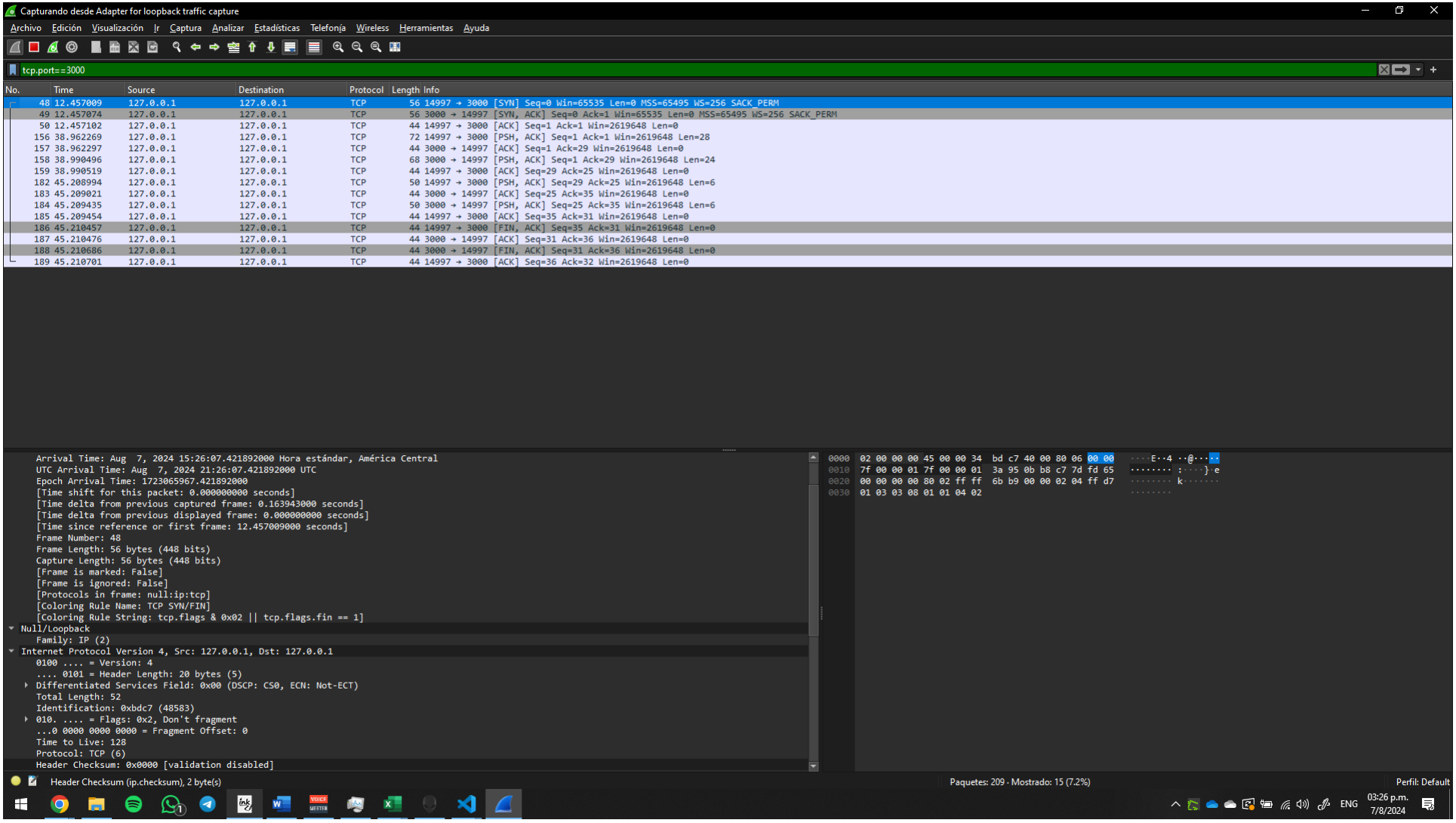
SECCIÓN TCP:

1. Captura de paquetes



2. Análisis de encabezados

No.	Source	Destination	Length	Flags	Seq	Ack	Win	Len	MSS	WS	SACK_PERM
48	14997	3000	56	SYN	0		65535	0	65495	256	SACK_PERM
49	3000	14997	56	SYN, ACK	0	1	65535	0	65495	256	SACK_PERM
50	14997	3000	44	ACK	1	1	2619648	0			
156	14997	3000	72	PSH, ACK	1	1	2619648	28			
157	3000	14997	44	ACK	1	29	2619648	0			
158	3000	14997	68	PSH, ACK	1	29	2619648	24			
159	14997	3000	44	ACK	29	25	2619648	0			
182	14997	3000	50	PSH, ACK	29	25	2619648	6			
183	3000	14997	44	ACK	25	35	2619648	0			
184	3000	14997	50	PSH, ACK	25	35	2619648	6			
185	14997	3000	44	ACK	35	31	2619648	0			
186	14997	3000	44	FIN, ACK	35	31	2619648	0			
187	3000	14997	44	ACK	31	36	2619648	0			
188	3000	14997	44	FIN, ACK	31	36	2619648	0			
189	14997	3000	44	ACK	36	32	2619648	0			

14997 -> Client
3000 -> Server

3. Preguntas de análisis

¿Cuál es el número de secuencia (Sequence Number) inicial y final de la conexión?¿Hace sentido que sean esa cantidad, por qué?

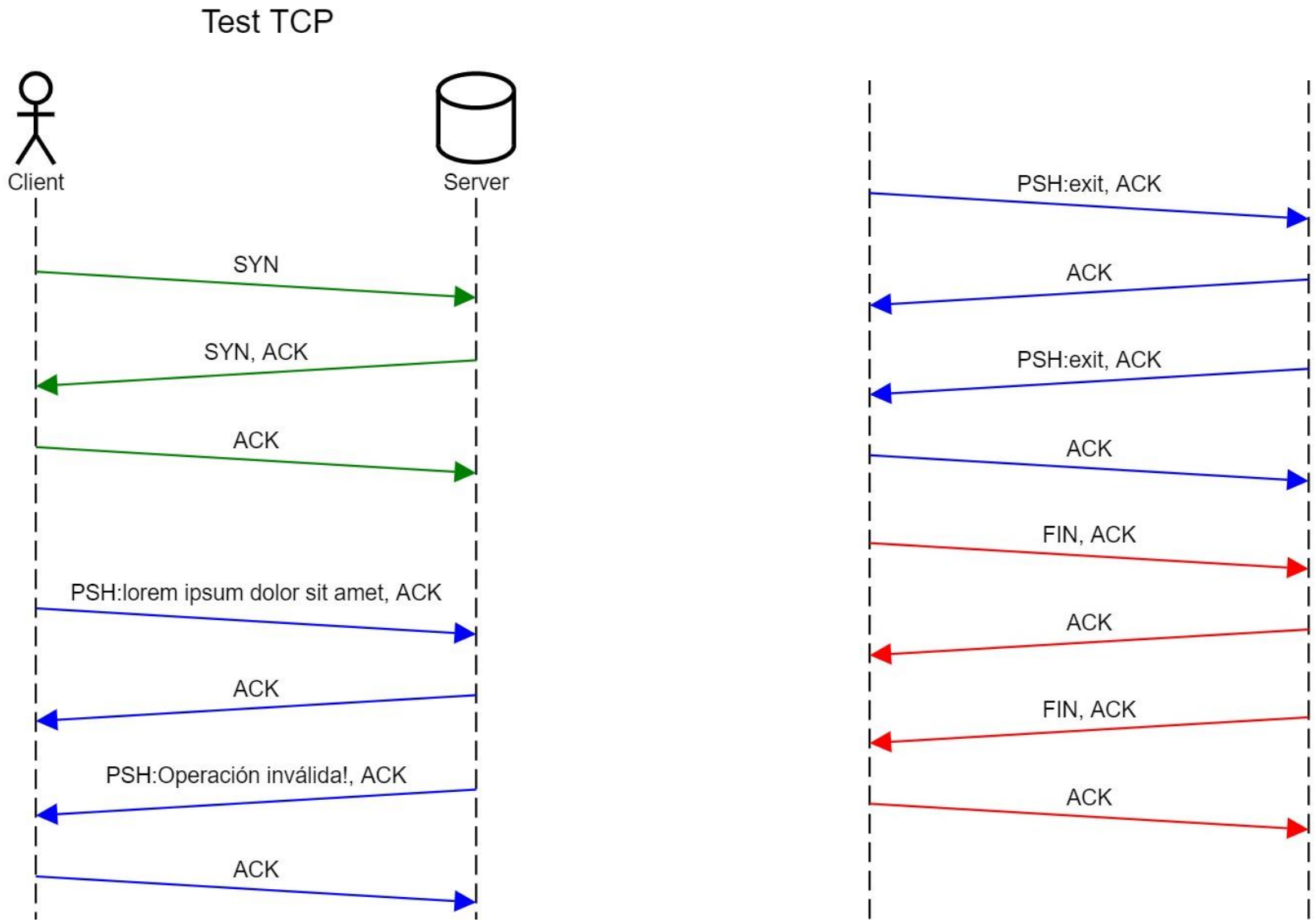
Seq inicial es 0. Seq final es 36 (31 para Server).

Sí tiene sentido, según lo que investigué, el Seq es el número de información que se ha mandado acumulado. Esta comienza en 0, vemos que con el primer mensaje aumenta a 1, cuando el cliente envía 28 el siguiente seq de cliente es 29 y el ack de server también. Y cuando el server manda lenght 24, el ack del cliente es 25 y el siguiente seq del server es 25. Y así continúa hasta cerrar la conexión.

¿Se enviaron 4 mensajes, por lo cual solo deben de existir 4 paquetes con DATA? ¿Es eso correcto o existen más paquetes?

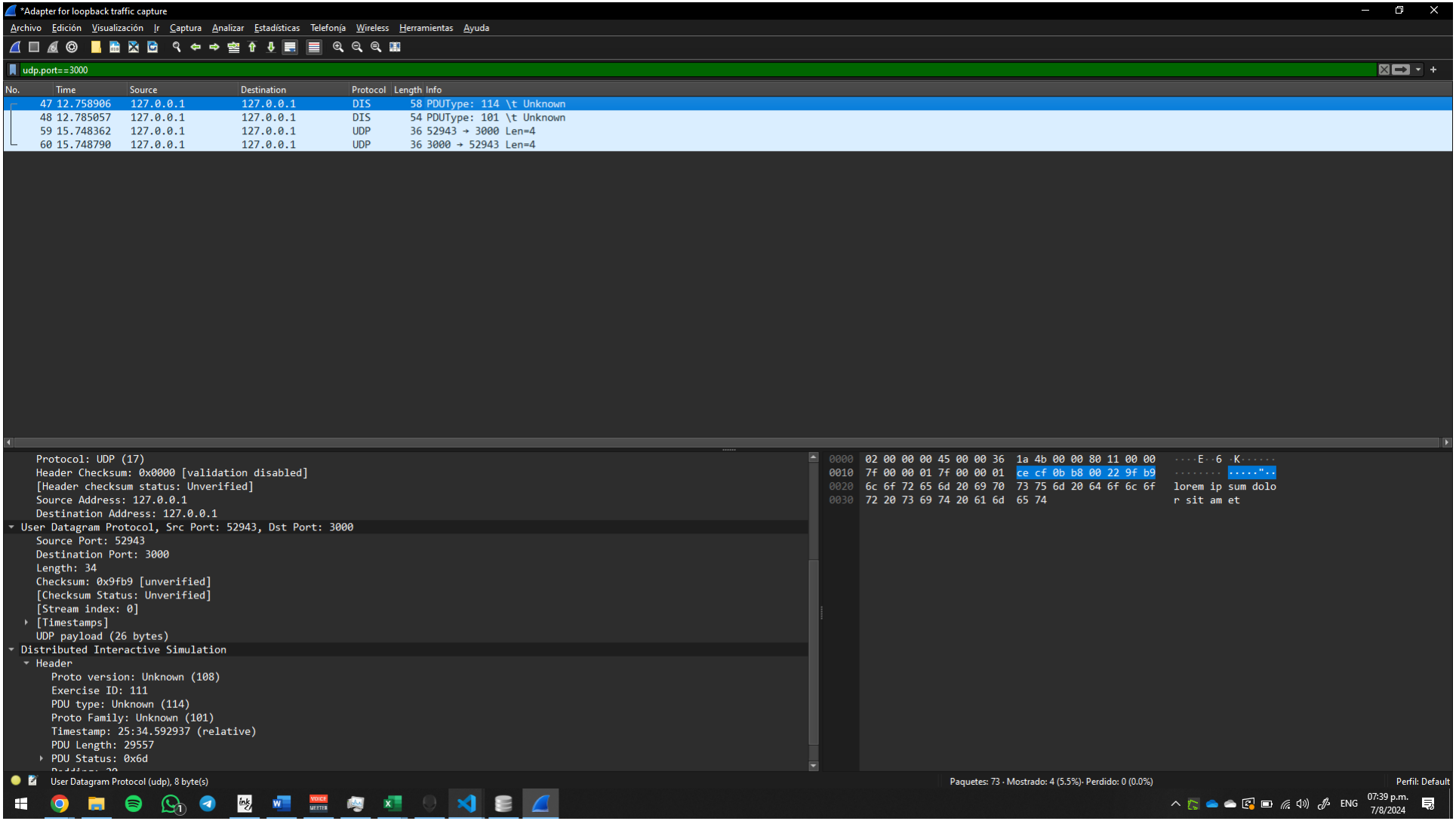
Sí. Los únicos 4 paquetes con DATA deberían ser los mensajes de: Lorem Ipsum...; Operación inválida! ; exit ; exit ;

4. Diagrama Secuencial



Sección UDP:

1. Captura de paquetes



2. Análisis de encabezados

No.	Source	Destination	Length	Len	Checksum	Data
47	52943	3000	58	34	0x9fb9	lorem ipsum dolor sit amet
48	3000	52943	54	30	0x816b	Operación no válida!
59	52943	3000	36	4	0x585f	exit
60	3000	52943	36	4	0x585f	Exit

14997 -> Client
3000 -> Server

3. Preguntas de análisis

¿Se generó algún ICMP durante la transmisión? ¿Podría generarse alguno?

Se colocó el filtro “udp.port==3000 || icmp” para verificar si había algún ICMP, pero tampoco apareció ningún paquete nuevo.

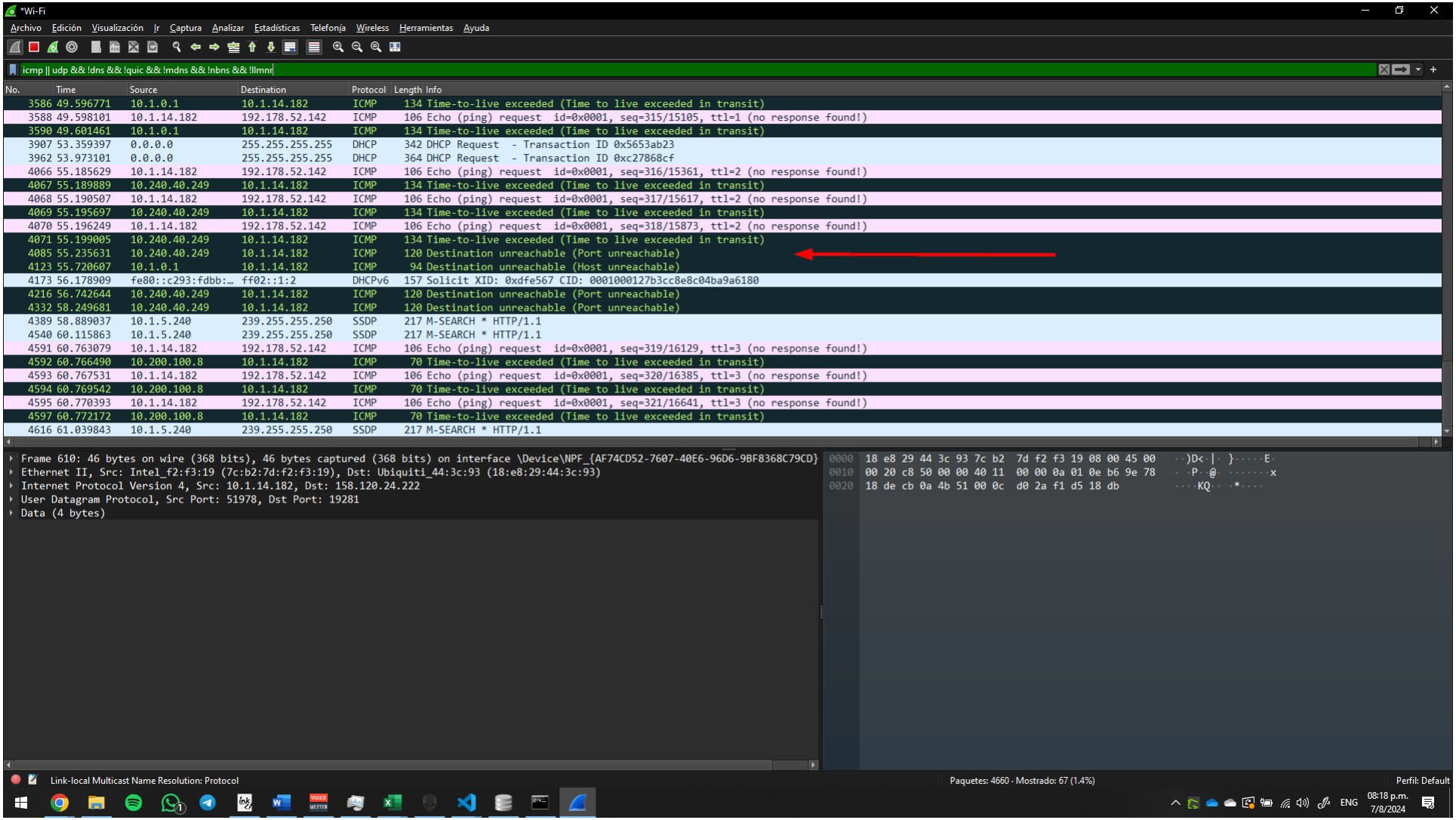
Según la captura, ¿los paquetes tenían el mismo length? Si no, ¿cuál piensa que sea el motivo?

No tenían el mismo length ya que el mensaje (data) no tiene el mismo length. Pero vemos que mensajes con el mismo tamaño (exit) ambos tienen el mismo length ya que tienen el mismo contenido.

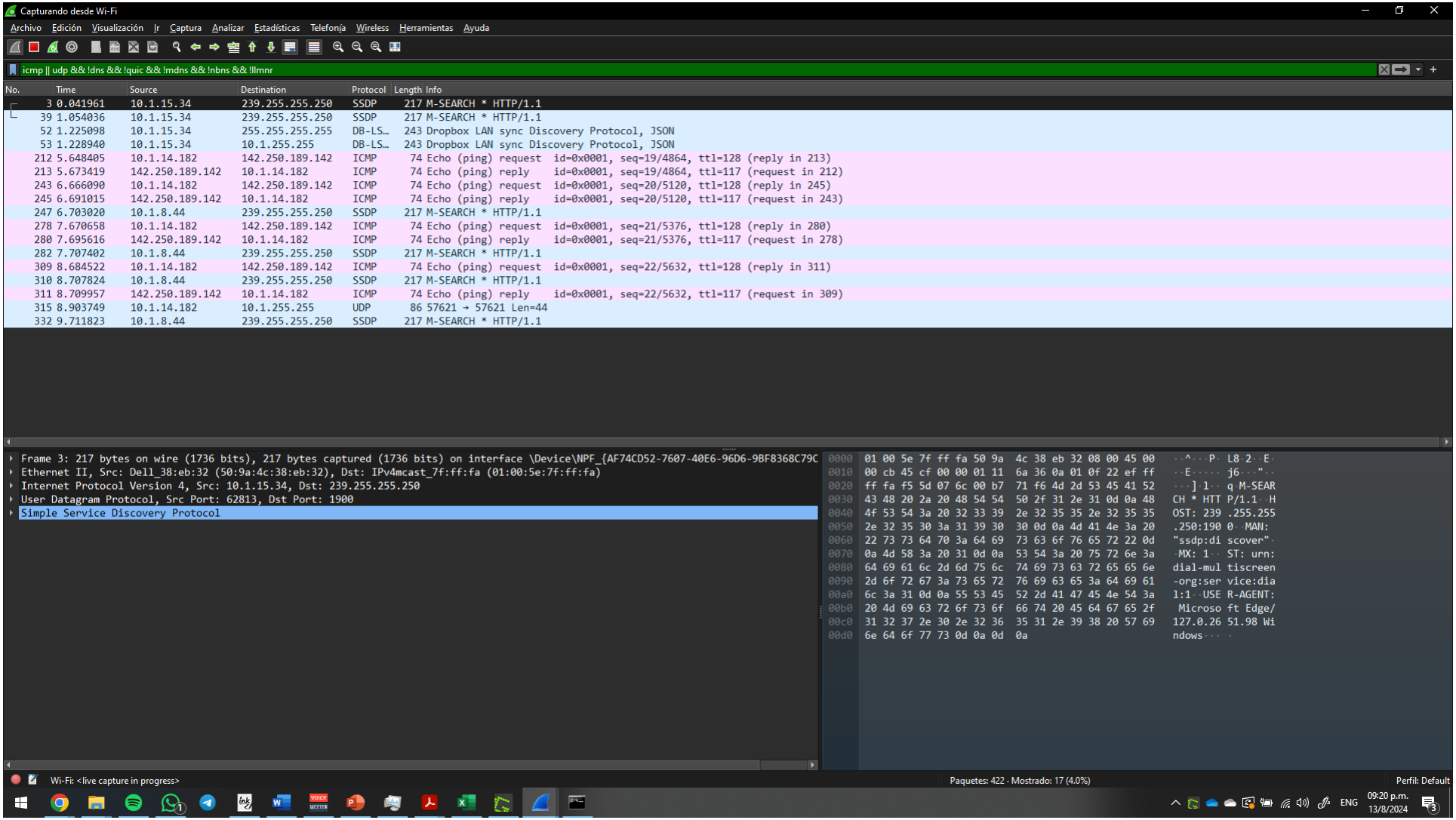
Sección ICMP

1. Captura de paquetes

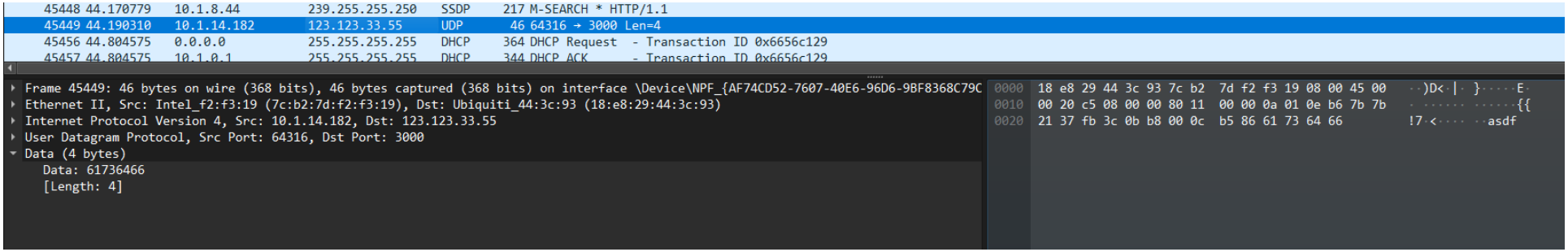
Tracert google.com ----- Port Unreachable, Host unreachable, Time-to-live exceeded



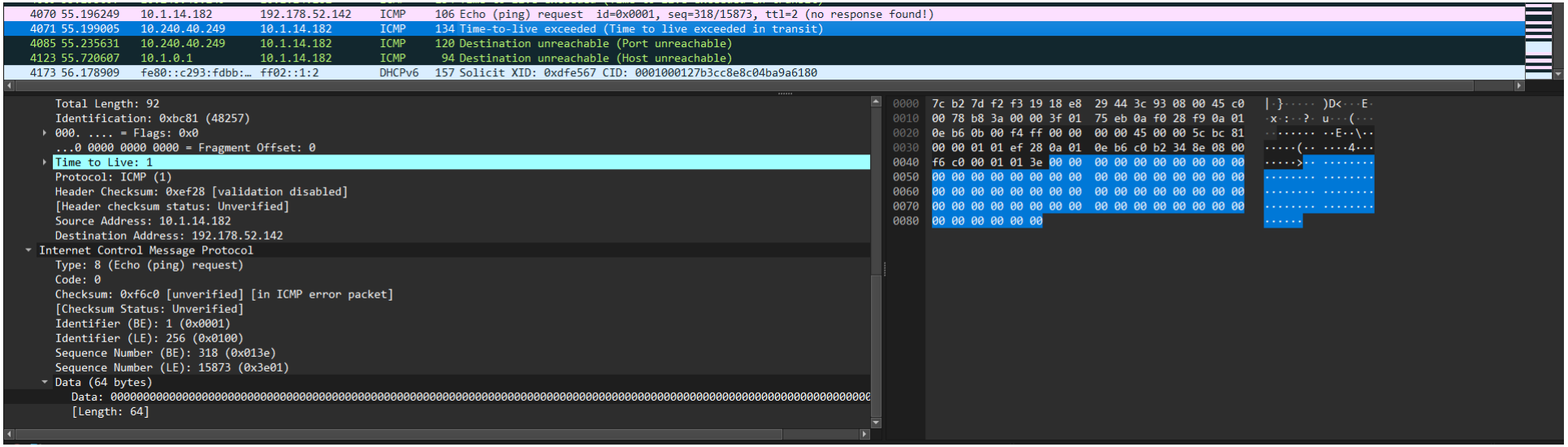
ping google.com ----- Echo Reply y Request



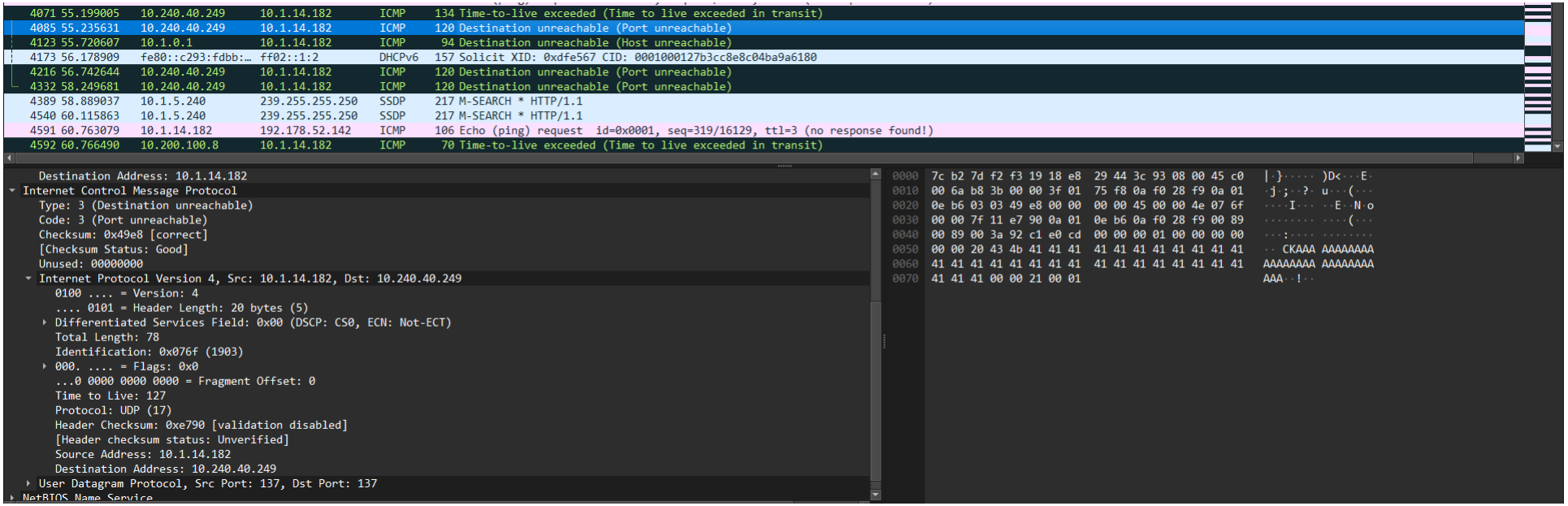
Lab 01: java Client Protocol UDP server 123.123.33.55



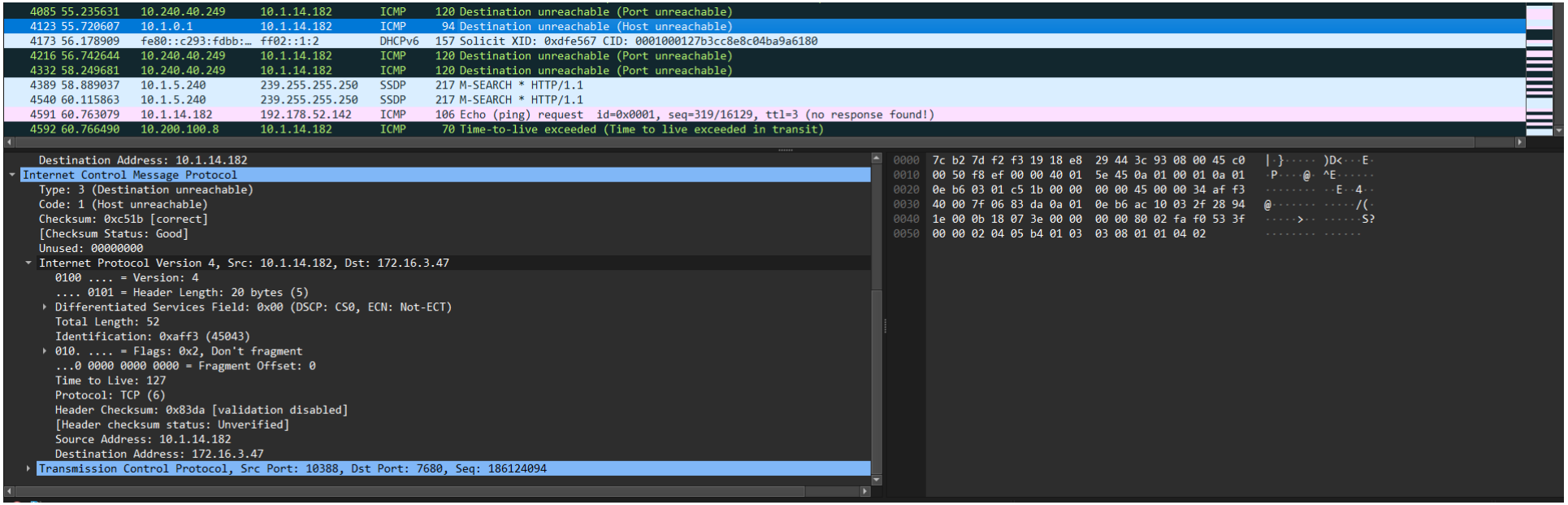
Time-to-live exceeeded:



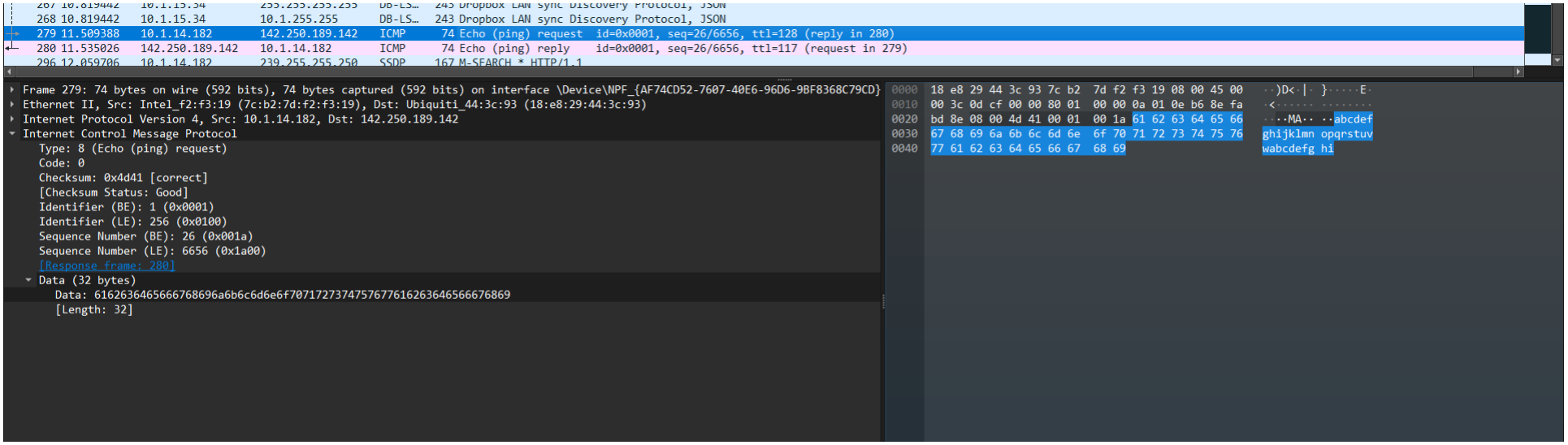
Port Unreachable:



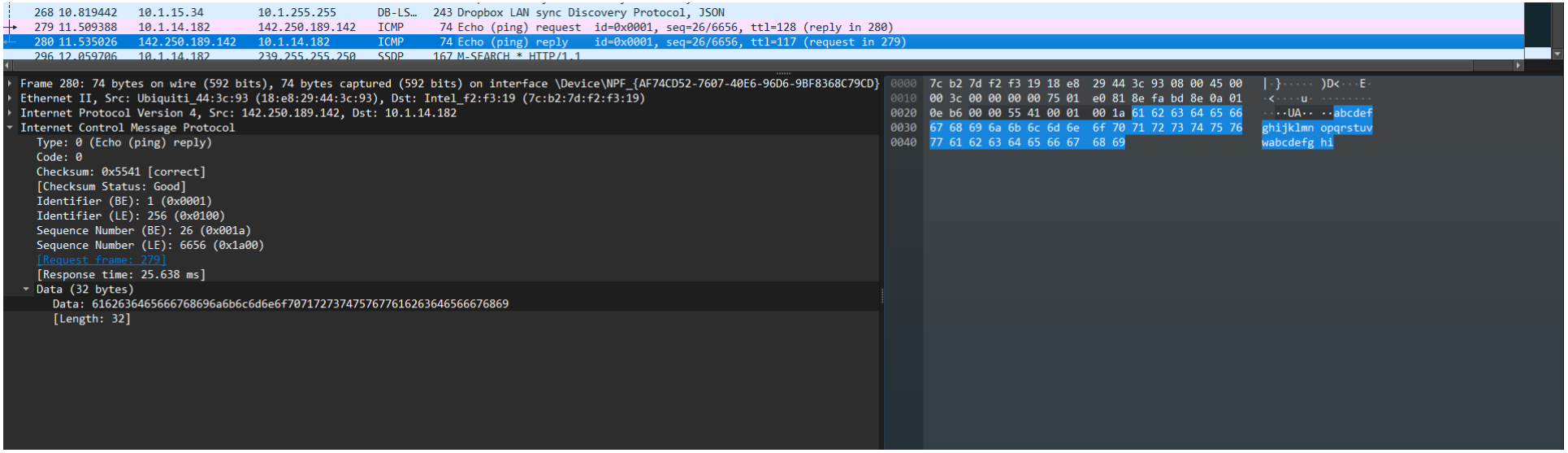
Host Unreachable:



Ping request:



Ping reply:


268 10.819442 10.1.15.34 10.1.255.255 DB-LS… 243 Dropbox LAN sync Discovery Protocol, JSON
279 11.509388 10.1.14.182 142.250.189.142 ICMP 74 Echo (ping) request id=0x0001, seq=26/6656, ttl=128 (reply in 280)
280 11.535026 142.250.189.142 10.1.14.182 ICMP 74 Echo (ping) reply id=0x0001, seq=26/6656, ttl=117 (request in 279)
296 12.059706 10.1.14.182 239.255.255.250 SSDP 167 M-SEARCH * HTTP/1.1
Frame 280: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface \Device\NPF_{AF74CD52-7607-40E6-96D6-9BF8368C79CD}
Ethernet II, Src: Ubiquiti_44:3c:93 (18:e8:29:44:3c:93), Dst: Intel_f2:f3:19 (7c:b2:7d:f2:f3:19)
Internet Protocol Version 4, Src: 142.250.189.142, Dst: 10.1.14.182
Internet Control Message Protocol
Type: 0 (Echo (ping) reply)
Code: 0
Checksum: 0x5541 [correct]
[Checksum Status: Good]
Identifier (BE): 1 (0x0001)
Identifier (LE): 256 (0x0100)
Sequence Number (BE): 26 (0x001a)
Sequence Number (LE): 6656 (0x1a00)
[Request frame: 279]
[Response time: 25.638 ms]
Data (32 bytes)
Data: 6162636465666768696a6b6c6d6e6f707172737475767761626364656667686 9
[Length: 32]
0000 7c b2 7d f2 f3 19 18 e8 29 44 3c 93 08 00 45 00
0010 00 3c 00 00 00 00 75 01 e0 81 8e fa bd 8e 0a 01
0020 0e b6 00 00 55 41 00 01 00 1a 61 62 63 64 65 66
0030 67 68 69 6a 6b 6c 6d 6e 6f 70 71 72 73 74 75 76
0040 77 61 62 63 64 65 66 67 68 69

2. Análisis de encabezados

No.	Source	Destination	Length	Type	Code	Checksum	Headers adicionales
Time-to-live exceeded							
4071	10.240.40.249	10.1.14.182	134	11	0	0xf4ff	
Port Unreachable							
4085	10.240.40.249	10.1.14.182	120	3	3	0xf4ff	
Host Unreachable							
4123	10.1.0.1	10.1.14.182	94	3	1	0xc51b	
Ping (request)							
279	10.1.14.182	142.250.189.142	74	0	0	0x4d41	id=0x0001, seq=26/6656, ttl=128 (reply in 280)
Ping (reply)							
280	142.250.189.142	10.1.14.182	74	0	0	0x5541	id=0x0001, seq=26/6656, ttl=117 (request in 279)

3. Preguntas de análisis

¿Cómo se pueden diferenciar los mensajes ICMP/IP de host y port unreachable?

Ambos son Type 3, pero el Code cambia:

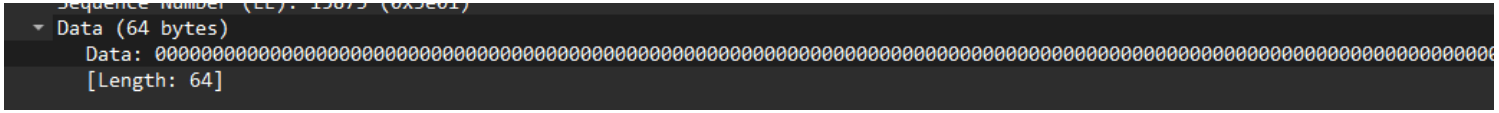
- Code: 1 (Host unreachable)
- Code: 3 (Port unreachable)

¿Cuál es la respuesta al enviar un UDP/IP con IP y PORT incorrectos y por qué?

No hay respuesta. En ambos casos solo se envía por UDP, pero no se recibe respuesta. Tampoco hay mensaje de error ni nada, porque UDP solo manda y no verifica si llega.

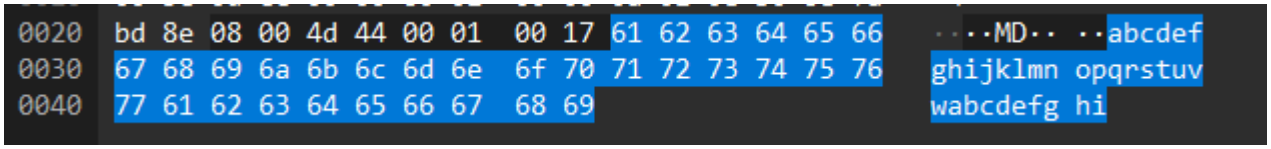
¿Qué contiene el mensaje ICMP de Time Exceeded en su data? ¿Viene vacío?

No, sí tiene DATA: 64 bytes de 0s:



¿Cuál es la particularidad de los mensajes ICMP Echo y Echo Reply?

Ambos llevan 32 bits de DATA: el abecedario:



Preguntas Generales

¿Qué diferencias observas en la captura de paquetes entre TCP y UDP?

En TCP vemos que hay más flags, y varios pasos previos antes de enviar el data. Vemos que siempre hay un handshake (SYN, SYN, ACK) y al finalizar hay un (FIN, ACK). En UDP no hay esto, solo se manda de una vez el paquete.

¿Qué información puedes extraer de los mensajes ICMP generados por las pruebas de UDP?

Son muy sencillos, similar a UDP. Pero llevan ciertos parámetros específicos del protocolo ICMP, que son el Type y el Code. Además del checksum que UDP ya implementa para verificaciones.

¿Cómo afectan los mensajes de control en TCP (como FIN, ACK) al cierre de la conexión?

El cliente manda un FIN para avisar que va a cerrar conexión, y luego el servidor manda su FIN, ACK. El cliente finaliza con un ACK, y ahí terminan los mensajes.

Esto afecta la conexión, ya que aunque se cierre, todavía se mandan unos últimos paquetes, asegurando que la conexión se cierre de ambos lados.

¿Qué diferencias notas en la ruta y los tiempos de respuesta entre TraceRoute y Ping?

Por observación, parece que TraceRoute busca varias rutsa para llegar a un servidor, en el caso default con un máximo de 30 saltos. Vemos que hay unas rutas que tardan más en responder, unas que no responden, y unas que responden bastante rápido.

En Ping al contrario, parece que encuentra la ruta más corta y solo manda unos paquetes de verificación, por lo que la respuesta es rápida.