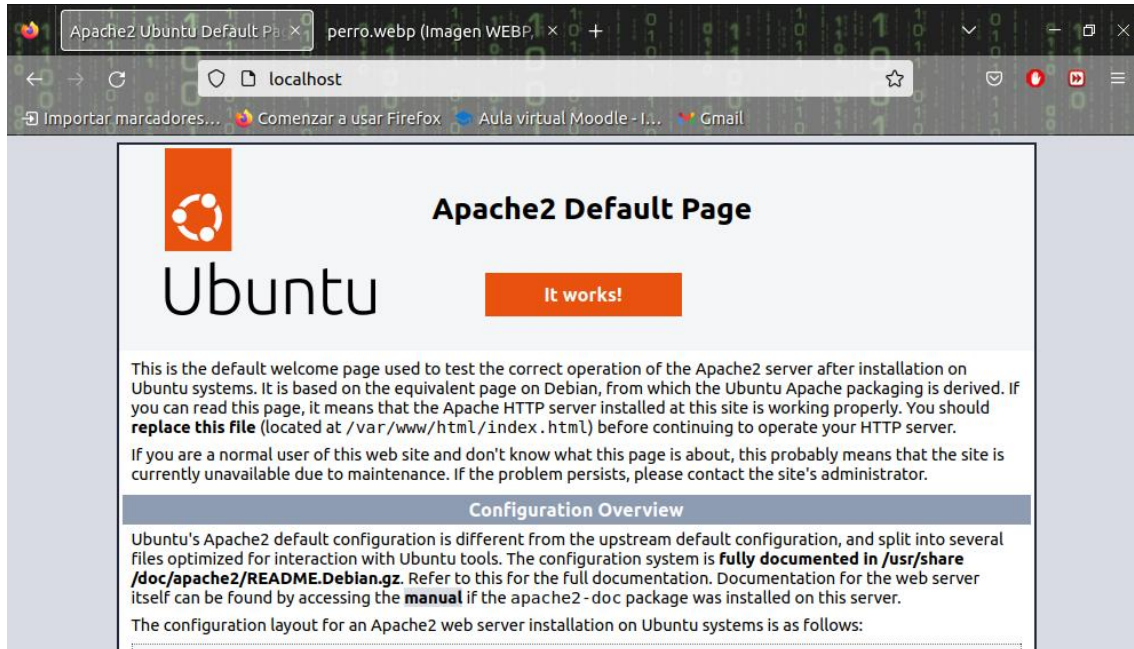
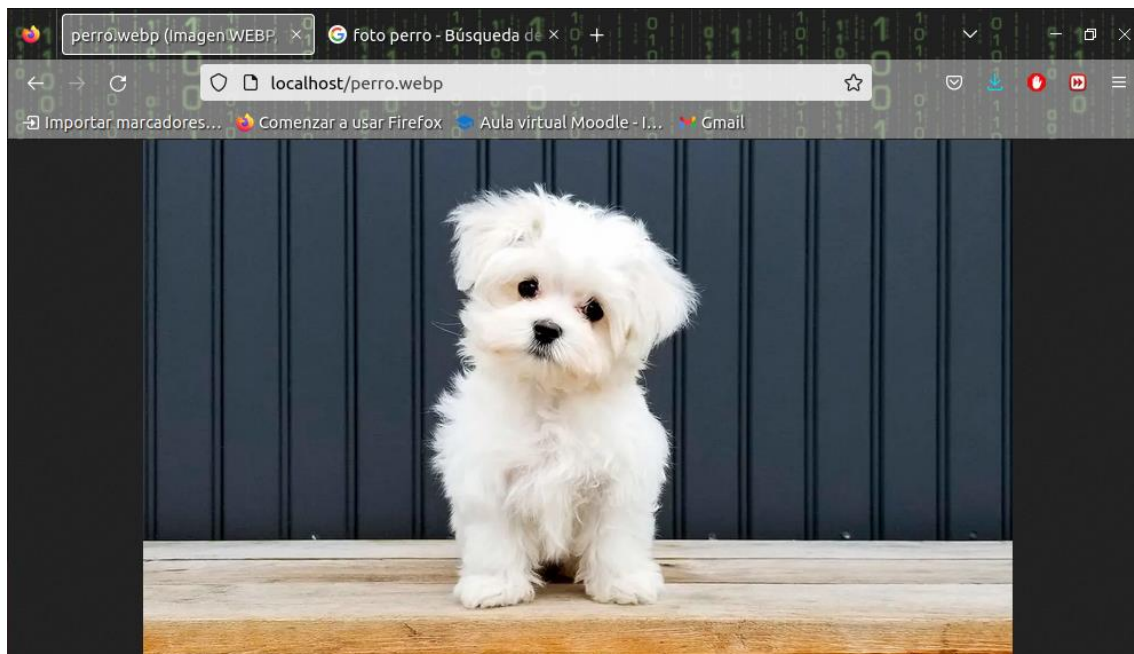


## Visualización de seq number con wireshark

- 1 – Dentro de nuestra máquina virtual con Ubuntu, escribimos en la terminal “sudo apt-get update” para comprobar nuevas actualizaciones.
- 2 – Instalamos Apache: “sudo apt-get install apache2”.
- 3 – Comprobamos que Apache funciona buscando “localhost” en el navegador de Ubuntu.

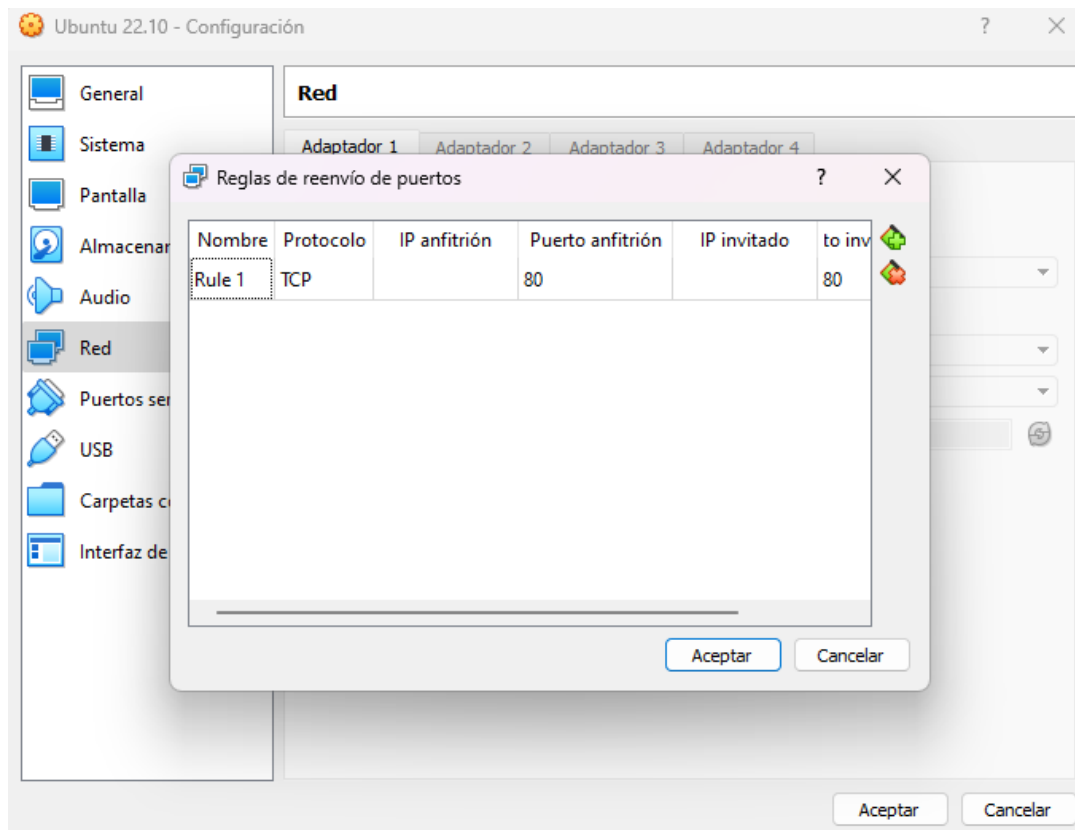


- 4 – Descargamos cualquier foto de Google y la guardamos con un nombre sencillo. En este caso “perro.webp”.
- 4 – Comprobamos que podemos visualizar la foto desde “localhost/nombredelafoto” (se debe haber guardado en la carpeta “/var/www/html/perro.webp”).



5 – Para poder visualizar la foto desde Windows, es decir, acceder al Apache de nuestra máquina virtual desde la máquina anfitriona, se deberán seguir los siguientes pasos:

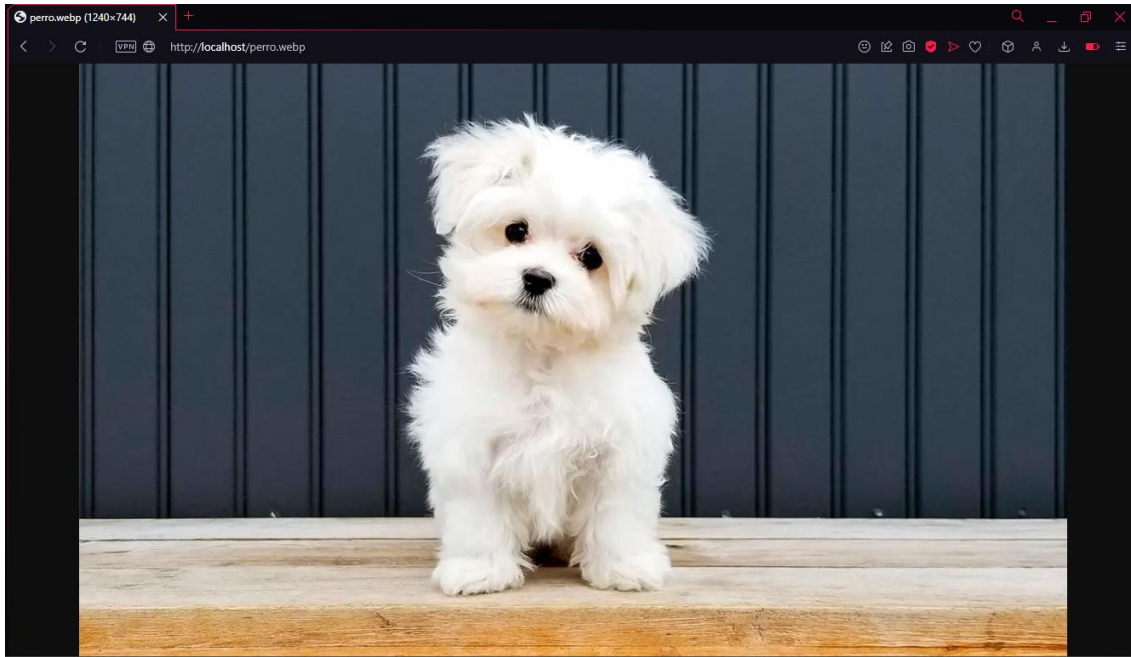
- Entrar en la configuración de la máquina virtual.
- Seleccionar el apartado de “red”.
- Ampliar apartado de “avanzadas”.
- Seleccionar “reenvío de puertos”.
- Seleccionar los puertos anfitrión y de invitado como 80 (http) con protocolo TCP:



6 – Desde el navegador de la máquina anfitriona buscamos “http://localhost” para comprobar que ha funcionado:



6 – Buscamos la imagen con “http://localhost/perro.webp”:



### Visualización de esta última búsqueda con Wireshark:

1 – Observamos el primer sequence number, que siempre es 0. El raw es 81547867. El acknowledgement number al principio también siempre es 0, así como su raw. El flag SYN está activo (1) y el ACK no (0).

Wireshark capture showing a TCP SYN packet. The packet list shows a SYN packet from 192.168.73.170 to 192.168.73.184. The packet details show the TCP header with Seq=0, Len=0, and the raw sequence number 81547867.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.73.170	192.168.73.184	TCP	66	50014 → 8013 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
2	0.033333	192.168.73.184	192.168.73.170	TCP	54	8013 → 50014 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
3	0.535072	192.168.73.170	192.168.73.184	TCP	66	[TCP Retransmission] [TCP Port numbers reused] 50014 → 8013 [SYN] Seq=0 Win=64240 Len=0
4	0.539942	192.168.73.184	192.168.73.170	TCP	54	8013 → 50014 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
5	1.086701	192.168.73.170	192.168.73.184	TCP	66	50015 → 8013 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
6	1.089330	192.168.73.184	192.168.73.170	TCP	54	8013 → 50015 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
7	1.589501	192.168.73.170	192.168.73.184	TCP	66	[TCP Retransmission] [TCP Port numbers reused] 50015 → 8013 [SYN] Seq=0 Win=64240 Len=0
8	2.004034	192.168.73.184	192.168.73.170	TCP	54	8013 → 50015 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
9	5.080696	192.168.73.170	142.250.110.188	TCP	55	49872 → 5228 [ACK] Seq=1 Ack=1 Win=510 Len=1
10	5.485421	142.250.110.188	192.168.73.170	TCP	66	5228 → 49872 [ACK] Seq=1 Ack=2 Win=265 Len=0 SLE=1 SRE=2
14	6.922054	192.168.73.170	185.26.182.111	TCP	66	50016 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
15	6.984477	185.26.182.111	192.168.73.170	TCP	58	443 → 50016 [SYN, ACK] Seq=0 Ack=1 Win=42340 Len=0 MSS=1460
16	6.984793	192.168.73.170	185.26.182.111	TCP	54	50016 → 443 [ACK] Seq=1 Ack=1 Win=64240 Len=0
17	6.985811	192.168.73.170	185.26.182.111	TLSv1.3	571	Client Hello
18	7.051591	185.26.182.111	192.168.73.170	TCP	54	443 → 50016 [ACK] Seq=1 Ack=518 Win=42383 Len=0
19	7.069488	185.26.182.111	192.168.73.170	TLSv1.3	1354	Server Hello, Change Cipher Spec, Application Data
20	7.069488	185.26.182.111	192.168.73.170	TCP	1354	443 → 50016 [ACK] Seq=1301 Ack=518 Win=42383 Len=1300 [TCP segment of a reassembled F

Frame 1: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface \Device\NPF...  
 Ethernet II, Src: IntelCor\_03:e1:ed (1c:4d:70:03:e1:ed), Dst: d6:06:d6:10:9d:7a (d6:06:d6:10:9d:7a)  
 Internet Protocol Version 4, Src: 192.168.73.170, Dst: 192.168.73.184  
 Transmission Control Protocol, Src Port: 50014, Dst Port: 8013, Seq: 0, Len: 0  
 Source Port: 50014  
 Destination Port: 8013  
 [Stream index: 0]  
 [Conversation completeness: Incomplete (37)]  
 [TCP Segment Len: 0]  
 Sequence Number: 0 (relative sequence number)  
 Sequence Number (raw): 81547867  
 [Next Sequence Number: 1 (relative sequence number)]

This shows the raw value of the sequence number (tcp.seq\_raw), 4 byte(s)

Packets: 97 - Mostrado: 87 (89.7%) - Perdido: 0 (0.0%)

Perfil: Default

2 – En el segundo segmento, el sequence number es 1 y el raw 0. El acknowledgement number es 1 y su raw 81547868 (como observamos, se ha sumado 1 por el flag SYN).

Wireshark packet capture showing a TCP SYN segment. The packet list shows a SYN segment from 192.168.73.170 to 192.168.73.184. The packet details pane shows the Transmission Control Protocol fields: Source Port: 8013, Destination Port: 50014, Seq=1, Ack=1, Len=0. The raw data pane shows the hex and ASCII representation of the packet header.

Transmission Control Protocol, Src Port: 8013, Dst Port: 50014, Seq: 1, Ack: 1, Len: 0

Source Port: 8013  
Destination Port: 50014  
[Stream index: 0]  
[Conversation completeness: Incomplete (37)]  
[TCP Segment Len: 0]  
Sequence Number: 1 (relative sequence number)  
Sequence Number (raw): 0  
[Next Sequence Number: 1 (relative sequence number)]  
Acknowledgment Number: 1 (relative ack number)  
Acknowledgment number (raw): 81547868  
0101 .... = Header Length: 20 bytes (5)

3 – En el tercer segmento, el sequence number es 0 y su raw es 81547867. El acknowledgement es 0 en ambos casos.

Wireshark packet capture showing a TCP segment. The packet list shows a segment from 192.168.73.170 to 192.168.73.184. The packet details pane shows the Transmission Control Protocol fields: Source Port: 50014, Destination Port: 8013, Seq=0, Len=0. The raw data pane shows the hex and ASCII representation of the packet header.

Transmission Control Protocol, Src Port: 50014, Dst Port: 8013, Seq: 0, Len: 0

Source Port: 50014  
Destination Port: 8013  
[Stream index: 0]  
[Conversation completeness: Incomplete (37)]  
[TCP Segment Len: 0]  
Sequence Number: 0 (relative sequence number)  
Sequence Number (raw): 81547867  
[Next Sequence Number: 1 (relative sequence number)]  
Acknowledgment Number: 0  
Acknowledgment number (raw): 0  
1000 .... = Header Length: 32 bytes (8)

La captura completa de wireshark se encuentra en la entrega del aula virtual, pero he tomado algunas notas sobre el sequence y el acknowledgement number:

#### SEQUENCE NUMBER

El sequence number es un contador utilizado para hacer un seguimiento de los bytes enviados.

En el primer paquete, este será 0 (el acknowledgement number también será 0). En el segundo paquete, el sequence number también será 0 como en el primero, pero ya el ACK será 1. En el tercero, ambos serán 1.

El flag ACK solo está desactivado en el primer segmento que manda el cliente solicitando la conexión.

El flag SYN incrementa el sequence number en 1, pero el flag ACK no.

A efectos del número de secuencia, el SYN equivale a 1 byte.

El número de ACK indica el número del primer byte que se espera recibir en el siguiente segmento.

El initial sequence number (ISN) es random.