

Redes de Ordenadores

PRÁCTICA 1 (PARTE I)

Alumno	Apellidos	Nombre	Curso
1	Escudero López	Gerardo	INSO 2C
2	Arriola Garcia	Ekaitz	MAIS 2A

Los objetivos de esta práctica son los siguientes:

I. DEMONIOS EN LINUX

1. Estudiar los servicios (demonios) en Linux y el proceso de conexión cliente-servidor.
2. Instalar, arrancar y administrar dos tipos de demonios en Linux:
 - a. Los demonios dependientes y gestionados por el súper demonio de red o súper servidor. Es este súper demonio el que escucha y arranca los demonios al llegar una petición al puerto correspondiente.
 - b. Se elegirán en este caso los demonios ftpd y telnetd, que deberán arrancarse mediante el súper demonio de red Inetd.
 - c. Los demonios *standalone*, llamados así porque funcionan de forma independiente del súper demonio de red.

Para ello instalaremos tres servicios: FTP/Telnet (ambos deberán arrancarse mediante el súper demonio de red Inetd) y SSH, (Open SSH)

II. USO DEL WIRESHARK COMO SNIFFER

1. Uso de un sniffer como el Wireshark para:
 - a. Identificar los paquetes del Three-way handshake.
 - b. Identificar las características más importantes de los paquetes enviados: direcciones IP origen y destino, flags, número de secuencia y número ack (Leer artículo de la revista Hackxcrack “1-port_scanning_hxc.pdf”).
2. Capturar los paquetes con el nombre de usuario y contraseña en una sesión ftp y telnet (no cifrada) y en una sesión ssh y sftp (cifrada) para ver sus datos, en este caso los usuarios y contraseñas:
 - a. En texto claro en telnet y en ftp.
 - b. Cifradas mediante claves pública y privada en SSH (ya explicaremos este método de cifrado en el curso más adelante).

III. USO DE NMAP PARA ESCANEAR PUERTOS

1. Aprender a usar nmap para escanear puertos.
2. Entender los diferentes tipos de escaneo que hay y el uso de los distintos flags en los paquetes TCP/IP.

ACTIVIDADES a REALIZAR

(En todos los comandos de toda la practica se dará por hecho que se están ejecutando como superusuario)

1. Instalar un servidor ftp en Linux. (1 punto)

a. ¿Cómo funciona y cómo se instala?

i. Lo lanza el súper demonio de red llamado **inetd**.

ii. Este súper demonio se instala con la orden:

apt install openssh-inetd

iii. Después se instala el demonio servidor ftp con la orden:

apt-get install ftp

También nos fijamos que en **/etc/inetd.conf** aparezca la línea

ftp stream tcp nowait root /usr/sbin/tcpd /usr/sbin/in.ftpd

la cual debería aparecer de forma automática.

```
#:STANDARD: These are standard services.  
ftp          stream  tcp     nowait  root    /usr/sbin/tcpd  /usr/sbin/in.ftpd
```

b. Arrancar el servidor ftp (que en realidad es arrancar el súper demonio de red) con la orden:

/etc/init.d/openssh-inetd start o **service openssh-inetd start**

c. ¿Cómo puedo saber si está arrancado o no?

i. Mirando los procesos que están ejecutándose en el sistema mediante la orden:

ps -ef | grep "inetd"

```
root@ekaitz-VirtualBox:/home/ekaitz# ps -ef | grep "inetd"  
root      2719      1  0 19:56 ?        00:00:00 /usr/sbin/inetd  
root      2741    2585  0 19:56 pts/0    00:00:00 grep --color=auto inetd
```

En la captura podemos ver dos procesos. El primero es **inetd**, el cual es proceso hijo del proceso **init**. El segundo es la búsqueda que estamos realizando.

Otra forma es usando uno de los siguientes comandos para ver si esta encendido:

systemctl status openbsd-inetd

service --status-all | grep inet | grep +

En este último solo aparecerá algo si esta encendido, pero si no esta instalado tampoco saldra nada. Para ver el estado del servicio, bastaría con **service --status-all | grep inet**, y aparecería [-] si esta apagado y [+] si esta encendido.

ii. Mediante nmap:

nmap 127.0.0.1

```
veranono29@veranono29:~$ nmap 127.0.0.1
Starting Nmap 7.80 ( https://nmap.org ) at 2020-10-19 19:47 CEST
Nmap scan report for localhost (127.0.0.1)
Host is up (0.000053s latency).
Not shown: 996 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
23/tcp    open  telnet
80/tcp    open  http
631/tcp   open  ipp

Nmap done: 1 IP address (1 host up) scanned in 0.08 seconds
```

Otra forma parecida es con:

netstat -ltp

o para ver los puertos en vez de los nombres:

netstat -ltpn

```
root@ekaitz-VirtualBox:/home/ekaitz# netstat -ltp
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 0.0.0.0:ftp             0.0.0.0:*               LISTEN      5505/inetd
tcp        0      0 localhost:domain        0.0.0.0:*               LISTEN      533/systemd-resolve
tcp        0      0 0.0.0.0:telnet          0.0.0.0:*               LISTEN      5505/inetd
tcp        0      0 localhost:ipp           0.0.0.0:*               LISTEN      576/cupsd
tcp6       0      0 [::]:http               [::]:*                 LISTEN      700/apache2
tcp6       0      0 ip6-localhost:ipp      [::]:*                 LISTEN      576/cupsd
root@ekaitz-VirtualBox:/home/ekaitz# netstat -ltpn
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 0.0.0.0:21             0.0.0.0:*               LISTEN      5505/inetd
tcp        0      0 127.0.0.53:53          0.0.0.0:*               LISTEN      533/systemd-resolve
tcp        0      0 0.0.0.0:23             0.0.0.0:*               LISTEN      5505/inetd
tcp        0      0 127.0.0.1:631          0.0.0.0:*               LISTEN      576/cupsd
tcp6       0      0 :::80                  :::*                   LISTEN      700/apache2
tcp6       0      0 :::1:631                :::*                   LISTEN      576/cupsd
```

- iii. Mediante un telnet al puerto del servidor ftp mediante la orden: (ejecutandolo desde el servidor)

telnet 127.0.0.1

O en vez de usar una dirección de loopback (127.0.0.0/8), usando la dirección ip del servidor si se ejecutase desde otro cliente.

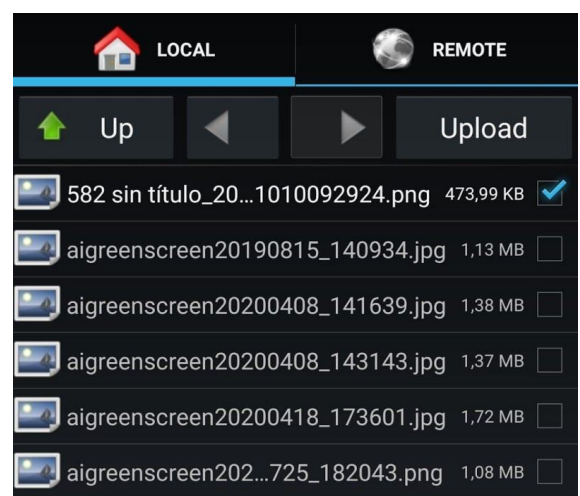
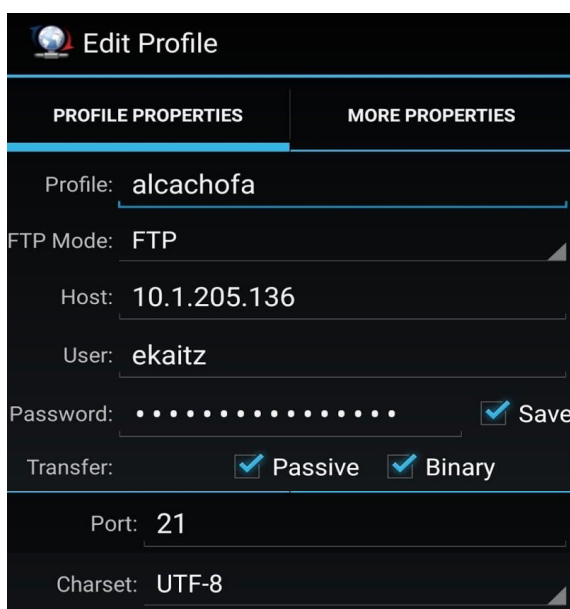
- d. ¿Qué puerto utiliza este servidor?

El servicio de ftp utiliza el puerto 21, el que es usado para conectarse de forma remota al servidor y autenticarse, y el puerto 20, para transferir archivos una vez se haya autenticado.

2. **Instalar un cliente ftp en el móvil o en otra máquina virtual diferente a la del servidor FTP (ojo que no sea SFTP, que sea FTP). Enviar una foto desde el móvil al servidor FTP. (1 punto)**

- a. Mostrar el fichero de la foto en el servidor FTP

Primero editamos la conexión (host, user, password, puerto, modo de transferencia), y una vez establecida la conexión seleccionamos una imagen y la enviamos.



Y posteriormente comprobamos que, efectivamente, la imagen aparece en el servidor FTP.

```
root@ekaitz-VirtualBox:/home/ekaitz# ls
'582 sin titulo_20201010092924.png' Desktop Downloads Pictures Templates
cuaderno Documents Music Public Videos
```

- b. Otra opción sería desde cualquier shell mediante la orden
ftp <ip> y después poniendo las credenciales.
send y después las rutas (local y remoto) de la imagen.
bye para terminar la conexión.

```
ekaitz@ekaitz-VirtualBox:~$ ftp 192.168.1.56
Connected to 192.168.1.56.
220 uno FTP server (Version 6.4/OpenBSD/Linux-ftp-0.17) ready.
Name (192.168.1.56:ekaitz): ekaitz
331 Password required for ekaitz.
Password:
230 User ekaitz logged in.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> send
(local-file) pato.png
(remote-file) pato.png
local: pato.png remote: pato.png
200 PORT command successful.
150 Opening BINARY mode data connection for 'pato.png'.
226 Transfer complete.
485370 bytes sent in 0.00 secs (123.7991 MB/s)
ftp> bye
221 Goodbye.
```

De igual modo, comprobamos que la imagen se encuentra en el servidor FTP.

```
ekaitz@uno:~$ ls
'582 sin titulo_20201010092924.png' cuaderno obviomunix.jpeg Templates
alcachofa Desktop patata Videos
common.h Documents pato.png
cpu Downloads Pictures
cpu.c Music Public
```

3. Monitorización con un sniffer (1 punto)

a. Una vez que se ha logrado la conexión cliente-servidor, capturar mediante el wireshark los siguientes paquetes:

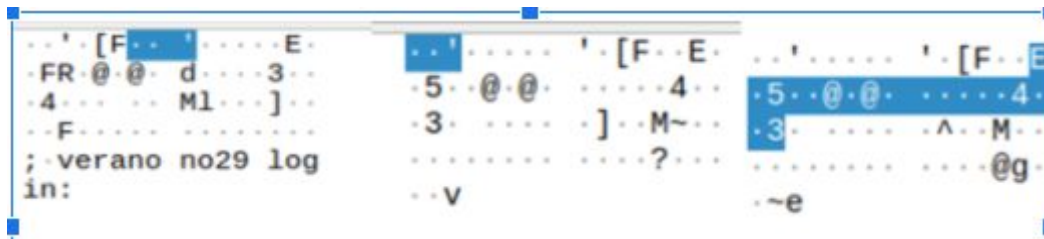
i. Three-way handshake - Tres paquetes con los que se establece la conexión cliente-servidor -

No.	Time	Source	Destination	Protocol	Length	Info
11	5.095375755	192.168.1.52	192.168.1.51	TCP	74	39968 → 23 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 T...
12	5.099016286	192.168.1.51	192.168.1.52	TCP	74	23 → 39968 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SA...
13	5.099035148	192.168.1.52	192.168.1.51	TCP	66	39968 → 23 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=4041157465...

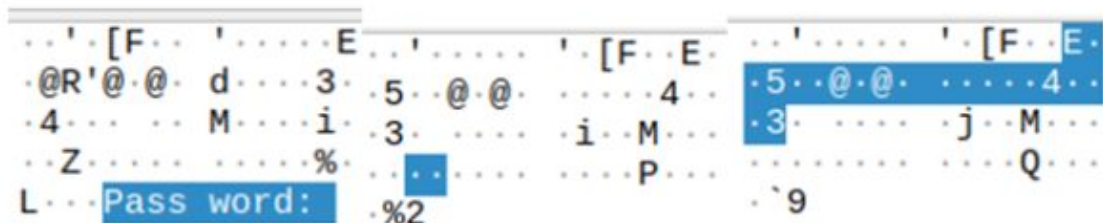
ii. Pantallazo donde aparezcan los puertos del cliente y el servidor, junto con sus direcciones IP.

Frame 11: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface enp0s3, id 0
Ethernet II, Src: PcsCompu_fc:5b:46 (08:00:27:fc:5b:46), Dst: PcsCompu_9e:19:99 (08:00:27:9e:19:99)
Internet Protocol Version 4, Src: 192.168.1.52, Dst: 192.168.1.51
Transmission Control Protocol, Src Port: 39968, Dst Port: 23, Seq: 0, Len: 0
Source Port: 39968
Destination Port: 23
[Stream index: 0]

iii. USUARIO Y CONTRASEÑA - usuario veranono29



Contraseña: 29*****



iv. Capturar los paquetes con los que se cierra la conexión -

169	23.550951209	192.168.1.52	192.168.1.51	TCP	66 39968 → 23 [ACK] Seq=180 Ack=1061 Win=64128 Len=0 TSval=40411...
171	23.579986902	192.168.1.51	192.168.1.52	TCP	66 23 → 39968 [FIN, ACK] Seq=1061 Ack=180 Win=65152 Len=0 TSval=...
172	23.580059193	192.168.1.52	192.168.1.51	TCP	66 39968 → 23 [FIN, ACK] Seq=180 Ack=1062 Win=64128 Len=0 TSval=...
173	23.580220991	192.168.1.51	192.168.1.52	TCP	66 23 → 39968 [ACK] Seq=1062 Ack=181 Win=65152 Len=0 TSval=21915...

4. Escaneo de los puertos mediante nmap (1 punto)

-nmap -sT -sU (todos los puertos TCP con conexión + los puertos UDP)

- Hacer un escaneo **FULL SCAN** al servidor donde está el servicio ftp mediante nmap

```
veranono29@veranono29-SegMaquina:~$ sudo nmap -sT -sU 127.0.0.1
[sudo] contraseña para veranono29:
Starting Nmap 7.80 ( https://nmap.org ) at 2020-10-28 09:19 CET
Nmap scan report for localhost (127.0.0.1)
Host is up (0.00014s latency).
Not shown: 1995 closed ports
PORT      STATE      SERVICE
21/tcp    open      ftp
23/tcp    open      telnet
631/tcp   open      ipp
631/udp   open|filtered ipp
5353/udp  open|filtered zeroconf

Nmap done: 1 IP address (1 host up) scanned in 1.30 seconds
```

- Identificar mediante wireshark y los filtros necesarios los paquetes mandados en ese escaneo a ese puerto en concreto. Para ello, debe aparecer:

- Un escaneo filtrado con éxito (a un puerto abierto)

```
veranono29@veranono29-SegMaquina:~$ nmap -p 21 127.0.0.1
Starting Nmap 7.80 ( https://nmap.org ) at 2020-10-28 09:24 CET
Nmap scan report for localhost (127.0.0.1)
Host is up (0.00080s latency).

PORT      STATE      SERVICE
21/tcp    open      ftp

Nmap done: 1 IP address (1 host up) scanned in 0.05 seconds
```


No.	Time	Source	Destination	Protocol	Length	Info
3	5.795933393	192.168.1.52	192.168.1.51	TCP	74	42236 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 T
4	5.795985073	192.168.1.52	192.168.1.51	TCP	74	54290 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 T
5	5.796227505	192.168.1.51	192.168.1.52	TCP	74	80 → 42236 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SA
6	5.796239363	192.168.1.52	192.168.1.51	TCP	66	42236 → 80 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=3205910950
7	5.796248657	192.168.1.51	192.168.1.52	TCP	60	443 → 54290 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
8	5.796276095	192.168.1.52	192.168.1.51	TCP	66	42236 → 80 [RST, ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=32059
11	5.804488951	192.168.1.52	192.168.1.51	TCP	74	39138 → 21 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 T
12	5.804721621	192.168.1.51	192.168.1.52	TCP	74	21 → 39138 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SA
13	5.804730909	192.168.1.52	192.168.1.51	TCP	66	39138 → 21 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=3205910959
14	5.804762853	192.168.1.52	192.168.1.51	TCP	66	39138 → 21 [RST, ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=32059

Frame 3: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface enp0s3, id 0
 Ethernet II, Src: PcsCompu_fc:5b:46 (08:00:27:fc:5b:46), Dst: PcsCompu_9e:19:99 (08:00:27:9e:19:99)
 Internet Protocol Version 4, Src: 192.168.1.52, Dst: 192.168.1.51
 Transmission Control Protocol, Src Port: 42236, Dst Port: 80, Seq: 0, Len: 0

ii. Un escaneo filtrado a un puerto cerrado

```
veranono29@veranono29-SegMaquina:~$ nmap -p 18 127.0.0.1
Starting Nmap 7.80 ( https://nmap.org ) at 2020-10-28 09:23 CET
Nmap scan report for localhost (127.0.0.1)
Host is up (0.000048s latency).

PORT      STATE SERVICE
18/tcp    closed msp

Nmap done: 1 IP address (1 host up) scanned in 0.06 seconds
```

No.	Time	Source	Destination	Protocol	Length	Info
261	5.851086872	192.168.1.52	192.168.1.51	TCP	74	42252 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 T
262	5.851137287	192.168.1.52	192.168.1.51	TCP	74	54306 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 T
263	5.851373429	192.168.1.51	192.168.1.52	TCP	74	80 → 42252 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SA
264	5.851384757	192.168.1.52	192.168.1.51	TCP	66	42252 → 80 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=3206376691
265	5.851393873	192.168.1.51	192.168.1.52	TCP	60	443 → 54306 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
266	5.851421717	192.168.1.52	192.168.1.51	TCP	66	42252 → 80 [RST, ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=32063
269	5.918923665	192.168.1.52	192.168.1.51	TCP	74	59714 → 24 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 T
270	5.919250012	192.168.1.51	192.168.1.52	TCP	60	24 → 59714 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0

Si al nmap le añadimos la opción **-O** (o mayúscula), nos mostrará el sistema operativo de la víctima. En este caso vemos que es un Unix.

NOTA: El resultado de estos pantallazos deben reflejar lo mismo que las diapositivas que hay en la teoría donde se explica este tipo de escaneo.

5. Instalar un servidor telnet en Linux. (1 punto)

a. ¿Cómo funciona y cómo se instala?

- Lo lanza el súper demonio de red llamado **inetd**.
- Este súper demonio se instala con la orden:

apt install openssh-inetd

iii. Después se instala el demonio servidor telnet con la orden:

apt-get install telnetd

También nos fijamos que en **/etc/inetd.conf** aparezca la línea

telnet stream tcp nowait telnetd /usr/sbin/tcpd /usr/sbin/in.telnetd

```
#:STANDARD: These are standard services.  
ftp          stream tcp    nowait root    /usr/sbin/tcpd  /usr/sbin/in.ftpd  
telnet       stream tcp    nowait telnetd /usr/sbin/tcpd  /usr/sbin/in.telnetd
```

la cual debería aparecer de forma automática.

b. Arrancar el servidor telnet (que en realidad es arrancar el súper demonio de red) con la orden

/etc/init.d/openbsd-inetd start o **service openbsd-inetd start**

c. ¿Cómo puedo saber si está arrancado o no?

i. Mirando los procesos que están ejecutándose en el sistema mediante la orden:

ps -ef | grep "inetd"

```
root@ekaitz-VirtualBox:/home/ekaitz# ps -ef | grep "inetd"  
root      2719      1  0 19:56 ?        00:00:00 /usr/sbin/inetd  
root      2741    2585  0 19:56 pts/0    00:00:00 grep --color=auto inetd
```

En la captura podemos ver dos procesos. El primero es inetd, el cual es proceso hijo del proceso init. El segundo es la búsqueda que estamos realizando.

Otra forma es usando uno de los siguientes comandos para ver si está encendido:

systemctl status openbsd-inetd

service --status-all | grep inet | grep +

En este último solo aparecerá algo si está encendido, pero si no está instalado tampoco saldrá nada. Para ver el estado del servicio, bastaría con **service --status-all | grep inet**, y aparecerá **[-]** si está apagado y **[+]** si está encendido.

ii. Mediante nmap:

nmap 127.0.0.1

```
veranono29@veranono29:~$ nmap 127.0.0.1
Starting Nmap 7.80 ( https://nmap.org ) at 2020-10-19 19:47 CEST
Nmap scan report for localhost (127.0.0.1)
Host is up (0.000053s latency).
Not shown: 996 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
23/tcp    open  telnet
80/tcp    open  http
631/tcp    open  ipp

Nmap done: 1 IP address (1 host up) scanned in 0.08 seconds
```

Otra forma parecida es con:

netstat -ltp

o para ver los puertos en vez de los nombres:

netstat -ltpn

```
root@ekaitz-VirtualBox:/home/ekaitz# netstat -ltp
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 0.0.0.0:ftp             0.0.0.0:*               LISTEN      5505/inetd
tcp        0      0 localhost:domain        0.0.0.0:*               LISTEN      533/systemd-resolve
tcp        0      0 0.0.0.0:telnet          0.0.0.0:*               LISTEN      5505/inetd
tcp        0      0 localhost:ipp           0.0.0.0:*               LISTEN      576/cupsd
tcp6       0      0 [::]:http              [::]:*                  LISTEN      700/apache2
tcp6       0      0 ip6-localhost:ipp      [::]:*                  LISTEN      576/cupsd
root@ekaitz-VirtualBox:/home/ekaitz# netstat -ltpn
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 0.0.0.0:21             0.0.0.0:*               LISTEN      5505/inetd
tcp        0      0 127.0.0.53:53          0.0.0.0:*               LISTEN      533/systemd-resolve
tcp        0      0 0.0.0.0:23             0.0.0.0:*               LISTEN      5505/inetd
tcp        0      0 127.0.0.1:631          0.0.0.0:*               LISTEN      576/cupsd
tcp6       0      0 :::80                  :::*                    LISTEN      700/apache2
tcp6       0      0 :::1:631               :::*                    LISTEN      576/cupsd
```

iii. Mediante un telnet al puerto del servidor ftp mediante la orden: (ejecutandolo desde el servidor)

telnet 127.0.0.1

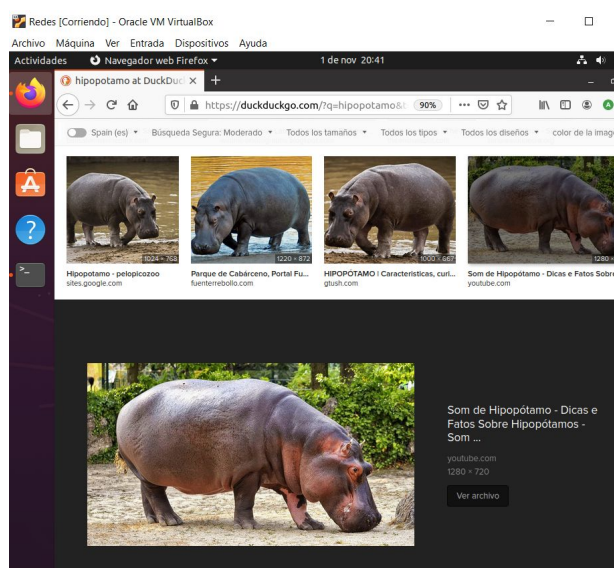
O en vez de usar una dirección de loopback (127.0.0.0/8), usando la dirección ip del servidor si se ejecutase desde otro cliente.

d. ¿Qué puerto utiliza este servidor?

Telnet usa el puerto 23 por defecto.

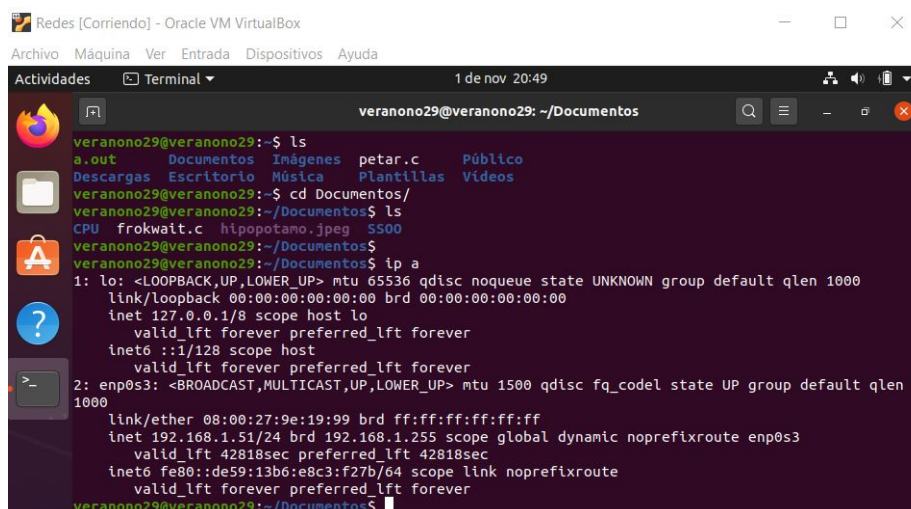
6. Acceder desde un cliente telnet desde otra máquina virtual. Una vez dentro del servidor borrar la imagen enviada en el punto anterior (la que enviamos con el FTP) **(1 punto)**

DESCARGAMOS LA FOTO



```
veranono29@veranono29:~/Documentos$ ls
CPU  frokwait.c  hipopotamo.jpeg  SS00
```

PILLAMOS LA IP



CONECTAMOS A TRAVÉS DEL TELNET

```

Redes 2 [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
Actividades Terminal 1 de nov 20:52
veranono29@veranono29: ~
veranono29@veranono29-SegMaquina:~$ telnet 192.168.1.51
Trying 192.168.1.51...
Connected to 192.168.1.51.
Escape character is '^]'.
Ubuntu 20.04.1 LTS
veranono29 login: veranono29
Password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-52-generic x86_64)

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

39 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: Sun Nov  1 16:46:16 CET 2020 from 192.168.1.52 on pts/1
veranono29@veranono29:~$

```

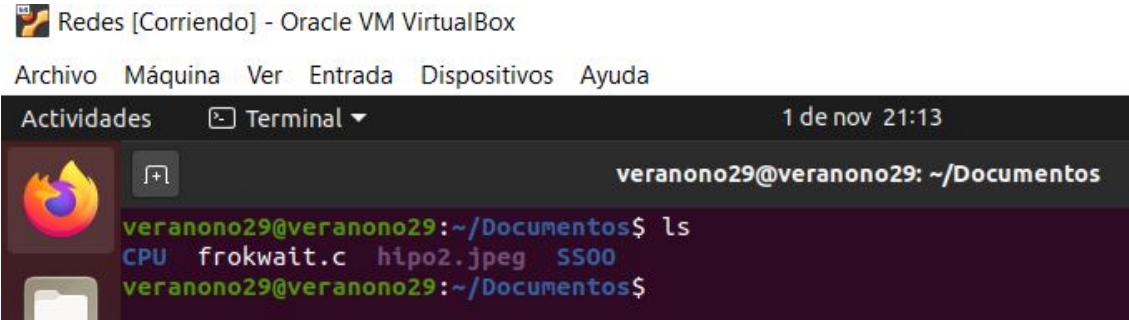
BORRAMOS LA IMAGEN DESCARGADA

```

Redes 2 [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
Actividades Terminal 1 de nov 21:09
veranono29@veranono29: ~/Documentos
veranono29@veranono29:~/Documentos$ ls
CPU frokwalt.c hipo2.jpeg hipo2.jpeg SS00
veranono29@veranono29:~/Documentos$ man rm
veranono29@veranono29:~/Documentos$ ffff
Orden «ffff» no encontrada. Quizá quiso decir:
la orden «ftff» del paquete deb «whichman (2.4-9)»
Pruebe con: sudo apt install <nombre del paquete deb>
veranono29@veranono29:~/Documentos$ cp hipo2.jpeg hipo2.jpeg
cp: no se puede efectuar 'stat' sobre 'hipo2.jpeg': No existe el archivo o el directorio
veranono29@veranono29:~/Documentos$ cp hipo2.jpeg hipo2.jpeg
veranono29@veranono29:~/Documentos$ ls
CPU frokwalt.c hipo2.jpeg hipo2.jpeg SS00
veranono29@veranono29:~/Documentos$ rm hipo
hipo2.jpeg hipo2.jpeg
veranono29@veranono29:~/Documentos$ rm hipo2.jpeg
veranono29@veranono29:~/Documentos$

```

CONFIRMAMOS QUE SE HAN BORRADO



“hipo2.jpeg” es una copia de “hipopotamo.jpeg” que se realizó de antes de eliminar esta última. En esta imagen se puede observar como, en efecto, se eliminó correctamente “hipopotamo.jpeg”.

7. Monitorización con un sniffer (1 punto)

a. Una vez que se ha logrado la conexión cliente-servidor, capturar mediante el wireshark los siguientes paquetes:

i. Three-way handshake - Tres paquetes con los que se establece la conexión cliente-servidor -

No.	Time	Source	Destination	Protocol	Length	Info
63	28.061123144	192.168.1.52	192.168.1.51	TCP	74	57678 → 23 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 T...
64	28.061388038	192.168.1.51	192.168.1.52	TCP	74	23 → 57678 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SA...
65	28.061401837	192.168.1.52	192.168.1.51	TCP	66	57678 → 23 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=4266733446...

ii. Imagen donde aparecen los puertos del cliente y el servidor, junto con sus direcciones IP.

No.	Time	Source	Destination	Protocol	Length	Info
63	28.061123144	192.168.1.52	192.168.1.51	TCP	74	57678 → 23 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 T...
64	28.061388038	192.168.1.51	192.168.1.52	TCP	74	23 → 57678 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SA...
65	28.061401837	192.168.1.52	192.168.1.51	TCP	66	57678 → 23 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=4266733446...
66	28.061545862	192.168.1.52	192.168.1.51	TELNET	93	Telnet Data ...
67	28.061705375	192.168.1.51	192.168.1.52	TCP	66	23 → 57678 [ACK] Seq=1 Ack=28 Win=65152 Len=0 TSval=386746964...
68	28.196966606	192.168.1.51	192.168.1.52	TELNET	78	Telnet Data ...
69	28.196978271	192.168.1.52	192.168.1.51	TCP	66	57678 → 23 [ACK] Seq=28 Ack=13 Win=64256 Len=0 TSval=42667335...
70	28.197247078	192.168.1.51	192.168.1.52	TELNET	105	Telnet Data ...
71	28.197252036	192.168.1.52	192.168.1.51	TCP	66	57678 → 23 [ACK] Seq=28 Ack=52 Win=64256 Len=0 TSval=42667335...
72	28.197323703	192.168.1.52	192.168.1.51	TELNET	179	Telnet Data ...
73	28.197400050	192.168.1.51	192.168.1.52	TCP	66	23 → 57678 [ACK] Seq=52 Ack=141 Win=65152 Len=0 TSval=3867469...
74	28.197675174	192.168.1.51	192.168.1.52	TELNET	69	Telnet Data ...
75	28.102670585	192.168.1.52	192.168.1.51	TCP	66	57678 → 23 [ACK] Seq=141 Ack=55 Win=64256 Len=0 TSval=42667335...

Frame 66: 93 bytes on wire (744 bits), 93 bytes captured (744 bits) on interface enp0s3, id 0

Ethernet II, Src: PcsCompu, f0:5b:4b:00:00:27, f0:5b:4b:00:00:27, Dst: PcsCompu, f0:5b:4b:00:00:27, f0:5b:4b:00:00:27

Internet Protocol Version 4, Src: 192.168.1.52, Dst: 192.168.1.51

Transmission Control Protocol, Src Port: 57678, Dst Port: 23, Seq: 1, Ack: 1, Len: 27

Source Port: 57678

Destination Port: 23

[Stream index: 1]

[TCP Segment Len: 27]

iii. USUARIO Y CONTRASEÑA -

84	28	229809628	192.168.1.51	192.168.1.52	TELNET	84 Telnet Data ...
85	28	229823897	192.168.1.52	192.168.1.51	TCP	66 57678 → 23 [ACK] Seq=147 Ack=96 Win=64256 Len=0 TSval=4266733...
97	32	058098335	192.168.1.52	192.168.1.51	TELNET	67 Telnet Data ...
98	32	058479520	192.168.1.51	192.168.1.52	TCP	66 23 → 57678 [ACK] Seq=96 Ack=148 Win=65152 Len=0 TSval=3867473...
99	32	058584852	192.168.1.51	192.168.1.52	TELNET	67 Telnet Data ...
100	32	058591709	192.168.1.52	192.168.1.51	TCP	66 57678 → 23 [ACK] Seq=148 Ack=97 Win=64256 Len=0 TSval=4266737...
101	32	203800688	192.168.1.52	192.168.1.51	TELNET	67 Telnet Data ...
▶ Frame 84: 84 bytes on wire (672 bits), 84 bytes captured (672 bits) on interface enp0s3, id 0 ▶ Ethernet II, Src: PcsCompu_9e:19:99 (08:00:27:9e:19:99), Dst: PcsCompu_fc:5b:46 (08:00:27:fc:5b:46) ▶ Internet Protocol Version 4, Src: 192.168.1.51, Dst: 192.168.1.52 ▶ Transmission Control Protocol, Src Port: 23, Dst Port: 57678, Seq: 78, Ack: 147, Len: 18 Source Port: 23 Destination Port: 57678 [Stream index: 1] [TCP Segment Len: 18] Sequence number: 78 (relative sequence number) Sequence number (raw): 1580894094 [Next sequence number: 96 (relative sequence number)] Acknowledgment number: 147 (relative ack number) Acknowledgment number (raw): 366912505 1000 ... = Header Length: 32 bytes (8)						
0000	08	00	27	fc	5b	46 08 00 27 9e 19 99 08 00 45 10 ... [F...E..
0010	00	46	02	5f	40	00 40 06 b4 8b c0 a8 01 33 c0 a8 ...F...@...3..
0020	01	34	00	17	e1	4e 5e 3a 87 8e 15 de a3 f9 80 18 ...4...N^:.....
0030	01	fd	47	cb	00	00 01 01 08 0a e6 84 e7 f4 fe 51 ...G.....Q
0040	30	10	76	65	72	61 6e 6f 6e 6f 32 39 20 6c 6f 67 0 verano no29 log
0050	69	6e	3a	20		in:

Usuario:

```

..'.[F..E..
5d@.@ R...4..
3.N...^:
.....Q?#
..v

```

Contraseña:

```

..'.[F..E..
@l@.@...3..
4...N^:
..C.....i.Q
M...Pass word:

```

```

..'.[F..E..
5d(@.@ R...4..
3.N...^:
.....QR
i2

```

iv. Capturar los paquetes con los que se cierra la conexión -

739	57	429624305	192.168.1.52	192.168.1.51	TCP	66 57678 → 23 [ACK] Seq=176 Ack=773 Win=64128 Len=0 TSval=426676...
740	57	483547846	192.168.1.51	192.168.1.52	TCP	66 23 → 57678 [FIN, ACK] Seq=773 Ack=176 Win=65152 Len=0 TSval=3...
741	57	483698494	192.168.1.52	192.168.1.51	TCP	66 57678 → 23 [FIN, ACK] Seq=176 Ack=774 Win=64128 Len=0 TSval=4...
742	57	483889544	192.168.1.51	192.168.1.52	TCP	66 23 → 57678 [ACK] Seq=774 Ack=177 Win=65152 Len=0 TSval=386749...

8. Escaneo de los puertos mediante nmap (1 punto)

- a. Hacer un escaneo **HALF SCAN** al servidor donde está el servicio ftp/telnet mediante nmap

-nmap -sS <ip> es para hacer un escaneo “encubierto” (sin responder a ACK, SYN).

```
veranono29@veranono29-SegMaquina:~$ sudo nmap -sS 127.0.0.1
Starting Nmap 7.80 ( https://nmap.org ) at 2020-10-28 09:28 CET
Nmap scan report for localhost (127.0.0.1)
Host is up (0.0000020s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
23/tcp    open  telnet
631/tcp   open  ipp

Nmap done: 1 IP address (1 host up) scanned in 0.07 seconds
```

- b. Identificar mediante wireshark y los filtros necesarios los paquetes mandados en ese escaneo a ese puerto en concreto. Para ello, debe aparecer:

- i. Un escaneo filtrado con éxito (a un puerto abierto)

```
veranono29@veranono29-SegMaquina:~$ sudo nmap -sS -p 21 192.168.1.51
Starting Nmap 7.80 ( https://nmap.org ) at 2020-10-29 19:57 CET
Nmap scan report for 192.168.1.51
Host is up (0.00030s latency).

PORT      STATE SERVICE
21/tcp    open  ftp
MAC Address: 08:00:27:9E:19:99 (Oracle VirtualBox virtual NIC)
```

(((ip.src==192.168.1.52)&&(ip.dst==192.168.1.51))(((ip.src==192.168.1.51)&&(ip.dst==192.168.1.52)))						
No.	Time	Source	Destination	Protocol	Length	Info
19	3.106616422	192.168.1.52	192.168.1.51	TCP	58	46083 → 21 [SYN] Seq=0 Win=0 Len=0 MSS=1460
20	3.106896296	192.168.1.51	192.168.1.52	TCP	60	21 → 46083 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460
21	3.106905491	192.168.1.52	192.168.1.51	TCP	54	46083 → 21 [RST] Seq=1 Win=0 Len=0

ii. Un escaneo filtrado a un puerto cerrado

```
veranono29@veranono29-SegMaquina:~$ sudo nmap -ss -p 24 192.168.1.51
[sudo] contraseña para veranono29:
Starting Nmap 7.80 ( https://nmap.org ) at 2020-10-29 19:55 CET
Nmap scan report for 192.168.1.51
Host is up (0.00030s latency).

PORT      STATE SERVICE
24/tcp    closed priv-mail
MAC Address: 08:00:27:9E:19:99 (Oracle VirtualBox virtual NIC)
```

(((ip.src==192.168.1.52)&&(ip.dst==192.168.1.51))(((ip.src==192.168.1.51)&&(ip.dst==192.168.1.52)))						
No.	Time	Source	Destination	Protocol	Length	Info
300	26.070309603	192.168.1.52	192.168.1.51	TCP	58	53155 → 24 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
301	26.070577095	192.168.1.51	192.168.1.52	TCP	60	24 → 53155 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0

NOTA: La única diferencia es que hay que hacer un escaneo HALF SCAN para el servidor y ver los paquetes (en especial los flags que están activados) que se mandan en este caso.

- Realizar el paso 3 (captura del Three-way handshake, fin de conexión y usuario y contraseña) para el servidor Open SSH, con sus correspondientes pantallazos. ¿Has podido ver la contraseña en Wireshark? ¿Qué paquetes son los que definen la autenticación? Señálalos en tu captura. (1 punto)

El cliente de SSH se instala con **apt-get install openssh-client**.

Realizamos la conexión mediante SSH:

```
veranono29@veranono29-SegMaquina:~$ ssh 192.168.1.51
The authenticity of host '192.168.1.51 (192.168.1.51)' can't be established.
ECDSA key fingerprint is SHA256:Mn1tHaveumlwa5DXoiuxqjpyTb3n8GnrrbiacA2Vt5E.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.1.51' (ECDSA) to the list of known hosts.
veranono29@192.168.1.51's password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-48-generic x86_64)

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

0 actualizaciones se pueden instalar inmediatamente.
0 de estas actualizaciones son una actualización de seguridad.

Your Hardware Enablement Stack (HWE) is supported until April 2025.
Last login: Thu Oct 29 16:36:50 2020 from 192.168.1.52
veranono29@veranono29:~$
```


THREE-WAY HANDSHAKE

No.	Time	Source	Destination	Protocol	Length	Info
7	7.506238084	192.168.1.52	192.168.1.51	TCP	74	43986 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 T...
8	7.506577559	192.168.1.51	192.168.1.52	TCP	74	22 → 43986 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SA...
9	7.506589842	192.168.1.52	192.168.1.51	TCP	66	43986 → 22 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=3207318695...

TAMBIÉN SE VE QUE LA COMUNICACIÓN ES CIFRADA

((ip.src==192.168.1.52)&&(ip.dst==192.168.1.51))(((ip.src==192.168.1.51)&&(ip.dst==192.168.1.52)))						
No.	Time	Source	Destination	Protocol	Length	Info
56	21.392635800	192.168.1.51	192.168.1.52	SSHv2	94	Server: Encrypted packet (len=28)
57	21.392650226	192.168.1.52	192.168.1.51	TCP	66	43986 → 22 [ACK] Seq=1878 Ack=1730 Win=64128 Len=0 TSval=3207...
58	21.392832904	192.168.1.52	192.168.1.51	SSHv2	178	Client: Encrypted packet (len=112)
59	21.435705083	192.168.1.51	192.168.1.52	TCP	66	22 → 43986 [ACK] Seq=1730 Ack=1990 Win=64128 Len=0 TSval=1056...
61	21.839411467	192.168.1.51	192.168.1.52	SSHv2	694	Server: Encrypted packet (len=628)
62	21.881695700	192.168.1.52	192.168.1.51	TCP	66	43986 → 22 [ACK] Seq=1990 Ack=2358 Win=64128 Len=0 TSval=3207...
63	21.881977836	192.168.1.51	192.168.1.52	SSHv2	110	Server: Encrypted packet (len=44)
64	21.881985143	192.168.1.52	192.168.1.51	TCP	66	43986 → 22 [ACK] Seq=1990 Ack=2402 Win=64128 Len=0 TSval=3207...
65	21.882098065	192.168.1.52	192.168.1.51	SSHv2	526	Client: Encrypted packet (len=460)
66	21.882353113	192.168.1.51	192.168.1.52	TCP	66	22 → 43986 [ACK] Seq=2402 Ack=2450 Win=64128 Len=0 TSval=1056...
67	21.894957134	192.168.1.51	192.168.1.52	SSHv2	174	Server: Encrypted packet (len=108)
68	21.894969702	192.168.1.52	192.168.1.51	TCP	66	43986 → 22 [ACK] Seq=2450 Ack=2510 Win=64128 Len=0 TSval=3207...
69	21.903453855	192.168.1.51	192.168.1.52	SSHv2	566	Server: Encrypted packet (len=500)
▶ Frame 63: 110 bytes on wire (880 bits), 110 bytes captured (880 bits) on interface enp0s3, id 0 ▶ Ethernet II, Src: PcsCompu_9e:19:99 (08:00:27:9e:19:99), Dst: PcsCompu_fc:5b:46 (08:00:27:fc:5b:46) ▶ Internet Protocol Version 4, Src: 192.168.1.51, Dst: 192.168.1.52 ▶ Transmission Control Protocol, Src Port: 22, Dst Port: 43986, Seq: 2358, Ack: 1990, Len: 44 ▶ SSH Protocol ▶ SSH Version 2 (encryption:chacha20-poly1305@openssh.com mac:<implicit> compression:none) [Direction: server-to-client]						

```

0000 08 00 27 fc 5b 46 08 00 27 9e 19 99 08 00 45 00  ..'[F..'. ....E.
0010 00 60 cd 51 40 00 40 06 e9 8e c0 a8 01 33 c0 a8  ..Q@.@. ....3..
0020 01 34 00 16 ab d2 23 c8 00 dd 2c 20 7d 77 80 18  -4....#... , }w.
0030 01 f5 ca 6d 00 01 01 08 0a 3e fb 54 3d bf 2c    ...m....>..T=,
0040 04 ce e0 46 77 2f 9f 44 67 93 a0 30 3f d7 61 ec  ...Fw/.Dg..0?.a.
0050 7d 40 93 1f 30 9d 64 29 83 96 b5 23 ec 0c 33 7a  }@..0.d) ...#.3z
0060 3f 39 ba 2c 2a c1 2f 9f e9 c0 c2 24 b7 bf      ?9.,*/./...$.

```

Y LOS PAQUETES QUE DEFINEN LA AUTENTICACIÓN SON:

((ip.src==192.168.1.52)&&(ip.dst==192.168.1.51))(((ip.src==192.168.1.51)&&(ip.dst==192.168.1.52)))						
No.	Time	Source	Destination	Protocol	Length	Info
317	28.087483892	192.168.1.51	192.168.1.52	TCP	66	22 → 58988 [ACK] Seq=1098 Ack=1682 Win=64128 Len=0 TSval=1383...
318	28.186882988	192.168.1.51	192.168.1.52	SSHv2	574	Server: Diffie-Hellman Key Exchange Reply, New Keys, Encrypte...
319	28.186914421	192.168.1.52	192.168.1.51	TCP	66	58988 → 22 [ACK] Seq=1682 Ack=1686 Win=64128 Len=0 TSval=2146...
320	28.135977569	192.168.1.52	192.168.1.51	SSHv2	82	Client: New Keys
321	28.136519769	192.168.1.51	192.168.1.52	TCP	66	22 → 58988 [ACK] Seq=1686 Ack=1618 Win=64128 Len=0 TSval=1383...
322	28.152717556	192.168.1.52	192.168.1.51	SSHv2	110	Client: Encrypted packet (len=44)
323	28.153333440	192.168.1.51	192.168.1.52	TCP	66	22 → 58988 [ACK] Seq=1686 Ack=1682 Win=64128 Len=0 TSval=1383...
324	28.153353298	192.168.1.51	192.168.1.52	SSHv2	110	Server: Encrypted packet (len=44)
325	28.153507782	192.168.1.52	192.168.1.51	SSHv2	134	Client: Encrypted packet (len=68)
326	28.160514803	192.168.1.51	192.168.1.52	SSHv2	118	Server: Encrypted packet (len=52)

La autenticación del usuario se pide nada más terminar de establecer el cifrado
simétrico entre ordenadores

10. Explica en pocas líneas qué es el servicio SSH y para qué sirve, qué puerto utiliza, cómo es su autenticación y cómo viajan los datos que se intercambian entre el cliente y el servidor. ¿Hay un servicio análogo para el servicio ftp basado en SSH? (1 punto)

El servicio SSH es un protocolo basado en TCP cuyo uso es acceder remotamente a otras terminales. Al estar basado en TCP, hace uso de un three-way handshake con el fin de establecer la conexión. Utiliza el puerto 22.

El cliente y el servidor se comunican encriptando los paquetes a través de 3 tipos de encriptado distintos.

Primero hashing, que se basa en largas secuencias que no pretenden ser descifradas, sino comparadas para saber si ambos ordenadores utilizan el mismo número primo.

Luego con un cifrado asimétrico basado en una semilla —número primo muy grande— y en algunos algoritmos de cifrado predeterminados para cada sistema operativo, el que sea común a ambas máquinas.

Por último con una clave simétrica que se crea con la clave pública compartida, la clave privada y la semilla.

```

▶ Frame 14: 1578 bytes on wire (12624 bits), 1578 bytes captured (12624 bits) on interface enp0s3, id 0
▶ Ethernet II, Src: PcsCompu_fc:5b:46 (08:00:27:fc:5b:46), Dst: PcsCompu_9e:19:99 (08:00:27:9e:19:99)
▶ Internet Protocol Version 4, Src: 192.168.1.52, Dst: 192.168.1.51
▶ Transmission Control Protocol, Src Port: 43986, Dst Port: 22, Seq: 42, Ack: 42, Len: 1512
▼ SSH Protocol
  ▶ SSH Version 2 (encryption:chacha20-poly1305@openssh.com mac:<implicit> compression:none)
    [Direction: client-to-server]

0000 08 00 27 9e 19 99 08 00 27 fc 5b 46 08 00 45 00  ...[F...E...
0010 06 1c a0 32 40 00 40 06 10 f2 c0 a8 01 34 c0 a8  ...2@...4...
0020 01 33 ab d2 00 16 2c 20 75 db 23 c7 f7 d1 80 18  ...3...u...#...
0030 01 f6 89 c6 00 00 01 01 08 0a bf 2b cc ad 3e fb  ......+...>...
0040 1c 1c 00 00 05 e4 0a 14 92 d7 a8 22 97 60 6b 6e  ..."...kn...
0050 df c2 ad 8c c5 28 2c 7c 00 00 00 f1 63 75 72 76  ......(|...curv...
0060 65 32 35 35 31 39 2d 73 68 61 32 35 36 2c 63 75  e25519-s ha256,cu
0070 72 76 65 32 35 35 31 39 2d 73 68 61 32 35 36 40  rve25519 -sha256@
0080 6c 69 62 73 73 68 2e 6f 72 67 2c 65 63 64 68 2d  libssh.o rg,ecdh-
0090 73 68 61 32 2d 6e 69 73 74 70 32 35 36 2c 65 63  sha2-nis tp256,ec
00a0 64 68 2d 73 68 61 32 2d 6e 69 73 74 70 33 38 34  dh-sha2- nistp384
00b0 2c 65 63 64 68 2d 73 68 61 32 2d 6e 69 73 74 70  ,ecdh-sh a2-nistp
00c0 35 32 31 2c 64 69 66 66 69 65 2d 68 65 6c 6c 6d  521 diff ie-hellm

```

Análogamente al servicio ftp y basados en ssh existen los protocolos SFTP (SSH File Transfer Protocol) y SCP (Secure Copy Protocol).

INSTRUCCIONES

- Entrega: Un archivo PDF a partir de este documento Word modificado con las **respuestas escritas** y los pantallazos, comandos, etc. solicitados (las que están señaladas en rojo).
- Rellenar el cuadro inicial con los **Apellidos, Nombre y Curso** de los autores de la práctica.
- Los ejercicios **SÓLO** podrán realizarse en grupos de dos o tres alumnos como máximo. **No se permiten entregas de prácticas por grupos de más de tres alumnos.**
- Se deberán usar al menos dos equipos diferentes (cliente y servidor) basado en máquinas virtuales.
- **La fecha límite de entrega será el domingo 1 de Noviembre a las 23:59.**
- No se recogerán memorias **entregadas fuera de fecha o por otro medio distinto de los indicados** (como por ejemplo el mail). **Debe entregarse en el apartado correspondiente en el campus virtual.**