instalaremos, como servicio de acceso remoto al servidor, VNC siguiendo los siguientes pasos:

1. En la máquina servidor vamos a instalar el servicio con el siguiente comando:

```
root@daniel-UBUNTU:/home/daniel# apt-get install tightvncserver
```
2. Vamos a generar una contraseña para cuando nos queramos conectar desde equipo remoto:

```
root@daniel-UBUNTU:/home/daniel# vncpasswd
```
3. Con el siguiente comando se crearán automáticamente los ficheros de configuración y se iniciará el servicio:

```
root@daniel-UBUNTU:/home/daniel# vncserver
```
4. Ya en la máquina remota instalaremos la aplicación cliente que nos va a permitir conectar con el servidor vnc utilizamos para ello el comando `apt-get install vinagre`. Una vez instalado el programa lo ejecutamos y en conexión indicamos 10.0.0.1:5901 que es la dirección del servidor junto con el puerto asociado por defecto para escritorio remoto. Introducimos la contraseña generada anteriormente y ya tenemos acceso al escritorio remoto de nuestro servidor como podemos ver en estas capturas de pantalla:

