wireshark.pdf

Practica WireShark Resuelta

3° Redes de Ordenadores

Grado en Ingeniería Informática

Escuela Politécnica Superior
Universidad Carlos III de Madrid

KEEP CALM AND ESTUDIA UN POQUITO

WUOLAH + #QuédateEnCasa

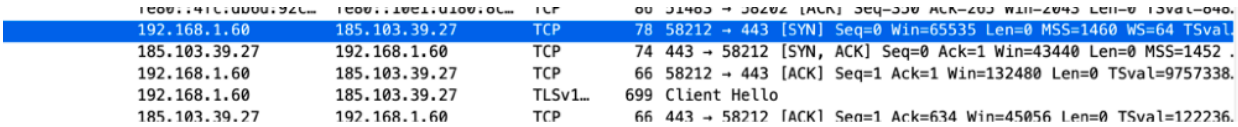
Parte 1. Capa de transporte.

Pregunta 1. Identifique los mensajes correspondientes al proceso de "three-way-handshake". ¿Qué información se intercambia en cada uno de ellos? ¿Cuáles son los números de secuencia y de ACK en cada mensaje?

Se intercambia puerto origen, puerto destino, num_seq, ACK.
SYN. SYN-ACK ACK.

Cliente PC
192.168.1.60

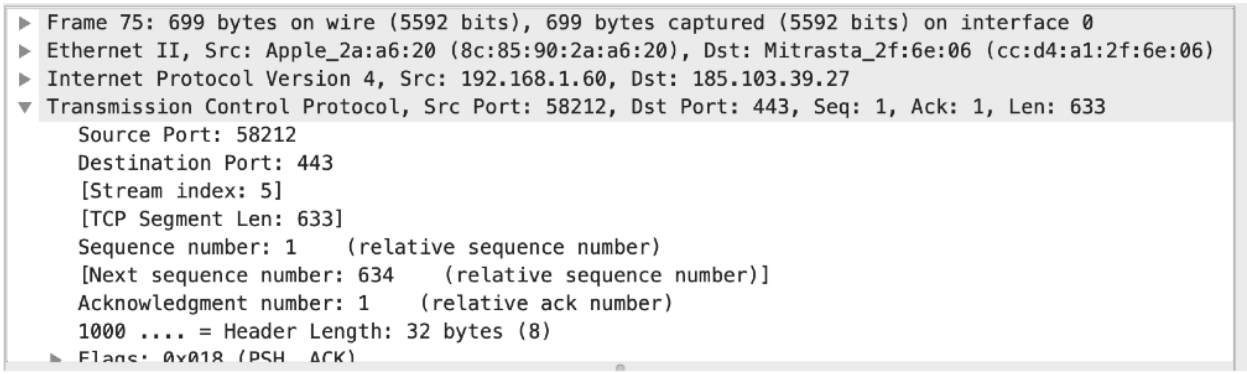
Servidor
185.103.39.27



(Recordar que TCP trabaja con flujo de byte. Que ACK es el próximo byte que se quiere recibir y que número de seq es el número de byte q se envía. Transporte fiable) + Acordarse del timeout. SYN para iniciar la comuniación.

Pregunta 2. Identifica el primer mensaje enviado una vez realizado el "three-way-handshake". ¿Cuáles son la IP y puerto de origen del mensaje? ¿Y la IP y puerto de destino? ¿Qué número de secuencia tiene el mensaje? ¿Cuál será el número de ACK esperado en la respuesta a este mensaje?

Client hello es el primer mensaje.

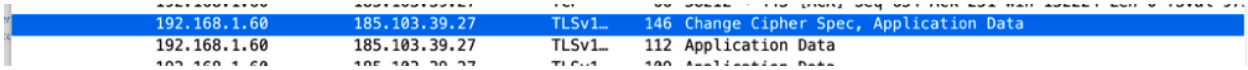


ORIGEN:
IP + puerto: 192.168.1.60 + 58212

DESTINO:
IP + puerto: 185.103.39.72 + 443

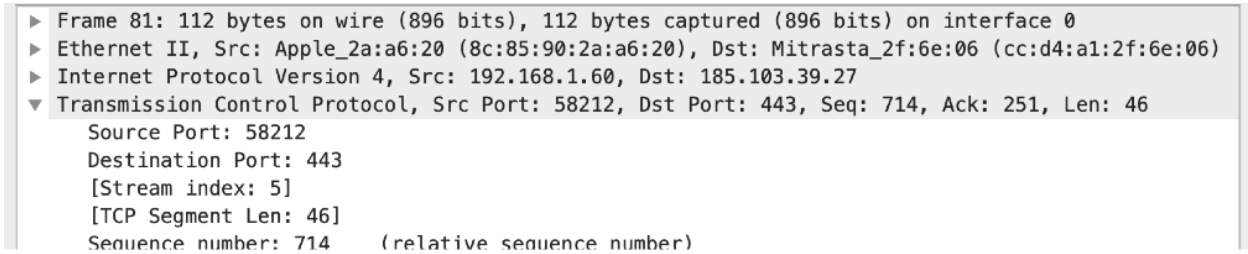
Num_seq mensaje: 1.
ACK esperado en la respuesta a este mensaje: 634.

Pregunta 3. Identifique el primer mensaje enviado en el test de bajada. ¿Cuál es el puerto de destino de la conexión? ¿Y el puerto origen? Adjunte una captura de pantalla.
Es change cipher spec, application data o bien solo application data.

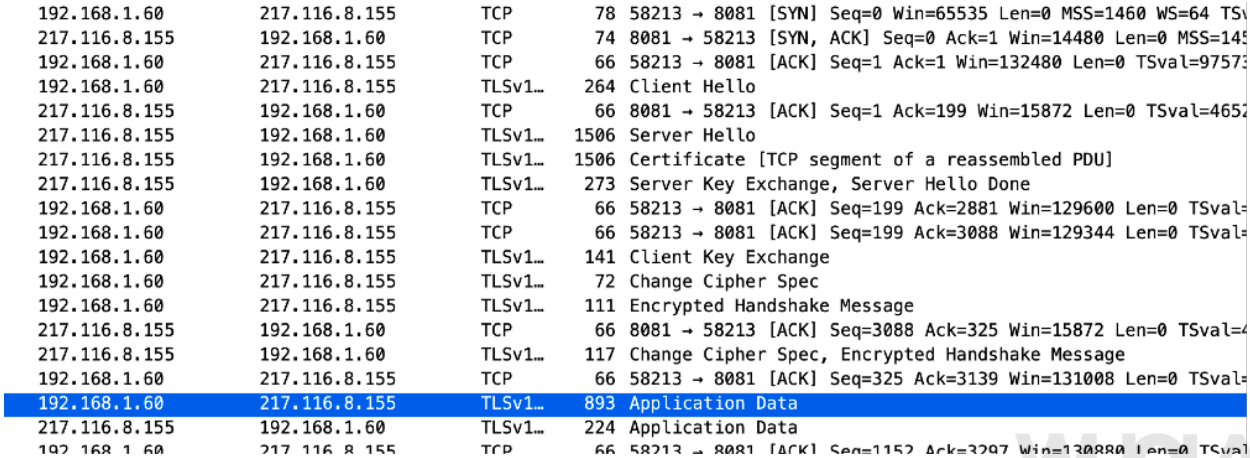


Puerto de destino: 443

Puerto de origen: 58512

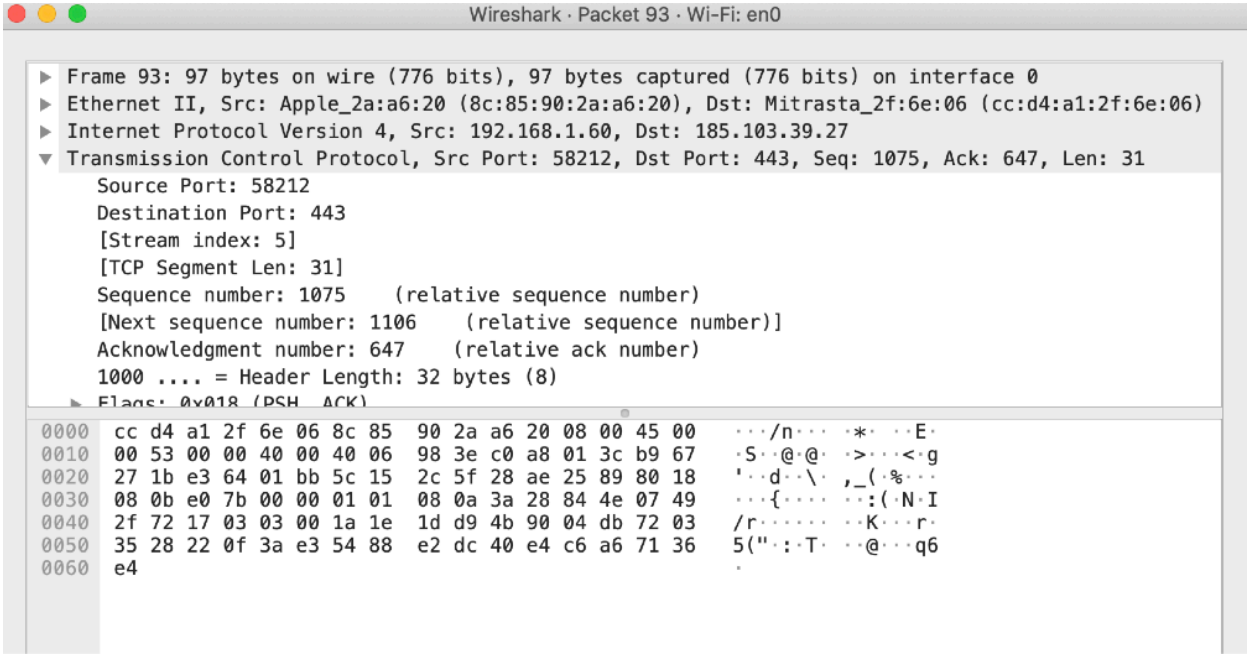


Pregunta 4. Identifique el primer mensaje enviado en el test de subida. ¿Cuál es el puerto de destino de la conexión? ¿Y el puerto origen? ¿Son los mismos puertos que en la prueba de bajada? Adjunte una captura de pantalla.



Puerto de origen: 58213
Puerto destino: 8081

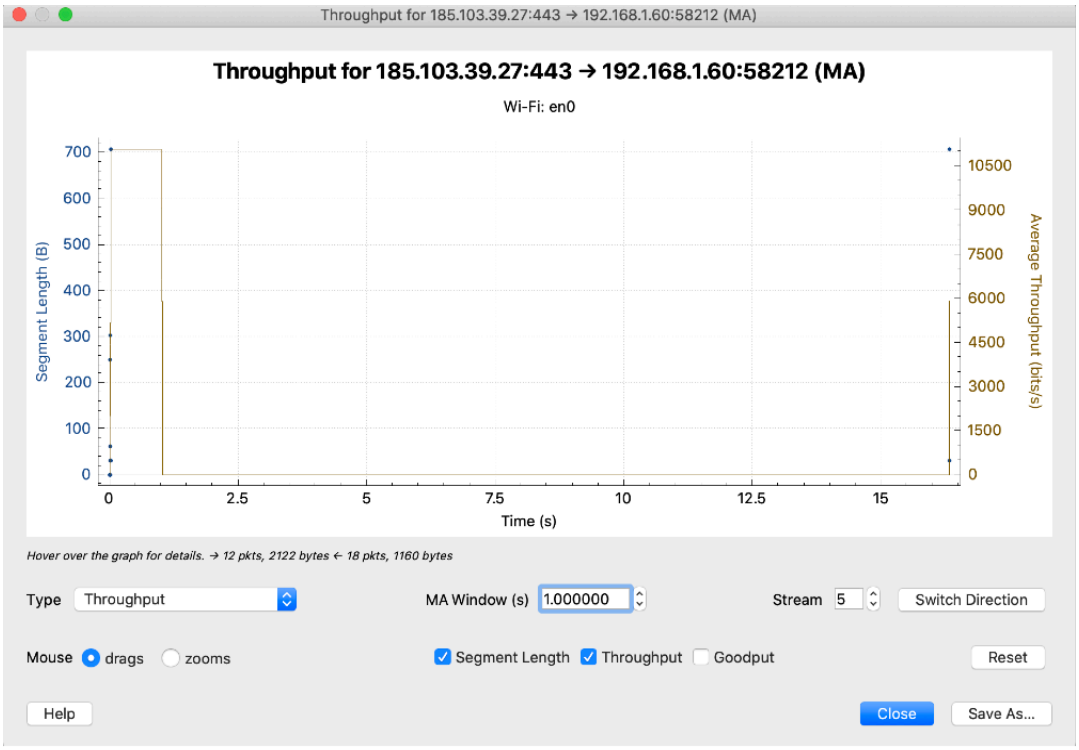
(Creo q esto es de otra conexión no de la pagina)



No son los mismos puertos que en la prueba de bajada.

En el 3 y 4. Test de bajada: de la ip 185 a 192 y test de subida de 192 a 185 (CREO).

Problema 5. **Seleccione el primer mensaje enviado en el test de bajada. Adjunte una captura de pantalla con la gráfica del "Throughput" (Statistics TCP Stream Graph...).**



CREO que el test de bajada es de 185 a 192.

P6. ¿Cuánto tiempo ha durado el test de bajada?

El test de bajada ha durado: 17.5 s.

P7. ¿Se corresponde la velocidad media de descarga con la obtenida en el test? ¿Qué velocidad es esa (en KBytes/s)? ¿Ha sido la velocidad más o menos constante a lo largo de la prueba? ¿Por qué cree que ha sido así? Justifique todas las respuestas.

222.94 Megabits / s. (La del test). —- > En KBytes / s : 27867,5.

¿Donde se ve la velocidad media de descarga?

Parte 2. Nivel de red.

Pregunta 1. ¿Cuál es el tamaño total del mensaje ICMP? ¿Cuántos datos se envían dentro del mensaje ICMP? Describa cuánto ocupa cada parte del mensaje.

¿El PING es a la ip del router?

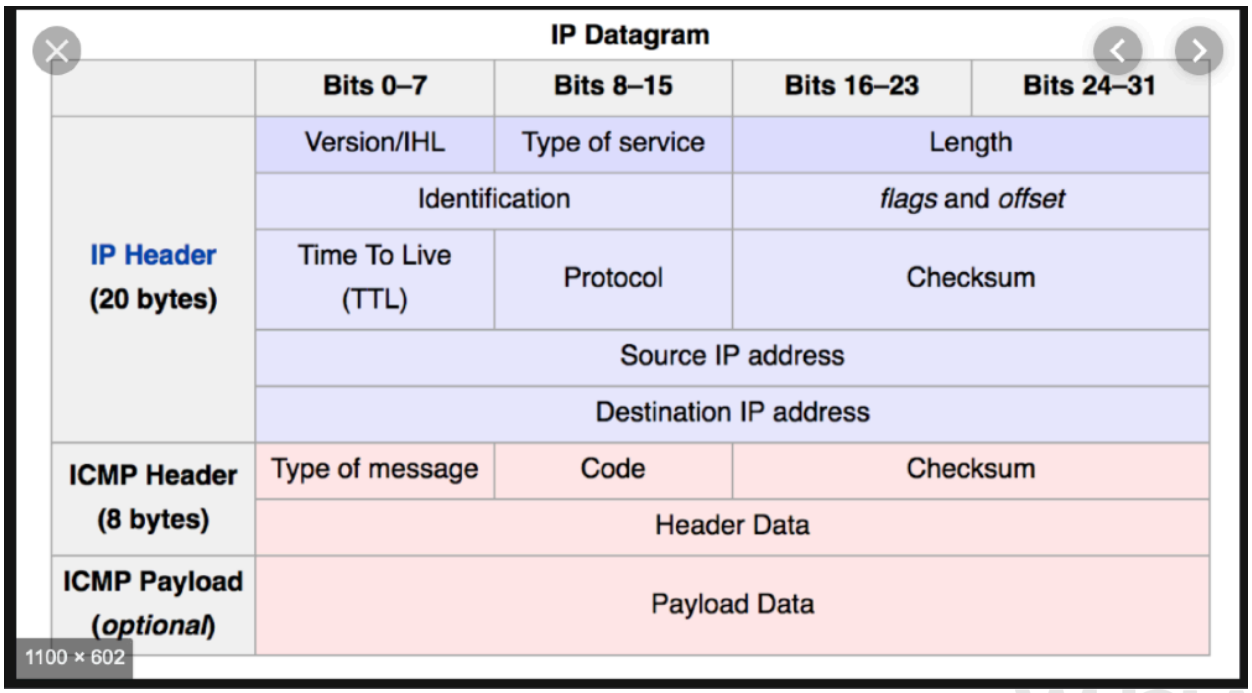
61	3.654027	192.168.1.60	192.168.1.1	ICMP	98 Echo (ping) reques
62	3.658201	192.168.1.1	192.168.1.60	ICMP	98 Echo (ping) reply
65	4.655277	192.168.1.60	192.168.1.1	ICMP	98 Echo (ping) reques
66	4.672907	192.168.1.1	192.168.1.60	ICMP	98 Echo (ping) reply
84	5.660533	192.168.1.60	192.168.1.1	ICMP	98 Echo (ping) reques
85	5.664351	192.168.1.1	192.168.1.60	ICMP	98 Echo (ping) reply
88	6.665809	192.168.1.60	192.168.1.1	ICMP	98 Echo (ping) reques
89	6.670081	192.168.1.1	192.168.1.60	ICMP	98 Echo (ping) reply
90	7.671341	192.168.1.60	192.168.1.1	ICMP	98 Echo (ping) reques
91	7.675359	192.168.1.1	192.168.1.60	ICMP	98 Echo (ping) reply
110	8.676423	192.168.1.60	192.168.1.1	ICMP	98 Echo (ping) reques
111	8.679397	192.168.1.1	192.168.1.60	ICMP	98 Echo (ping) reply

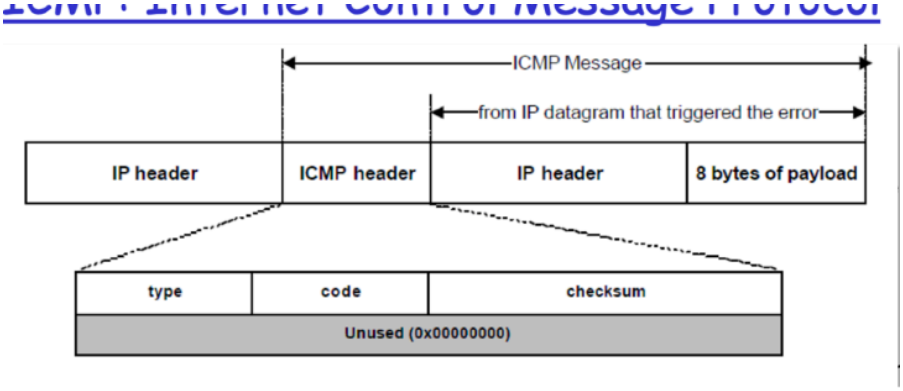
Tamaño total es 98.

Los datos que se envían dentro del mensaje ICMP:

```
▼ Data (48 bytes)
    Data: 08090a0b0c0d0e0f101112131415161718191a1b1c1d1e1f…
    [Length: 48]
```

Describe cuanto ocupa cada parte del mensaje ICMP:





Pregunta 2. ¿Cuál es el puerto origen y destino del mensaje ICMP? ¿Por qué cree que es así? Justifique las respuestas.

Puerto origen:
Puerto destino:

ICMP no tiene puerto origen ni puerto destino. Lo que tiene es IP origen e IP destino.

```
▶ Internet Protocol Version 4, Src: 192.168.1.60, Dst: 192.168.1.1
▼ Internet Control Message Protocol
    Type: 8 (Echo (ping) request)
```

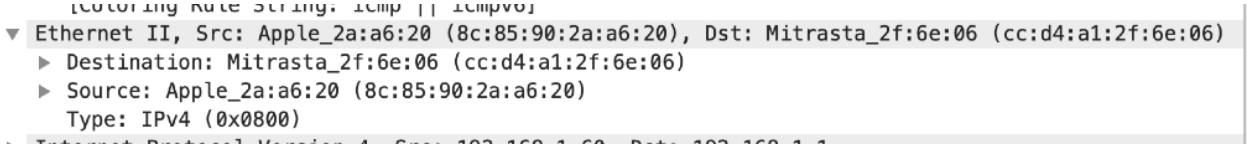
Repita los pasos anteriores, pero esta vez envíe 60000 bytes en cada mensaje. Conteste las siguientes preguntas:

Pregunta 3. Explique detalladamente cómo ha cambiado la situación ahora con respecto a la prueba anterior.

TERMINAR.ç

PARTE 3. Ethernet.

Pregunta 1. ¿Cuál es la dirección MAC de origen y de destino en el mensaje ICMP?



Pregunta 2. ¿Puede ver en el mensaje la dirección IP de la puerta de enlace? ¿Y su MAC? Justifique todas las respuestas.

En mensaje ICMP, Ethernet solo puedo ver las MAC no las IP.

Pregunta 3. ¿Qué significan los flags del mensaje ICMP en la capa Ethernet?

DUUUUDAAAAAAAA. No sé que flags son.

Pregunta 4. **¿Cuál es la dirección MAC de origen y de destino en el mensaje ICMP?**

MAC origen:

MAC destino:

```
▼ Ethernet II, Src: Apple_2a:a6:20 (8c:85:90:2a:a6:20), Dst: Mitrasta_2f:6e:06 (cc:d4:a1:2f:6e:06)
  ▶ Destination: Mitrasta_2f:6e:06 (cc:d4:a1:2f:6e:06)
  ▶ Source: Apple_2a:a6:20 (8c:85:90:2a:a6:20)
    Type: IPv4 (0x0800)
```

Pregunta 5. **¿Puede ver en el mensaje la dirección IP de Google? ¿Y su MAC? Justifique todas las respuestas.**

En ethernet no puedo ver la ip pero la mac si aunq no de google si no q la mac es la de mi router.

112	14.670576	192.168.1.60	172.217.17.14	ICMP	98	Echo (ping) reques
113	14.676282	172.217.17.14	192.168.1.60	ICMP	98	Echo (ping) reply
128	15.671768	192.168.1.60	172.217.17.14	ICMP	98	Echo (ping) reques
129	15.678493	172.217.17.14	192.168.1.60	ICMP	98	Echo (ping) reply
302	16.675225	192.168.1.60	172.217.17.14	ICMP	98	Echo (ping) reques
304	16.680343	172.217.17.14	192.168.1.60	ICMP	98	Echo (ping) reply
603	17.676066	192.168.1.60	172.217.17.14	ICMP	98	Echo (ping) reques

ARP

Pregunta 6. **Haciendo uso del comando arp, obtenga la tabla ARP de la máquina. Describa los distintos campos y adjunta una captura de pantalla.**

```
andres@ubuntu:~$ arp -a
_gateway (10.211.55.1) at 00:1c:42:00:00:18 [ether] on enp0s5
andres@ubuntu:~$ arp
Address                  HWtype  HWaddress           Flags Mask            Iface
prl-local-ns-server.sha  ether   00:1c:42:00:00:18   C                     enp0s5
andres@ubuntu:~$
```

Flags—>8

The flags indicate if the mac address has been learned, manually set, published (announced by another node than the requested) or is incomplete.

I think you can must check your kernel source to figure out what the flags mean or you simply try it. My system translates

- 0x0 incomplete
- 0x2 complete
- 0x6 complete and manually set

"Each complete entry in the ARP cache will be marked with the C flag. Permanent entries are marked with M and published entries have the P flag."

Mask no se q es. Mask es mascara de la misma forma q el prefijo ip.

IFace creo que es interface.

Adresss dirección a la que va.

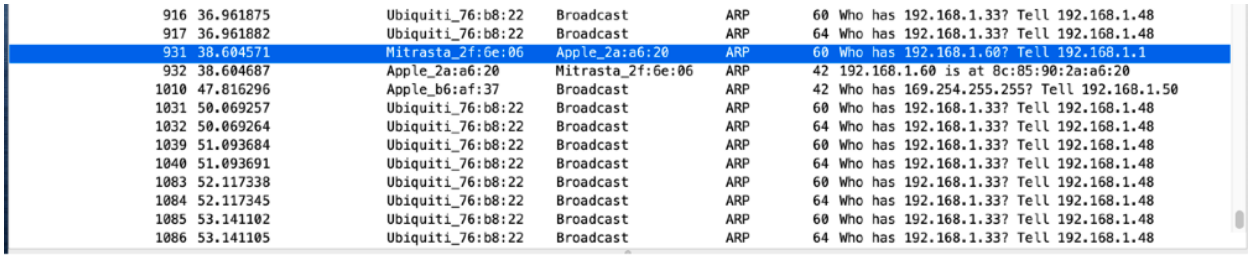
HWType es protocolo. Ether de ethernet.

Pregunta 7. Localice todos los mensajes que han permitido obtener la IP de Google (tanto DNS como ARP), y justifique cómo se ha podido llevar a cabo, paso a paso, la identificación del mismo.

No.	Time	Source	Destination	Protocol	Length	Info
147	8.915097	192.168.1.60	80.58.61.254	DNS	75	Standard query 0x670
150	8.922245	80.58.61.254	192.168.1.60	DNS	174	Standard query respo
883	33.588131	192.168.1.60	80.58.61.254	DNS	70	Standard query 0x16c
884	33.593050	80.58.61.254	192.168.1.60	DNS	86	Standard query respo
943	39.613216	192.168.1.60	80.58.61.254	DNS	96	Standard query 0xe58
946	39.619129	80.58.61.254	192.168.1.60	DNS	282	Standard query respo

Explicar como funciona DNS.

No.	Time	Source	Destination	Protocol	Length	Info
67	0.916262	Ubiquiti_76:b8:22	Broadcast	ARP	60	Who has 192.168.1.33? Tell 192.168.1.48
68	0.916266	Ubiquiti_76:b8:22	Broadcast	ARP	64	Who has 192.168.1.33? Tell 192.168.1.48
172	10.337532	Apple_b6:af:37	Broadcast	ARP	42	Who has 169.254.255.255? Tell 192.168.1.50
179	14.023525	Ubiquiti_76:b8:22	Broadcast	ARP	60	Who has 192.168.1.33? Tell 192.168.1.48
180	14.023531	Ubiquiti_76:b8:22	Broadcast	ARP	64	Who has 192.168.1.33? Tell 192.168.1.48
192	14.844479	Ubiquiti_76:b8:22	Broadcast	ARP	60	Who has 192.168.1.33? Tell 192.168.1.48
193	14.844922	Ubiquiti_76:b8:22	Broadcast	ARP	64	Who has 192.168.1.33? Tell 192.168.1.48
228	15.866663	Ubiquiti_76:b8:22	Broadcast	ARP	60	Who has 192.168.1.33? Tell 192.168.1.48
229	15.866667	Ubiquiti_76:b8:22	Broadcast	ARP	64	Who has 192.168.1.33? Tell 192.168.1.48
243	16.890942	Ubiquiti_76:b8:22	Broadcast	ARP	60	Who has 192.168.1.33? Tell 192.168.1.48
244	16.890948	Ubiquiti_76:b8:22	Broadcast	ARP	64	Who has 192.168.1.33? Tell 192.168.1.48
247	17.914951	Ubiquiti_76:b8:22	Broadcast	ARP	60	Who has 192.168.1.33? Tell 192.168.1.48
248	17.914956	Ubiquiti_76:b8:22	Broadcast	ARP	64	Who has 192.168.1.33? Tell 192.168.1.48
249	18.938975	Ubiquiti_76:b8:22	Broadcast	ARP	60	Who has 192.168.1.33? Tell 192.168.1.48
250	18.938982	Ubiquiti_76:b8:22	Broadcast	ARP	64	Who has 192.168.1.33? Tell 192.168.1.48
872	32.046383	Ubiquiti_76:b8:22	Broadcast	ARP	60	Who has 192.168.1.33? Tell 192.168.1.48
873	32.046389	Ubiquiti_76:b8:22	Broadcast	ARP	64	Who has 192.168.1.33? Tell 192.168.1.48
874	33.070995	Ubiquiti_76:b8:22	Broadcast	ARP	60	Who has 192.168.1.33? Tell 192.168.1.48
875	33.071001	Ubiquiti_76:b8:22	Broadcast	ARP	64	Who has 192.168.1.33? Tell 192.168.1.48
887	33.889858	Ubiquiti_76:b8:22	Broadcast	ARP	60	Who has 192.168.1.33? Tell 192.168.1.48
888	33.889864	Ubiquiti_76:b8:22	Broadcast	ARP	64	Who has 192.168.1.33? Tell 192.168.1.48
900	34.915444	Ubiquiti_76:b8:22	Broadcast	ARP	60	Who has 192.168.1.33? Tell 192.168.1.48
901	34.915804	Ubiquiti_76:b8:22	Broadcast	ARP	64	Who has 192.168.1.33? Tell 192.168.1.48
907	35.937663	Ubiquiti_76:b8:22	Broadcast	ARP	60	Who has 192.168.1.33? Tell 192.168.1.48



916	36.961875	Ubiquiti_76:b8:22	Broadcast	ARP	60 Who has 192.168.1.33? Tell 192.168.1.48
917	36.961882	Ubiquiti_76:b8:22	Broadcast	ARP	64 Who has 192.168.1.33? Tell 192.168.1.48
931	38.604571	Mitrasta_2f:6e:06	Apple_2a:a6:20	ARP	60 Who has 192.168.1.60? Tell 192.168.1.1
932	38.604687	Apple_2a:a6:20	Mitrasta_2f:6e:06	ARP	42 192.168.1.60 is at 8c:85:90:2a:a6:20
1010	47.816296	Apple_b6:af:37	Broadcast	ARP	42 Who has 169.254.255.255? Tell 192.168.1.50
1031	50.069257	Ubiquiti_76:b8:22	Broadcast	ARP	60 Who has 192.168.1.33? Tell 192.168.1.48
1032	50.069264	Ubiquiti_76:b8:22	Broadcast	ARP	64 Who has 192.168.1.33? Tell 192.168.1.48
1039	51.093684	Ubiquiti_76:b8:22	Broadcast	ARP	60 Who has 192.168.1.33? Tell 192.168.1.48
1040	51.093691	Ubiquiti_76:b8:22	Broadcast	ARP	64 Who has 192.168.1.33? Tell 192.168.1.48
1083	52.117338	Ubiquiti_76:b8:22	Broadcast	ARP	60 Who has 192.168.1.33? Tell 192.168.1.48
1084	52.117345	Ubiquiti_76:b8:22	Broadcast	ARP	64 Who has 192.168.1.33? Tell 192.168.1.48
1085	53.141102	Ubiquiti_76:b8:22	Broadcast	ARP	60 Who has 192.168.1.33? Tell 192.168.1.48
1086	53.141105	Ubiquiti_76:b8:22	Broadcast	ARP	64 Who has 192.168.1.33? Tell 192.168.1.48

Explicar como funciona ARP.

Pregunta 8.Explique para qué sirven los comandos del paso 2.

sudo ip neigh flush dev eth0 —-> Para eliminar de la cache ARP eth0

sudo /etc/init.d/nscd restart —-> elimina la cache DNS.

Pregunta 9. Localice todos los mensajes que han permitido obtener la IP de Google (tanto DNS como ARP). ¿Ha cambiado algo respecto al caso analizado en la P6? Justifique la respuesta.

VERRRRRRRRRRRr no me ejecutan estos comandos debería ser Linux.

Pregunta 10. Adjunte capturas de pantalla significativas en las que se muestre la nueva tabla ARP y los mensajes capturados por Wireshark. Describa el procedimiento empleado para conseguir el desvío de tráfico propuesto, indicando los comandos utilizados y justificando su uso.

Para conseguir el desvio de tráfico propuesto. Asociar dirección MAC del atacante con la dirección Ip del atacado así, de esta forma se consigue desviar .

Los comandos serían añadir arp -S 1.1.1.1 de-ad-be-ef-de-ad .

Ip neigh add lladder dev devide nud state

arp -s address hw_adress