U-tad CENTRO UNIVERSITARIO DE TECNOLOGÍA Y ARTE DIGITAL

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 openbsd-inetd
        iii. Después se instala el demonio servidor ftp con la orden:
            apt-get install ftp

    Tambíen nos fijamos que en **/etc/inetd.conf** aparezca la linea

    ftp stream tcp nowait root /usr/sbin/tcpd /usr/sbit/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/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 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.

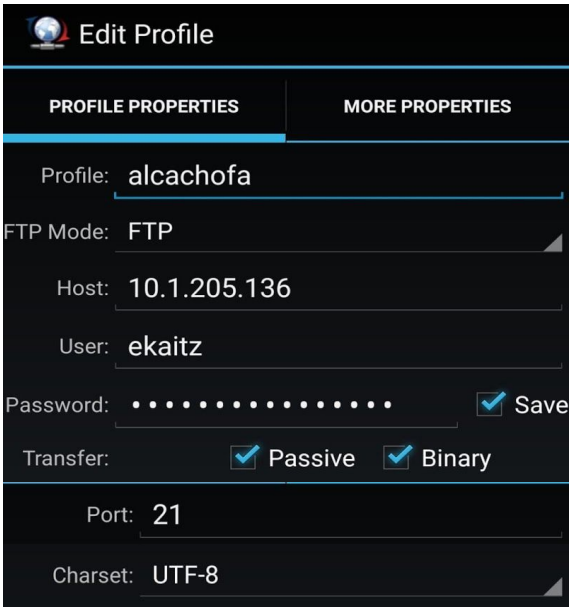
d. ¿Qué puerto utiliza este servidor?

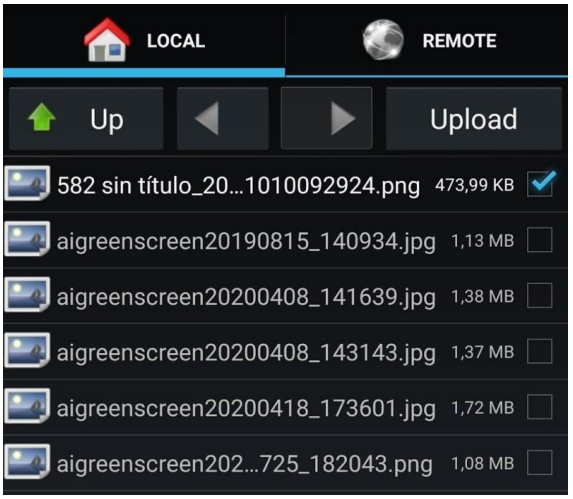
El servicio de ftp utiliza el puerto 21, el que es usado para conectarse de forma remota al servidor y autenticarse, y el puerto 20, para transferir archivos una vez se haya autenticado.

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)**

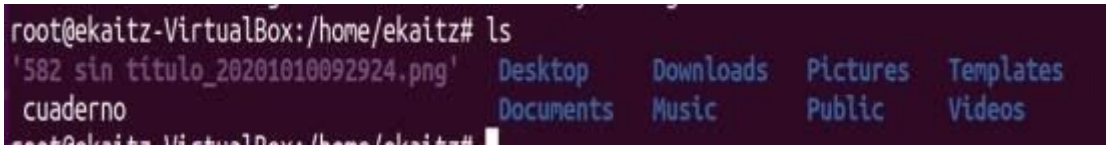
a. Mostrar el fichero de la foto en el servidor FTP

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.

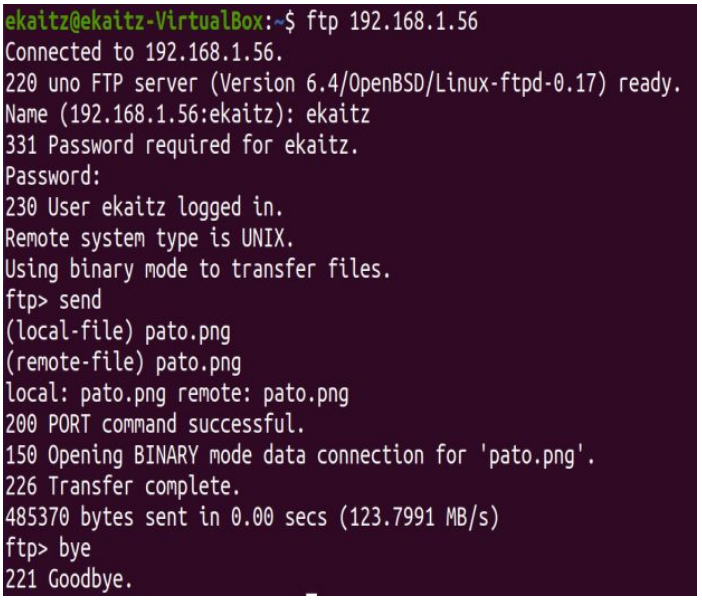


b. Otra opción sería desde cualquier shell mediante la orden

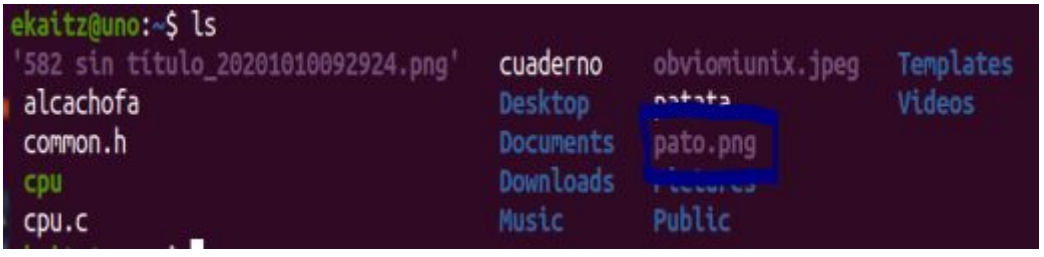
ftp <ip> y después poniendo las credenciales.

send y después las rutas (local y remoto) de la imagen.

bye para terminar la conexión.



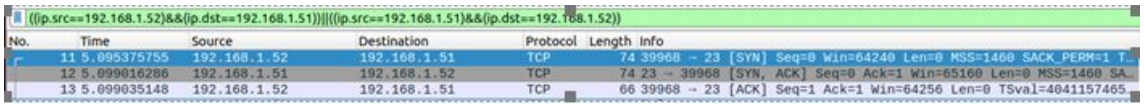
De igual modo, comprobamos que la imagen se encuentra en el servidor FTP.



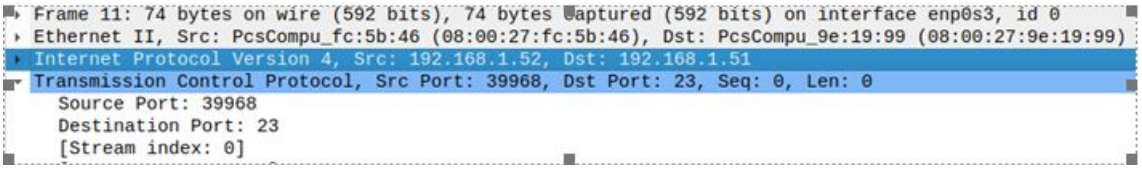
3. Monitorización con un sniffer (1 punto)

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

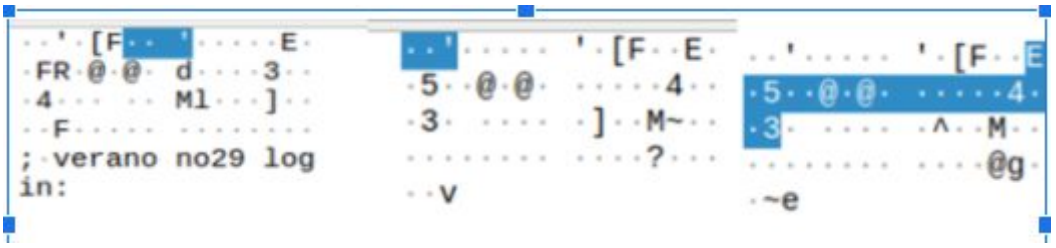
i. Three-way handshake – Tres paquetes con los que se establece la conexión cliente-servidor -

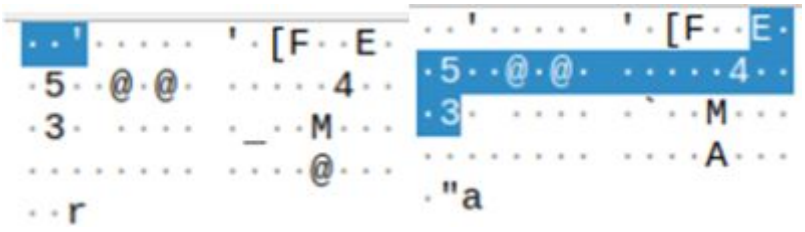


ii. Pantallazo donde aparezcan los puertos del cliente y el servidor, junto con sus direcciones IP.

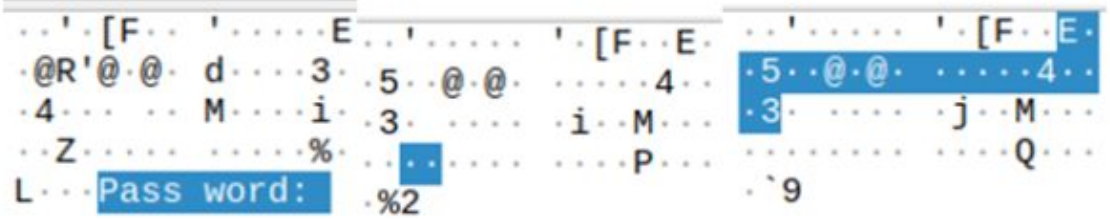


iii. USUARIO Y CONTRASEÑA - usuario veranono29

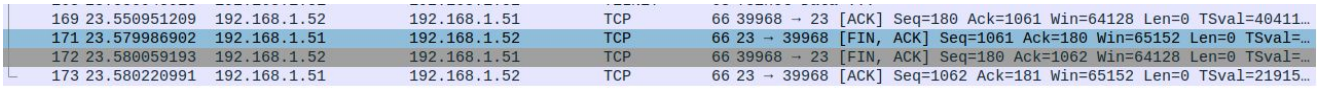




Contraseña: 29********



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)

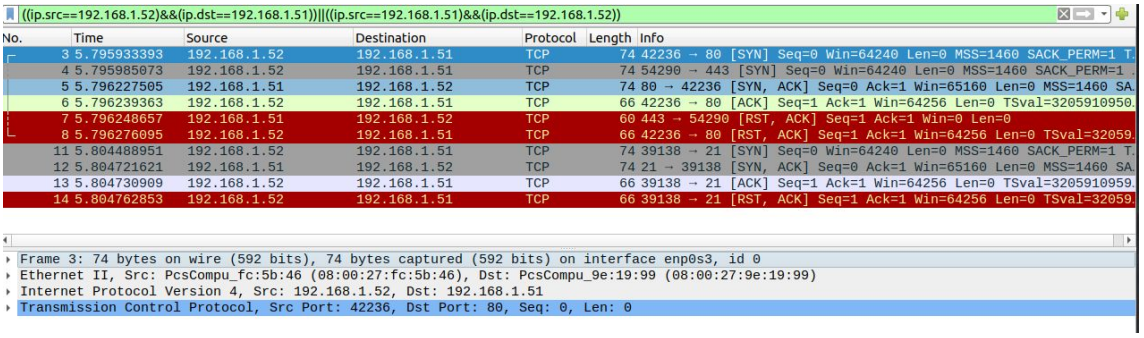
a. Hacer un escaneo **FULL SCAN** al servidor donde está el servicio ftp mediante nmap

```
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
```

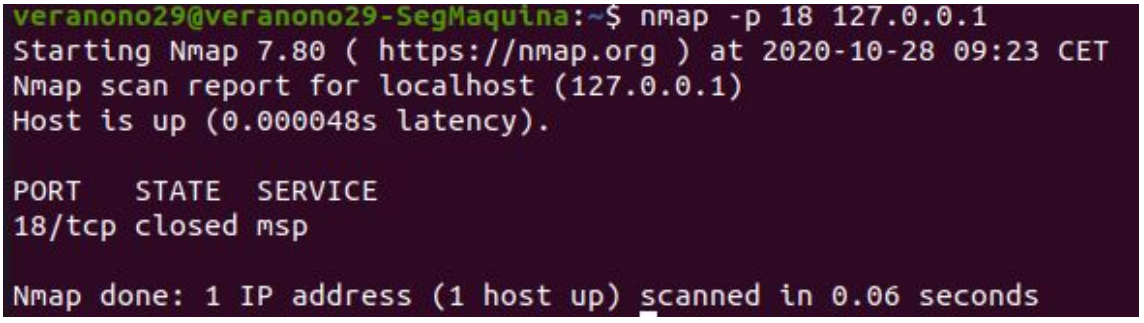
b. Identificar mediante wireshark y los filtros necesarios los paquetes mandados en ese escaneo a ese puerto en concreto. Para ello, debe aparecer:

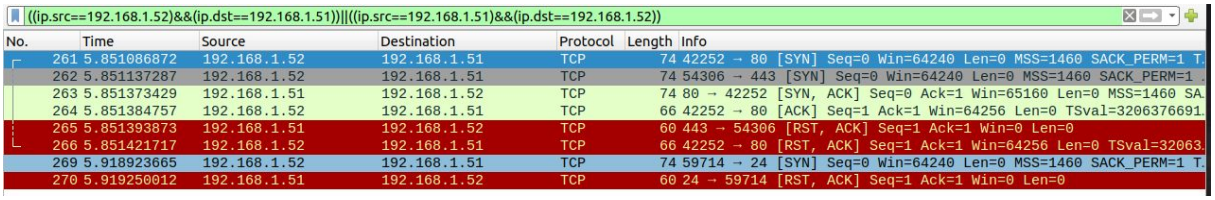
i. Un escaneo filtrado con éxito (a un puerto abierto)

```
veranono29@veranono29-SegMaquina:~$ 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
```



ii. Un escaneo filtrado a un puerto cerrado





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?

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

      ii. Este súper demonio se instala con la orden:

      apt install openbsd-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 linea

telnet stream tcp nowait telnetd /usr/sbin/tcpd /usr/sbit/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 esta encendido:

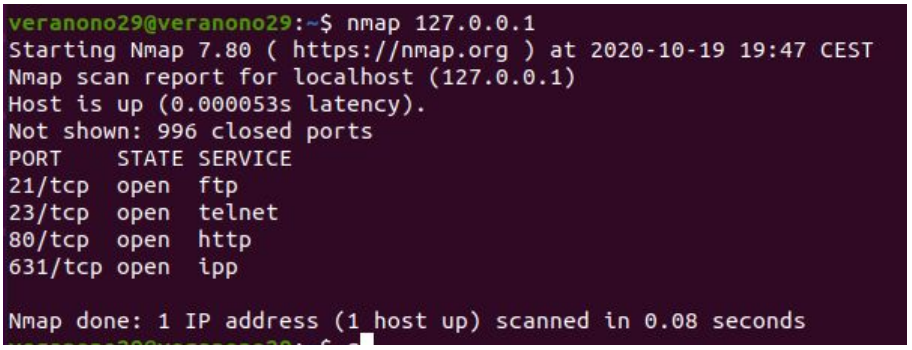
systemctl status openbsd-inetd

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 aparecerá [-] si esta apagado y [+] si esta encendido.

ii. Mediante nmap:

nmap 127.0.0.1

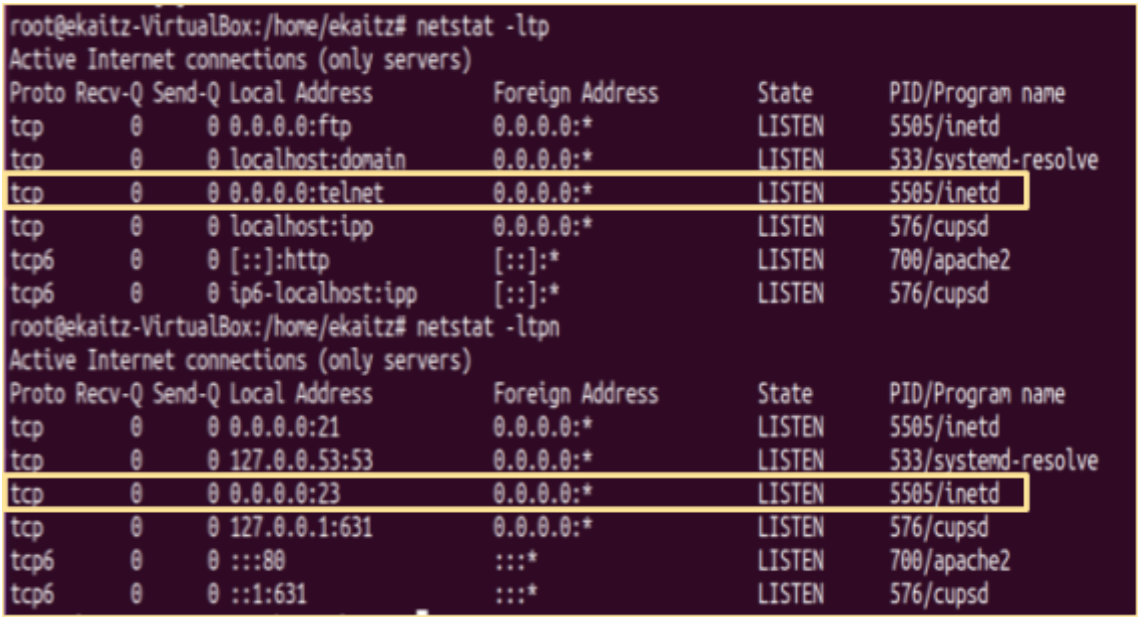


Otra forma parecida es con:

netstat -ltp

o para ver los puertos en vez de los nombres:

netstat -ltpn



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

telnet 127.0.0.1

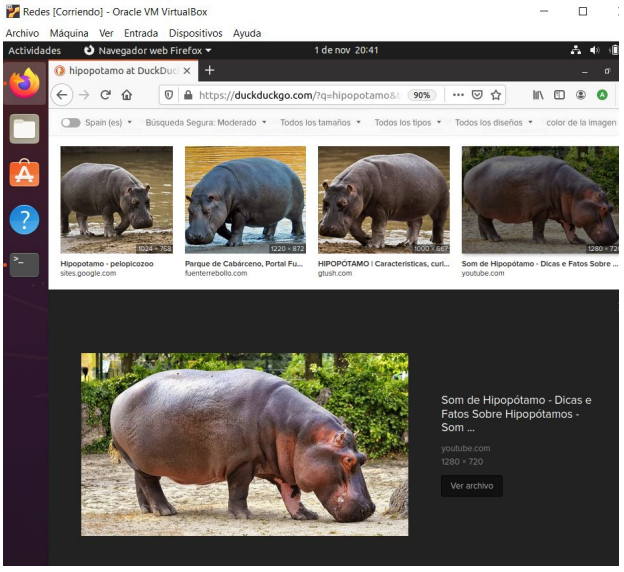
O en vez de usar una dirección de loopback (127.0.0.0/8), usando la dirección ip del servidor si se ejecutase desde otro cliente.

d. ¿Qué puerto utiliza este servidor?

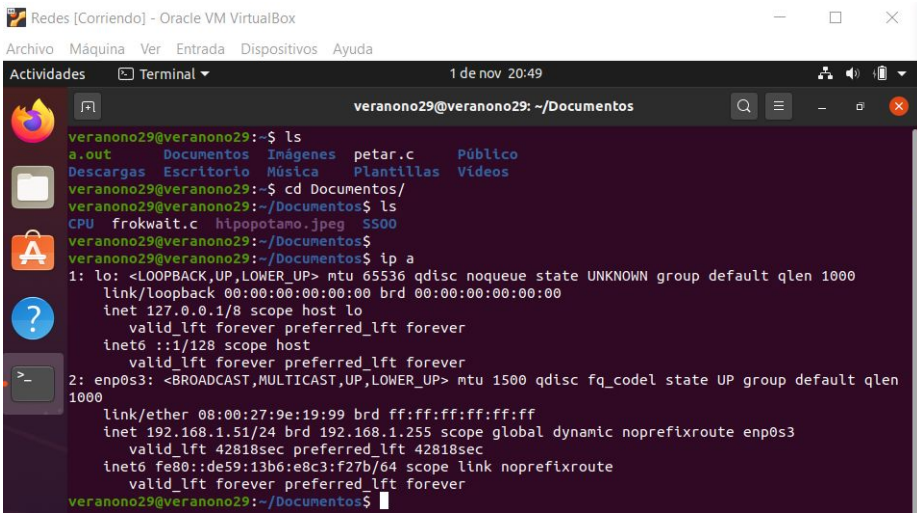
Telnet usa el puerto 23 por defecto.

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)

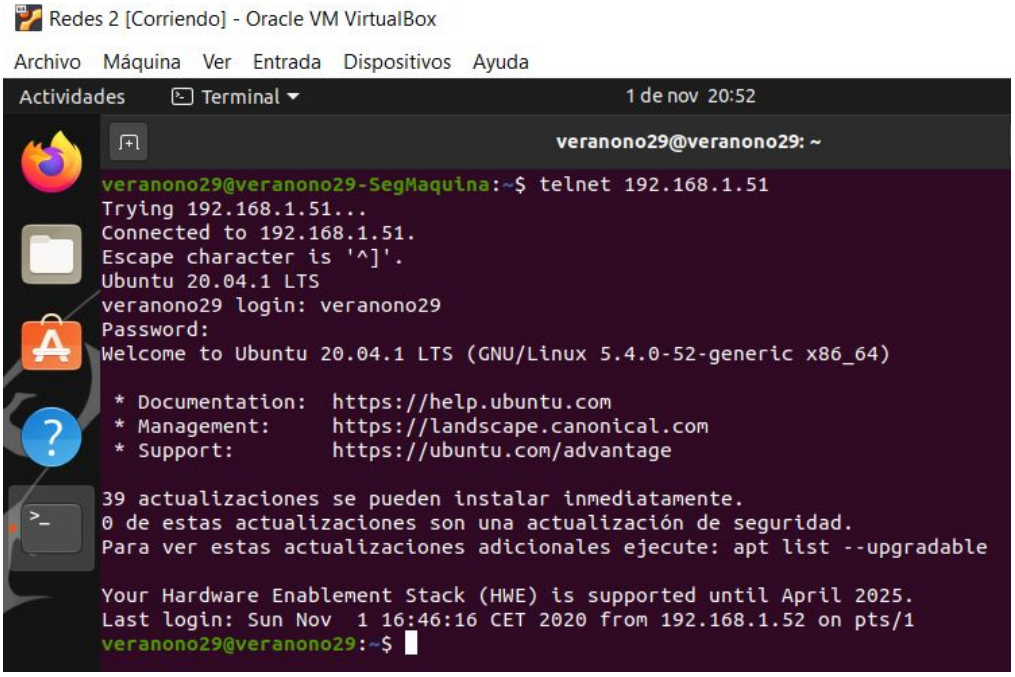
DESCARGAMOS LA FOTO



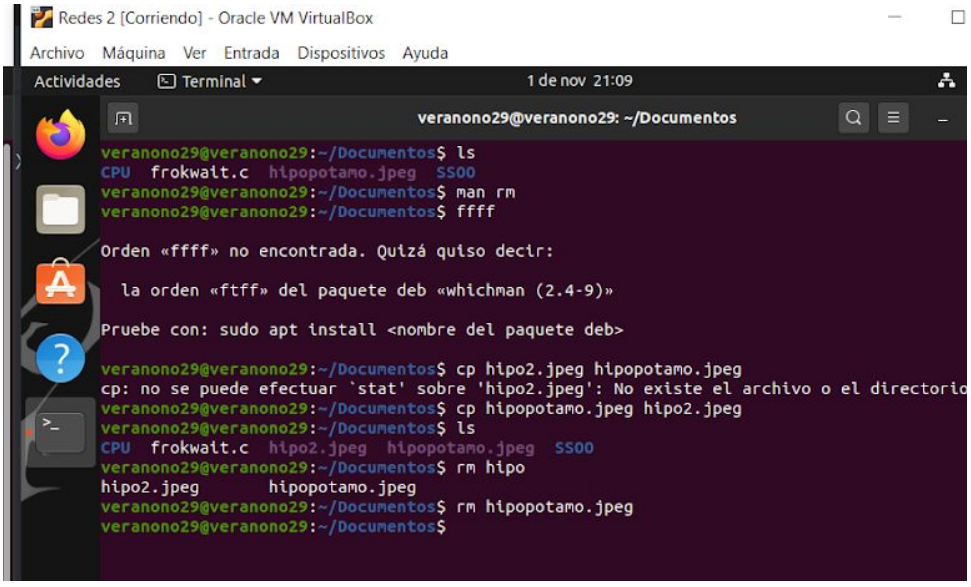
PILLAMOS LA IP



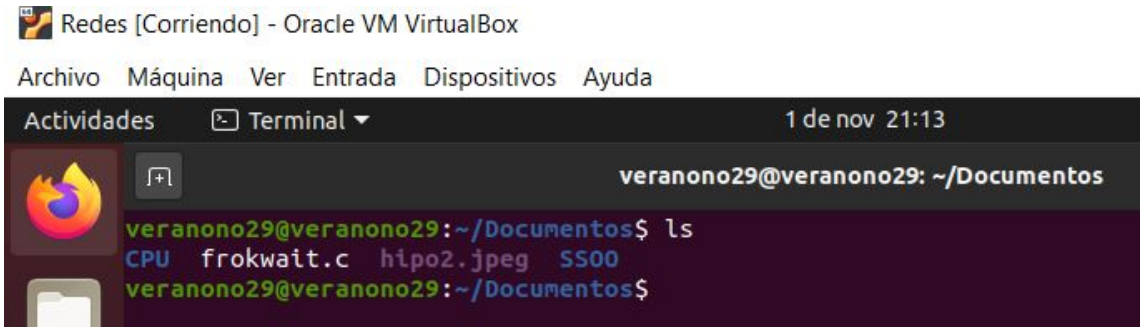
CONECTAMOS A TRAVÉS DEL TELNET



BORRAMOS LA IMAGEN DESCARGADA



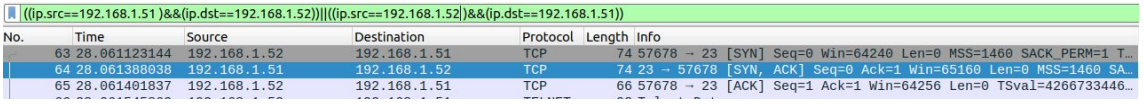
CONFIRMAMOS QUE SE HAN BORRADO



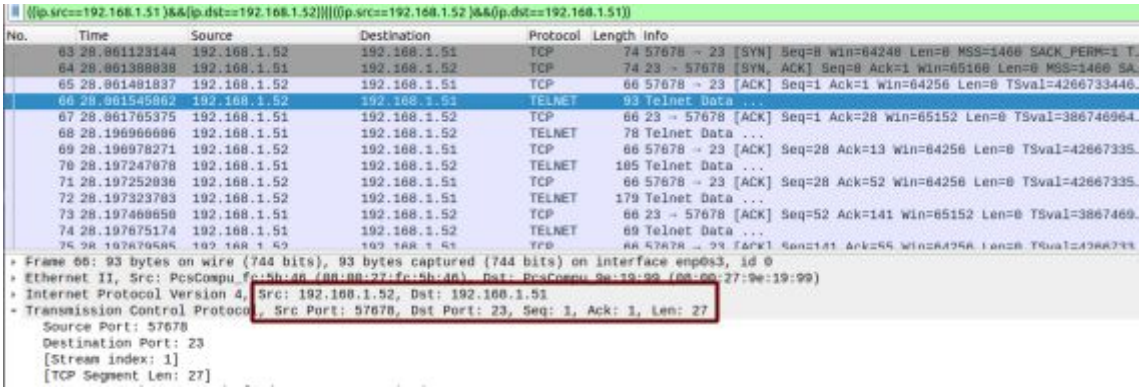
"hipo2.jpeg" es una copia de "hipopotamo.jpeg" que se realizó de antes de eliminar esta última. En esta imagen se puede observar como, en efecto, se eliminó correctamente "hipopotamo.jpeg".

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:

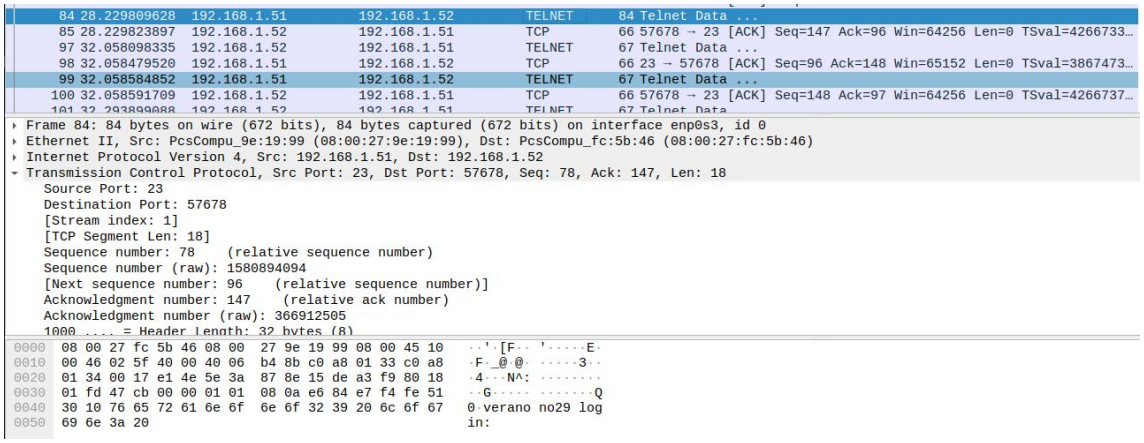
        i. Three-way handshake – Tres paquetes con los que se establece la conexión cliente-servidor -



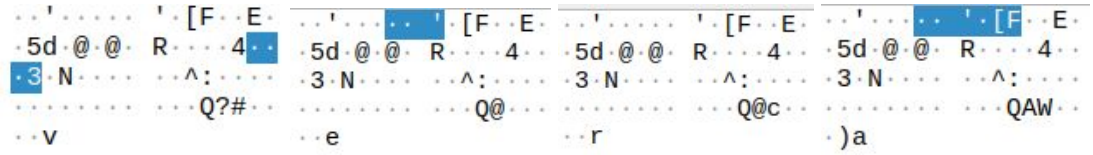
        ii. Imagen donde aparecen los puertos del cliente y el servidor, junto con sus direcciones IP.



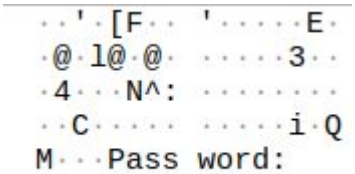
iii. USUARIO Y CONTRASEÑA -

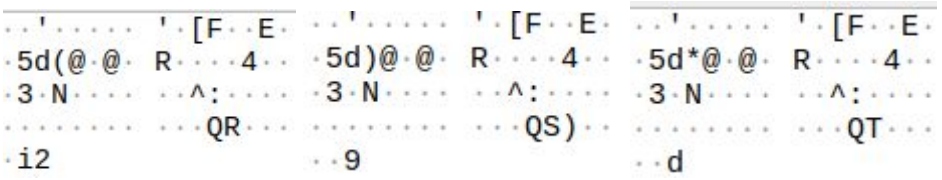


Usuario:

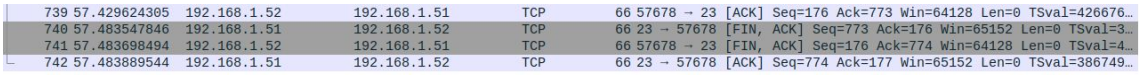


Contraseña:





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



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:

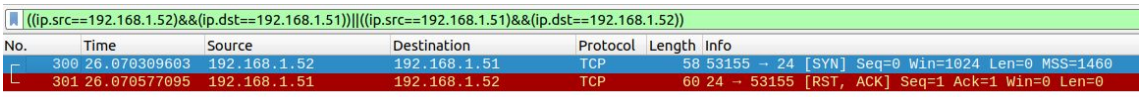
      i. Un escaneo filtrado con éxito (a un puerto abierto)

```
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=1024 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)
```



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.

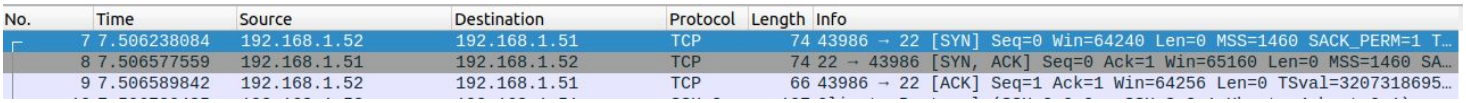
9. **Realizar el paso 3 (captura del Three-way handshake, fin de conexión y usuario y contraseña) para el servidor Open SSH, con sus correspondientes pantallazos. ¿Has podido ver la contraseña en Wireshark? ¿Qué paquetes son los que definen la autenticación? Señálalos en tu captura. (1 punto)**

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

Realizamos la conexión mediante SSH:

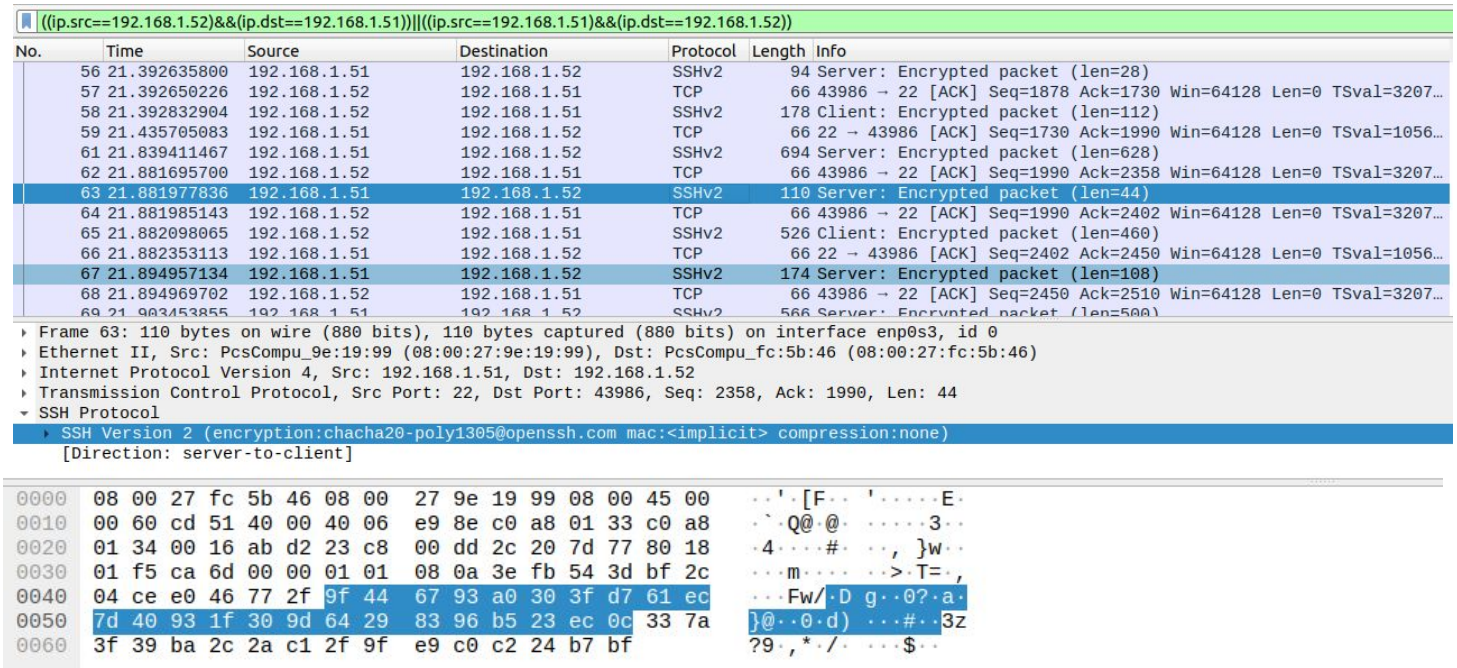
```
veranono29@veranono29-SegMaquina:~$ ssh 192.168.1.51
The authenticity of host '192.168.1.51 (192.168.1.51)' can't be established.
ECDSA key fingerprint is SHA256:Mn1tHaveumlwa5DXoiuxqjpyTb3n8GnrrbiacA2Vt5E.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently 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



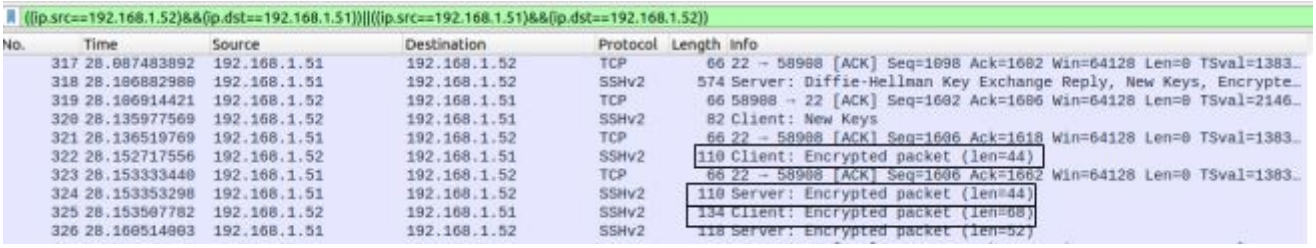
((ip.src==192.168.1.52)&&(ip.dst==192.168.1.51))||((ip.src==192.168.1.51)&&(ip.dst==192.168.1.52))

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 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/·D g··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))

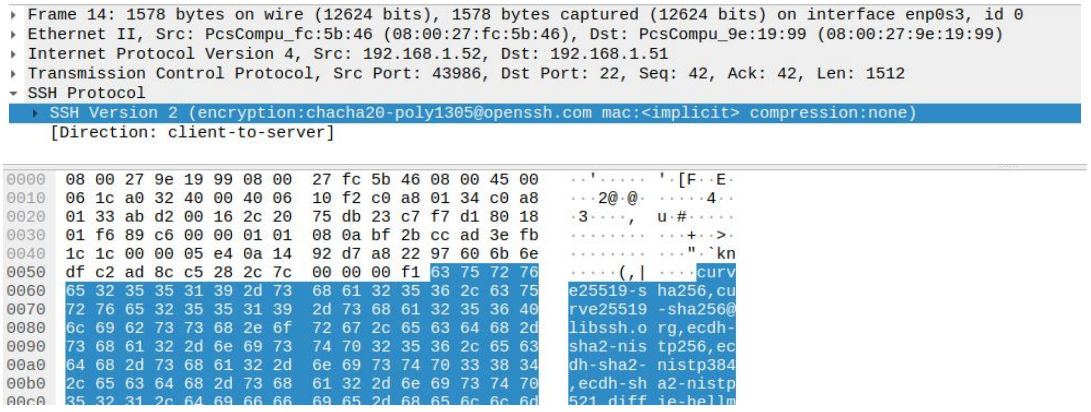
No.	Time	Source	Destination	Protocol	Length	Info
317	28.087483892	192.168.1.51	192.168.1.52	TCP	66	22 → 58908 [ACK] Seq=1098 Ack=1602 Win=64128 Len=0 TSval=1383…
318	28.106882988	192.168.1.51	192.168.1.52	SSHv2	574	Server: Diffie-Hellman Key Exchange Reply, New Keys, Encrypte…
319	28.106914421	192.168.1.52	192.168.1.51	TCP	66	58908 → 22 [ACK] Seq=1602 Ack=1606 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 → 58908 [ACK] Seq=1606 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 → 58908 [ACK] Seq=1606 Ack=1662 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	28.160514003	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.



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.**