

# Práctica Bloque III

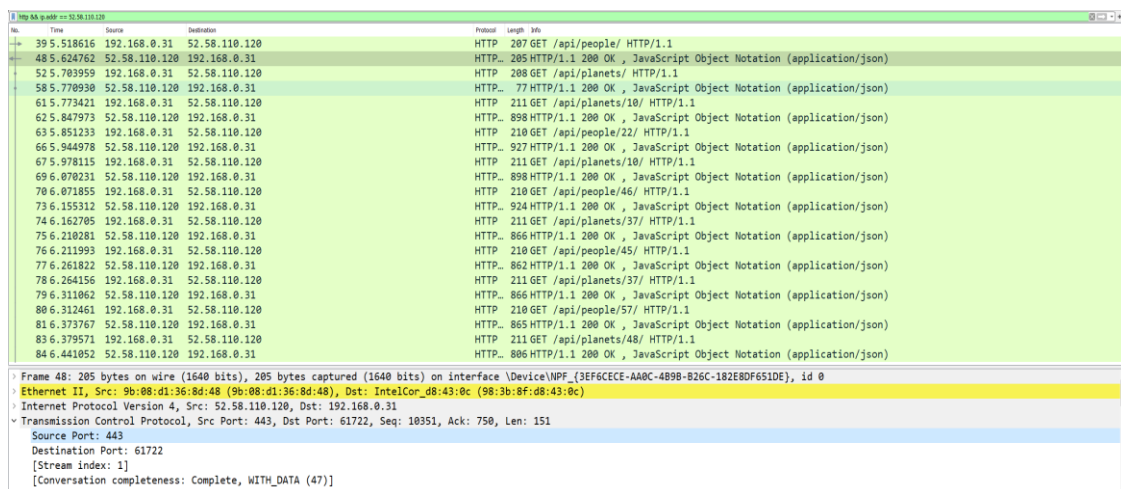
Alumno: **Ponce Arrocha, Santiago**

Titulación: Grado de Ingeniería Informática

PC de la práctica: **Realizado en casa.**

**Ejercicio 1.** ¿Cuál es el puerto utilizado por el servidor? ¿Es el normal de HTTP (80)? ¿Por qué?

- **Tramas analizadas:** Se utilizó la trama 48 para este ejemplo.



No.	Time	Source	Destination	Protocol	Length	Info
395	51.8616	192.168.0.31	52.58.110.120	HTTP	207	GET /api/people/ HTTP/1.1
485	62.4762	52.58.110.120	192.168.0.31	HTTP	205	HTTP/1.1 200 OK, JavaScript Object Notation (application/json)
525	70.9359	192.168.0.31	52.58.110.120	HTTP	208	GET /api/planets/ HTTP/1.1
585	77.0930	52.58.110.120	192.168.0.31	HTTP	77	HTTP/1.1 200 OK, JavaScript Object Notation (application/json)
615	77.3421	192.168.0.31	52.58.110.120	HTTP	211	GET /api/planets/10/ HTTP/1.1
625	84.7973	52.58.110.120	192.168.0.31	HTTP	898	HTTP/1.1 200 OK, JavaScript Object Notation (application/json)
635	85.1233	192.168.0.31	52.58.110.120	HTTP	210	GET /api/people/22/ HTTP/1.1
665	94.4978	52.58.110.120	192.168.0.31	HTTP	927	HTTP/1.1 200 OK, JavaScript Object Notation (application/json)
675	97.8115	192.168.0.31	52.58.110.120	HTTP	211	GET /api/planets/10/ HTTP/1.1
696	07.0231	52.58.110.120	192.168.0.31	HTTP	898	HTTP/1.1 200 OK, JavaScript Object Notation (application/json)
706	07.1855	192.168.0.31	52.58.110.120	HTTP	210	GET /api/people/46/ HTTP/1.1
736	15.5312	52.58.110.120	192.168.0.31	HTTP	924	HTTP/1.1 200 OK, JavaScript Object Notation (application/json)
746	16.2705	192.168.0.31	52.58.110.120	HTTP	211	GET /api/planets/37/ HTTP/1.1
756	21.0281	52.58.110.120	192.168.0.31	HTTP	866	HTTP/1.1 200 OK, JavaScript Object Notation (application/json)
766	21.1993	192.168.0.31	52.58.110.120	HTTP	210	GET /api/people/45/ HTTP/1.1
776	26.1822	52.58.110.120	192.168.0.31	HTTP	862	HTTP/1.1 200 OK, JavaScript Object Notation (application/json)
786	26.4156	192.168.0.31	52.58.110.120	HTTP	211	GET /api/planets/37/ HTTP/1.1
796	31.1062	52.58.110.120	192.168.0.31	HTTP	866	HTTP/1.1 200 OK, JavaScript Object Notation (application/json)
806	31.2461	192.168.0.31	52.58.110.120	HTTP	210	GET /api/people/57/ HTTP/1.1
816	37.3767	52.58.110.120	192.168.0.31	HTTP	865	HTTP/1.1 200 OK, JavaScript Object Notation (application/json)
836	37.9571	192.168.0.31	52.58.110.120	HTTP	211	GET /api/planets/48/ HTTP/1.1
846	44.1052	52.58.110.120	192.168.0.31	HTTP	806	HTTP/1.1 200 OK, JavaScript Object Notation (application/json)

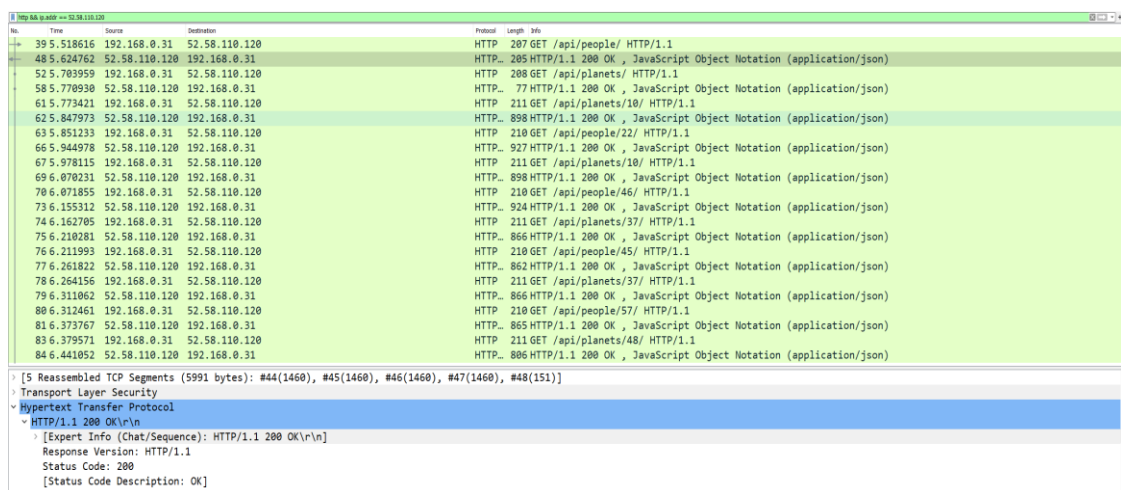
Frame 48: 205 bytes on wire (1640 bits), 205 bytes captured (1640 bits) on interface \Device\NPF\_{3EF6CECE-AA0C-4B9B-B26C-182E8DF651DE}, id 0  
Ethernet II, Src: 9b:08:d1:36:8d:48 (9b:08:d1:36:8d:48), Dst: IntelCor\_d8:43:0c (98:3b:8f:d8:43:0c)  
Internet Protocol Version 4, Src: 52.58.110.120, Dst: 192.168.0.31  
Transmission Control Protocol, Src Port: 443, Dst Port: 61722, Seq: 10351, Ack: 750, Len: 151  
Source Port: 443  
Destination Port: 61722  
[Stream index: 1]  
[Conversation completeness: Complete, WITH\_DATA (47)]

El puerto utilizado por el servidor es el 443.

Es el puerto que emplea HTTPS (HyperText Transfer Protocol Secure). Este se diferencia de HTTP en que transfiere los datos de forma segura.

**Ejercicio 2.** Observe el número de conexiones realizadas. ¿Cuántas hace? ¿Usa una conexión permanente (en la misma conexión hace varias peticiones) o no permanente (solo realiza una por conexión)? En caso de ser permanente, ¿qué cabecera de la petición indica que queremos que sea permanente?

- **Tramas analizadas:** Se utilizó la trama 48 para este ejemplo.



No.	Time	Source	Destination	Protocol	Length	Info
395	51.8616	192.168.0.31	52.58.110.120	HTTP	207	GET /api/people/ HTTP/1.1
485	62.4762	52.58.110.120	192.168.0.31	HTTP	205	HTTP/1.1 200 OK, JavaScript Object Notation (application/json)
525	70.9359	192.168.0.31	52.58.110.120	HTTP	208	GET /api/planets/ HTTP/1.1
585	77.0930	52.58.110.120	192.168.0.31	HTTP	77	HTTP/1.1 200 OK, JavaScript Object Notation (application/json)
615	77.3421	192.168.0.31	52.58.110.120	HTTP	211	GET /api/planets/10/ HTTP/1.1
625	84.7973	52.58.110.120	192.168.0.31	HTTP	898	HTTP/1.1 200 OK, JavaScript Object Notation (application/json)
635	85.1233	192.168.0.31	52.58.110.120	HTTP	210	GET /api/people/22/ HTTP/1.1
665	94.4978	52.58.110.120	192.168.0.31	HTTP	927	HTTP/1.1 200 OK, JavaScript Object Notation (application/json)
675	97.8115	192.168.0.31	52.58.110.120	HTTP	211	GET /api/planets/10/ HTTP/1.1
696	07.0231	52.58.110.120	192.168.0.31	HTTP	898	HTTP/1.1 200 OK, JavaScript Object Notation (application/json)
706	07.1855	192.168.0.31	52.58.110.120	HTTP	210	GET /api/people/46/ HTTP/1.1
736	15.5312	52.58.110.120	192.168.0.31	HTTP	924	HTTP/1.1 200 OK, JavaScript Object Notation (application/json)
746	16.2705	192.168.0.31	52.58.110.120	HTTP	211	GET /api/planets/37/ HTTP/1.1
756	21.0281	52.58.110.120	192.168.0.31	HTTP	866	HTTP/1.1 200 OK, JavaScript Object Notation (application/json)
766	21.1993	192.168.0.31	52.58.110.120	HTTP	210	GET /api/people/45/ HTTP/1.1
776	26.1822	52.58.110.120	192.168.0.31	HTTP	862	HTTP/1.1 200 OK, JavaScript Object Notation (application/json)
786	26.4156	192.168.0.31	52.58.110.120	HTTP	211	GET /api/planets/37/ HTTP/1.1
796	31.1062	52.58.110.120	192.168.0.31	HTTP	866	HTTP/1.1 200 OK, JavaScript Object Notation (application/json)
806	31.2461	192.168.0.31	52.58.110.120	HTTP	210	GET /api/people/57/ HTTP/1.1
816	37.3767	52.58.110.120	192.168.0.31	HTTP	865	HTTP/1.1 200 OK, JavaScript Object Notation (application/json)
836	37.9571	192.168.0.31	52.58.110.120	HTTP	211	GET /api/planets/48/ HTTP/1.1
846	44.1052	52.58.110.120	192.168.0.31	HTTP	806	HTTP/1.1 200 OK, JavaScript Object Notation (application/json)

[5 Reassembled TCP Segments (5991 bytes): #44(1460), #45(1460), #46(1460), #47(1460), #48(151)]  
Transport Layer Security  
Hypertext Transfer Protocol  
HTTP/1.1 200 OK\r\n  
[Expert Info (Chat/Sequence): HTTP/1.1 200 OK\r\n]  
Response Version: HTTP/1.1  
Status Code: 200  
[Status Code Description: OK]

```

v Hypertext Transfer Protocol
v GET /api/people/ HTTP/1.1\r\n
  > [Expert Info (Chat/Sequence): GET /api/people/ HTTP/1.1\r\n]
    Request Method: GET
    Request URI: /api/people/
    Request Version: HTTP/1.1
    Accept: application/json\r\n
    User-Agent: Star Wars-2022\r\n
    Host: swapi.dev\r\n
    Connection: keep-alive\r\n

```

Se realiza una única conexión de forma permanente, haciendo varias repeticiones, usándola de forma permanente.

La cabecera que lo indica es la destacada en la segunda captura de pantalla (Connection: keep-alive \r\n).

**Ejercicio 3.** Describa el significado de las cabeceras de una petición y una respuesta (sin incluir las x-\*).

- **Tramas analizadas:** Se utilizó la trama 39 y 48 para este ejemplo.

No.	Time	Source	Destination	Protocol	Length	Info
39	5.518616	192.168.0.31	52.58.110.120	HTTP	207	GET /api/people/ HTTP/1.1
48	5.624762	52.58.110.120	192.168.0.31	HTTP	205	HTTP/1.1 200 OK , JavaScript Object Notation (application/json)
52	5.783959	192.168.0.31	52.58.110.120	HTTP	208	GET /api/planets/ HTTP/1.1
58	5.770930	52.58.110.120	192.168.0.31	HTTP	77	HTTP/1.1 200 OK , JavaScript Object Notation (application/json)
61	5.773421	192.168.0.31	52.58.110.120	HTTP	211	GET /api/planets/10/ HTTP/1.1
62	5.847973	52.58.110.120	192.168.0.31	HTTP	898	HTTP/1.1 200 OK , JavaScript Object Notation (application/json)
63	5.851233	192.168.0.31	52.58.110.120	HTTP	210	GET /api/people/22/ HTTP/1.1
66	5.944978	52.58.110.120	192.168.0.31	HTTP	927	HTTP/1.1 200 OK , JavaScript Object Notation (application/json)
67	5.978115	192.168.0.31	52.58.110.120	HTTP	211	GET /api/planets/10/ HTTP/1.1
69	6.070231	52.58.110.120	192.168.0.31	HTTP	898	HTTP/1.1 200 OK , JavaScript Object Notation (application/json)
70	6.071855	192.168.0.31	52.58.110.120	HTTP	210	GET /api/people/46/ HTTP/1.1
73	6.155312	52.58.110.120	192.168.0.31	HTTP	924	HTTP/1.1 200 OK , JavaScript Object Notation (application/json)
74	6.162705	192.168.0.31	52.58.110.120	HTTP	211	GET /api/planets/37/ HTTP/1.1
75	6.210281	52.58.110.120	192.168.0.31	HTTP	865	HTTP/1.1 200 OK , JavaScript Object Notation (application/json)
76	6.211993	192.168.0.31	52.58.110.120	HTTP	210	GET /api/people/45/ HTTP/1.1
77	6.261822	52.58.110.120	192.168.0.31	HTTP	862	HTTP/1.1 200 OK , JavaScript Object Notation (application/json)
78	6.264156	192.168.0.31	52.58.110.120	HTTP	211	GET /api/planets/37/ HTTP/1.1
79	6.311062	52.58.110.120	192.168.0.31	HTTP	866	HTTP/1.1 200 OK , JavaScript Object Notation (application/json)
80	6.312461	192.168.0.31	52.58.110.120	HTTP	210	GET /api/people/57/ HTTP/1.1
81	6.373767	52.58.110.120	192.168.0.31	HTTP	865	HTTP/1.1 200 OK , JavaScript Object Notation (application/json)
83	6.379571	192.168.0.31	52.58.110.120	HTTP	211	GET /api/planets/48/ HTTP/1.1
84	6.441052	52.58.110.120	192.168.0.31	HTTP	806	HTTP/1.1 200 OK , JavaScript Object Notation (application/json)

Transport Layer Security
Hypertext Transfer Protocol
GET /api/people/ HTTP/1.1\r\n
> [Expert Info (Chat/Sequence): GET /api/people/ HTTP/1.1\r\n]
Request Method: GET
Request URI: /api/people/
Request Version: HTTP/1.1
Accept: application/json\r\n
User-Agent: Star Wars-2022\r\n
Host: swapi.dev\r\n
Connection: keep-alive\r\n
\r\n
[Full request URI: https://swapi.dev/api/people/]
[HTTP request 1/11]
[Response in frame: 48]

**Accept:** Nos dice el tipo de contenido que el cliente puede procesar.

**User-Agent:** Envía la información del cliente o del usuario al servidor.

**Host:** Especifica el nombre del dominio del servidor, proporcionando información del servidor.

**Connection:** Nos informa sobre la conexión establecida (si es permanente o si no).

No.	Time	Source	Destination	Protocol	Length	Info
39.5	518616	192.168.0.31	52.58.110.120	HTTP	207	GET /api/people/ HTTP/1.1
48.5	624762	52.58.110.120	192.168.0.31	HTTP	205	HTTP/1.1 200 OK , JavaScript Object Notation (application/json)
52.5	703959	192.168.0.31	52.58.110.120	HTTP	208	GET /api/planets/ HTTP/1.1
58.5	770930	52.58.110.120	192.168.0.31	HTTP	77	HTTP/1.1 200 OK , JavaScript Object Notation (application/json)
61.5	773421	192.168.0.31	52.58.110.120	HTTP	211	GET /api/planets/10/ HTTP/1.1
62.5	847973	52.58.110.120	192.168.0.31	HTTP	898	HTTP/1.1 200 OK , JavaScript Object Notation (application/json)
63.5	851233	192.168.0.31	52.58.110.120	HTTP	210	GET /api/people/22/ HTTP/1.1
66.5	944978	52.58.110.120	192.168.0.31	HTTP	927	HTTP/1.1 200 OK , JavaScript Object Notation (application/json)
67.5	978115	192.168.0.31	52.58.110.120	HTTP	211	GET /api/planets/10/ HTTP/1.1
69.6	070231	52.58.110.120	192.168.0.31	HTTP	898	HTTP/1.1 200 OK , JavaScript Object Notation (application/json)
70.6	071855	192.168.0.31	52.58.110.120	HTTP	210	GET /api/people/46/ HTTP/1.1
73.6	155312	52.58.110.120	192.168.0.31	HTTP	924	HTTP/1.1 200 OK , JavaScript Object Notation (application/json)
74.6	162705	192.168.0.31	52.58.110.120	HTTP	211	GET /api/planets/37/ HTTP/1.1
75.6	210281	52.58.110.120	192.168.0.31	HTTP	866	HTTP/1.1 200 OK , JavaScript Object Notation (application/json)
76.6	211993	192.168.0.31	52.58.110.120	HTTP	210	GET /api/people/45/ HTTP/1.1
77.6	261822	52.58.110.120	192.168.0.31	HTTP	862	HTTP/1.1 200 OK , JavaScript Object Notation (application/json)
78.6	264156	192.168.0.31	52.58.110.120	HTTP	211	GET /api/planets/37/ HTTP/1.1
79.6	311062	52.58.110.120	192.168.0.31	HTTP	866	HTTP/1.1 200 OK , JavaScript Object Notation (application/json)
80.6	312461	192.168.0.31	52.58.110.120	HTTP	210	GET /api/people/57/ HTTP/1.1
81.6	373767	52.58.110.120	192.168.0.31	HTTP	865	HTTP/1.1 200 OK , JavaScript Object Notation (application/json)
83.6	379571	192.168.0.31	52.58.110.120	HTTP	211	GET /api/planets/48/ HTTP/1.1
84.6	441052	52.58.110.120	192.168.0.31	HTTP	806	HTTP/1.1 200 OK , JavaScript Object Notation (application/json)

Hypertext Transfer Protocol	
HTTP/1.1 200 OK\r\n	
[Expert Info (Chat/Sequence): HTTP/1.1 200 OK\r\n]	
Response Version: HTTP/1.1	
Status Code: 200	
[Status Code Description: OK]	
Response Phrase: OK	
Server: nginx/1.16.1\r\n	
Date: Tue, 14 Jun 2022 17:50:47 GMT\r\n	
Content-Type: application/json\r\n	
Transfer-Encoding: chunked\r\n	
Connection: keep-alive\r\n	
Vary: Accept, Cookie\r\n	
X-Frame-Options: SAMEORIGIN\r\n	
ETag: "b493126da58af6fec015ec116fec193"\r\n	
Allow: GET, HEAD, OPTIONS\r\n	
Strict-Transport-Security: max-age=15768000\r\n	
\r\n	
[HTTP response 1/11]	
[Time since request: 0.106146000 seconds]	

**Server:** Software usado en el servidor.

**Date:** Fecha y hora de la conexión.

**Content-Type:** Tipo de fichero.

**Transfer-Encoding:** Tipo de codificación.

**Connection:** Nos informa sobre la conexión establecida.

**Vary:** Estudia la forma de hacer que las próximas cabeceras coincidan.

**X-Frame-Options:** Indica al navegador si tiene permiso para renderizar una página en formato <frame>, <iframe> o <object>.

**ETag:** Versión del recurso.

**Allow:** Indica los métodos permitidos por un recurso.

**Strict-Transport-Security:** Indica si se debe realizar la comunicación de forma segura (HTTPS) en vez de con HTTP.

**Ejercicio 4.** Filtre por el protocolo **rtsp** y use la opción **Follow TCP Stream** de Wireshark para observar el diálogo completo que han mantenido el cliente de correo y el servidor. Explique brevemente (una línea) el significado de cada comando enviado por el cliente (si algún comando se repite solo debe explicarlo una vez).

- Tramas analizadas: 72, 76, 82, 84, 94, 1079, 1155, 1532, 1545, 2355

No.	Time	Source	Destination	Protocol	Length	Info
69.1	889618	192.168.0.31	34.227.104.115	TCP	66	54100 → 554 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
70.1	985708	34.227.104.115	192.168.0.31	TCP	66	554 → 54100 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460 SACK_PERM=1 WS=1
71.1	985820	192.168.0.31	34.227.104.115	TCP	66	54100 → 554 [ACK] Seq=65536 Win=65535 Len=0
72.1	995837	192.168.0.31	34.227.104.115	RTSP	108	RTSP/1.0 200 OK
74.2	091458	34.227.104.115	192.168.0.31	RTSP	108	RTSP/1.0 200 OK
75.2	091458	34.227.104.115	192.168.0.31	RTSP	108	RTSP/1.0 200 OK
76.2	091757	192.168.0.31	34.227.104.115	RTSP	108	RTSP/1.0 200 OK
79.2	195625	34.227.104.115	192.168.0.31	RTSP	108	RTSP/1.0 200 OK
82.2	196762	192.168.0.31	34.227.104.115	RTSP	108	RTSP/1.0 200 OK
83.2	298001	34.227.104.115	192.168.0.31	RTSP	108	RTSP/1.0 200 OK
84.2	299646	192.168.0.31	34.227.104.115	RTSP	108	RTSP/1.0 200 OK
85.2	402421	34.227.104.115	192.168.0.31	RTSP	108	RTSP/1.0 200 OK
94.2	483663	192.168.0.31	34.227.104.115	RTSP	108	RTSP/1.0 200 OK
96.2	501149	34.227.104.115	192.168.0.31	RTSP	108	RTSP/1.0 200 OK
97.2	547306	192.168.0.31	34.227.104.115	RTSP	108	RTSP/1.0 200 OK
10.20	451360	192.168.0.31	34.227.104.115	RTSP	108	RTSP/1.0 200 OK
10.20	550360	34.227.104.115	192.168.0.31	RTSP	108	RTSP/1.0 200 OK
10.20	593861	192.168.0.31	34.227.104.115	RTSP	108	RTSP/1.0 200 OK
11.26	671587	192.168.0.31	34.227.104.115	RTSP	108	RTSP/1.0 200 OK
11.26	772379	34.227.104.115	192.168.0.31	RTSP	108	RTSP/1.0 200 OK
11.26	814522	192.168.0.31	34.227.104.115	RTSP	108	RTSP/1.0 200 OK
15.32	735946	192.168.0.31	34.227.104.115	RTSP	108	RTSP/1.0 200 OK

Frame 72: 219 bytes on wire (175 bytes captured on interface 0)	
Ethernet II, Src: IntelCor_d8:43:16, Dst: 34:227:104:115	
Internet Protocol Version 4, Src: 192.168.0.31, Destination: 34.227.104.115	
Transmission Control Protocol, Src Port: 54100, Destination Port: 554	
Real Time Streaming Protocol	
[Community ID: 1:cxj1P+QVY7+AXX]	

RTSP/1.0	
/trackID=1 RTSP/1.0	
/trackID=2 RTSP/1.0	
RTSP/1.0	
/ RTSP/1.0	
RTSP/1.0	
/ RTSP/1.0	

```
OPTIONS rtsp://wowzaec2demo.streamlock.net:554/vod/mp4:BigBuckBunny_115k.mp4 RTSP/1.0
CSeq: 2
User-Agent: LibVLC/3.0.17.4 (LIVE555 Streaming Media v2016.11.28)
```

**Options rtsp:** Hace saber al servidor el enlace que debe aceptar.

**User-Agent:** Envía la información del cliente al servidor.

```
DESCRIBE rtsp://wowzaec2demo.streamlock.net:554/vod/mp4:BigBuckBunny_115k.mp4 RTSP/1.0
CSeq: 3
User-Agent: LibVLC/3.0.17.4 (LIVE555 Streaming Media v2016.11.28)
Accept: application/sdp
```

**DESCRIBE:** Cliente muestra el contenido de la URL.

```
SETUP rtsp://wowzaec2demo.streamlock.net:554/vod/mp4:BigBuckBunny_115k.mp4/trackID=2 RTSP/1.0
CSeq: 5
User-Agent: LibVLC/3.0.17.4 (LIVE555 Streaming Media v2016.11.28)
Transport: RTP/AVP;unicast;client_port=56518-56519
Session: 1268367094
```

**SETUP:** Cliente le pregunta al servidor donde puede conseguir los datos.

```
PLAY rtsp://wowzaec2demo.streamlock.net:554/vod/mp4:BigBuckBunny_115k.mp4/ RTSP/1.0
CSeq: 6
User-Agent: LibVLC/3.0.17.4 (LIVE555 Streaming Media v2016.11.28)
Session: 1268367094
Range: npt=0.000-
```

**PLAY:** Comienza la reproducción del video (el cliente le indica al servidor que puede comenzar a enviar los flujos de datos).

```
PAUSE rtsp://wowzaec2demo.streamlock.net:554/vod/mp4:BigBuckBunny_115k.mp4/ RTSP/1.0
CSeq: 7
User-Agent: LibVLC/3.0.17.4 (LIVE555 Streaming Media v2016.11.28)
Session: 1268367094
```

**PAUSE:** Se pausa el video (el cliente deja de recibir flujo de datos, pero no libera los recursos ya recibidos desde el PLAY).

```
TEARDOWN rtsp://wowzaec2demo.streamlock.net:554/vod/mp4:BigBuckBunny_115k.mp4/ RTSP/1.0
CSeq: 11
User-Agent: LibVLC/3.0.17.4 (LIVE555 Streaming Media v2016.11.28)
Session: 1268367094
```

**TEARDOWN:** Se para el video (el cliente solicita al servidor detener definitivamente el flujo de datos y le solicita liberar todos los recursos recibidos).

**Ejercicio 5.** ¿Por qué se hacen dos comandos SETUP? ¿Cómo sabía que debía hacer dos comandos de ese estilo?

- **Tramas analizadas:** Trama 82, 83, 84, 85

Un SETUP es para el envío del video (trackID=1) y el otro para el envío del audio (trackID=2).

In this example, the client is recommended to establish a single RTSP session to the server, and it uses the URIs `rtsp://example.com/movie/trackID=1` and `rtsp://example.com/movie/trackID=2` to set up the video and audio streams, respectively. The URI `rtsp://example.com/movie/`, which is resolved from the "\*", controls the whole presentation (movie).

Los dos SETUPS son los siguientes:

```

SETUP rtsp://wowzaec2demo.streamlock.net:554/vod/mp4:BigBuckBunny_115k.mp4/trackID=1 RTSP/1.0
CSeq: 4
User-Agent: LibVLC/3.0.17.4 (LIVE555 Streaming Media v2016.11.28)
Transport: RTP/AVP;unicast;client_port=56516-56517

RTSP/1.0 200 OK
CSeq: 4
Server: Wowza Streaming Engine 4.8.18+1 build20220318091926
Cache-Control: no-cache
Expires: Wed, 15 Jun 2022 07:19:38 UTC
Transport: RTP/AVP;unicast;client_port=56516-56517;source=34.227.104.115;server_port=7062-7063;ssrc=11ADB57E
Date: Wed, 15 Jun 2022 07:19:38 UTC
Session: 1268367094;timeout=60

SETUP rtsp://wowzaec2demo.streamlock.net:554/vod/mp4:BigBuckBunny_115k.mp4/trackID=2 RTSP/1.0
CSeq: 5
User-Agent: LibVLC/3.0.17.4 (LIVE555 Streaming Media v2016.11.28)
Transport: RTP/AVP;unicast;client_port=56518-56519
Session: 1268367094

RTSP/1.0 200 OK
CSeq: 5
Server: Wowza Streaming Engine 4.8.18+1 build20220318091926
Cache-Control: no-cache
Expires: Wed, 15 Jun 2022 07:19:38 UTC
Transport: RTP/AVP;unicast;client_port=56518-56519;source=34.227.104.115;server_port=7192-7193;ssrc=5A38D8CE
Date: Wed, 15 Jun 2022 07:19:38 UTC
Session: 1268367094;timeout=60

```

**Ejercicio 6.** ¿Qué comandos ha provocado adelantar la reproducción del vídeo? ¿Cómo indica por donde debe seguir la reproducción tras el cambio?

- **Tramas analizadas: 1545, 1547, 2355**

El comando "PAUSE". Gracias a la variable "Range", el cliente le indica al servidor la parte del video desde donde debe empezar a devolver el flujo de datos.

```

PLAY rtsp://wowzaec2demo.streamlock.net:554/vod/mp4:BigBuckBunny_115k.mp4/ RTSP/1.0
CSeq: 10
User-Agent: LibVLC/3.0.17.4 (LIVE555 Streaming Media v2016.11.28)
Session: 1268367094
Range: npt=330.830-

RTSP/1.0 200 OK
RTP-Info: url=rtsp://wowzaec2demo.streamlock.net:554/vod/mp4:BigBuckBunny_115k.mp4/trackID=1;seq=241;rtptime=3972096,url=rtsp://wowzaec2demo.streamlock.net:554/vod/mp4:BigBuckBunny_115k.mp4/trackID=2;seq=770;rtptime=29790720
CSeq: 10
Server: Wowza Streaming Engine 4.8.18+1 build20220318091926
Cache-Control: no-cache
Range: npt=330.83-634.625
Session: 1268367094;timeout=60

TEARDOWN rtsp://wowzaec2demo.streamlock.net:554/vod/mp4:BigBuckBunny_115k.mp4/ RTSP/1.0
CSeq: 11
User-Agent: LibVLC/3.0.17.4 (LIVE555 Streaming Media v2016.11.28)
Session: 1268367094

```



**Ejercicio 7.** Si observa los comandos y las respuestas son muy similares a las que usa HTTP. Indique dos cabeceras que use RTSP que también se usen en HTTP e indique (y explique) dos cabeceras de RTSP que no se usen en HTTP.

- **Tramas analizadas: 96**

-Por ejemplo, **Range** y **User-Agent** se comparten en ambos protocolos.

-**RTP-Info** y **Session** se comparten en ambos protocolos.

**RTP-Info:** “Real-Time Transport Protocol”. Campo usado para establecer un parámetro específico para, en nuestro caso, indicar la URL de la transmisión.

**Session:** Indicador de sesión proporcionado por el servidor, que el cliente debe devolver para cualquier solicitud relacionada con esa sesión.

**Ejercicio 8.** Ahora filtre por el protocolo rtp que se utiliza para transmitir el recurso multimedia tal cual. ¿Cómo se decidieron los puertos a utilizar en estas comunicaciones RTP? ¿Se confirman de alguna forma cada uno de los envíos RTP?

- **Tramas analizadas: 99**

No.	Time	Source	Destination	Protocol	Length	Info
99	2.622366	34.227.104.115	192.168.0.31	RTP	10...	PT=DynamicRTP-Type-97, SSRC=0x5A38D8CE, Seq=1, Time=
100	2.622366	34.227.104.115	192.168.0.31	RTP	295	PT=DynamicRTP-Type-96, SSRC=0x11ADB57E, Seq=1, Time=
102	2.718771	34.227.104.115	192.168.0.31	RTP	217	PT=DynamicRTP-Type-97, SSRC=0x5A38D8CE, Seq=2, Time=
103	2.718771	34.227.104.115	192.168.0.31	RTP	67	PT=DynamicRTP-Type-97, SSRC=0x5A38D8CE, Seq=3, Time=
104	2.718771	34.227.104.115	192.168.0.31	RTP	295	PT=DynamicRTP-Type-96, SSRC=0x11ADB57E, Seq=2, Time=
105	2.843262	34.227.104.115	192.168.0.31	RTP	434	PT=DynamicRTP-Type-97, SSRC=0x5A38D8CE, Seq=4, Time=
106	2.843262	34.227.104.115	192.168.0.31	RTP	13...	PT=DynamicRTP-Type-97, SSRC=0x5A38D8CE, Seq=5, Time=
107	2.843262	34.227.104.115	192.168.0.31	RTP	296	PT=DynamicRTP-Type-96, SSRC=0x11ADB57E, Seq=3, Time=
108	2.919170	34.227.104.115	192.168.0.31	RTP	62	PT=DynamicRTP-Type-97, SSRC=0x5A38D8CE, Seq=6, Time=
109	2.919170	34.227.104.115	192.168.0.31	RTP	14...	PT=DynamicRTP-Type-97, SSRC=0x5A38D8CE, Seq=7, Time=
110	2.919170	34.227.104.115	192.168.0.31	RTP	77	PT=DynamicRTP-Type-97, SSRC=0x5A38D8CE, Seq=8, Time=
111	2.919170	34.227.104.115	192.168.0.31	RTP	78	PT=DynamicRTP-Type-97, SSRC=0x5A38D8CE, Seq=9, Time=
112	2.919170	34.227.104.115	192.168.0.31	RTP	307	PT=DynamicRTP-Type-96, SSRC=0x11ADB57E, Seq=4, Time=
113	3.019994	34.227.104.115	192.168.0.31	RTP	83	PT=DynamicRTP-Type-97, SSRC=0x5A38D8CE, Seq=10, Time=
114	3.019994	34.227.104.115	192.168.0.31	RTP	82	PT=DynamicRTP-Type-97, SSRC=0x5A38D8CE, Seq=11, Time=

>	Frame 99: 1068 bytes on wire (8544 bits), 1068 bytes captured (8544 bits) on interface \Device\NPF_{3EF6CECE-AA0C-4B9B-B26C-182E8DF651DE}, id 0
>	Ethernet II, Src: 9b:08:d1:36:8d:48 (9b:08:d1:36:8d:48), Dst: IntelCor_d8:43:0c (98:3b:8f:d8:43:0c)
>	Internet Protocol Version 4, Src: 34.227.104.115, Dst: 192.168.0.31
>	User Datagram Protocol, Src Port: 7192, Dst Port: 56518
>	Source Port: 7192
>	Destination Port: 56518
>	Length: 1034
>	Checksum: 0xd2db [unverified]
>	[Checksum Status: Unverified]
>	[Stream index: 7]
>	[Timestamps]
>	UDP payload (1026 bytes)
>	Real-Time Transport Protocol
>	[Community ID: 1:8YHfMH4QUSid5TwKBgsmm/UhX8A=]

Observando los puertos de la capa de transporte se puede observar los puertos del emisor (7192) y del destinatario (56518).

166	4.227854	34.227.104.115	192.168.0.31	RTP	395	PT=DynamicRTP-Type-96, SSRC=0x11ADB57E, Seq=17, Time=18432, Mark
167	4.299208	34.227.104.115	192.168.0.31	RTCP	82	Sender Report Source description
168	4.328730	34.227.104.115	192.168.0.31	RTP	78	PT=DynamicRTP-Type-97, SSRC=0x5A38D8CE, Seq=45, Time=150030, Mark
169	4.328730	34.227.104.115	192.168.0.31	RTP	294	PT=DynamicRTP-Type-97, SSRC=0x5A38D8CE, Seq=46, Time=157500, Mark
170	4.328730	34.227.104.115	192.168.0.31	RTP	163	PT=DynamicRTP-Type-97, SSRC=0x5A38D8CE, Seq=47, Time=164970, Mark
171	4.328730	34.227.104.115	192.168.0.31	RTP	785	PT=DynamicRTP-Type-96, SSRC=0x11ADB57E, Seq=18, Time=19456, Mark
172	4.415462	34.227.104.115	192.168.0.31	RTCP	82	Sender Report Source description
173	4.429266	34.227.104.115	192.168.0.31	RTP	67	PT=DynamicRTP-Type-97, SSRC=0x5A38D8CE, Seq=48, Time=161280, Mark
23...	47.084386	34.227.104.115	192.168.0.31	RTP	390	PT=DynamicRTP-Type-96, SSRC=0x11ADB57E, Seq=379, Time=4140032, Mark
23...	47.148823	192.168.0.31	34.227.104.115	RTCP	82	Receiver Report Goodbye
23...	47.213590	34.227.104.115	192.168.0.31	RTP	118	PT=DynamicRTP-Type-97, SSRC=0x5A38D8CE, Seq=1202, Time=31061250, Mark
23...	47.213590	34.227.104.115	192.168.0.31	RTP	193	PT=DynamicRTP-Type-97, SSRC=0x5A38D8CE, Seq=1203, Time=31065030, Mark
23...	47.213590	34.227.104.115	192.168.0.31	RTP	154	PT=DynamicRTP-Type-97, SSRC=0x5A38D8CE, Seq=1204, Time=31068720, Mark
23...	47.213866	34.227.104.115	192.168.0.31	RTP	354	PT=DynamicRTP-Type-96, SSRC=0x11ADB57E, Seq=380, Time=4141056, Mark

Las tramas RTCP confirman que las trama RTP se hayan enviado, mostrando una descripción.

**Bibliografía de apoyo para el desarrollo de la práctica:**

- <https://datatracker.ietf.org/doc/html/rfc2326#page-57>