



**Universidade do Minho**  
Escola de Engenharia

Mestrado em Engenharia Informática  
Engenharia de Serviços em Rede

**Trabalho Prático nº2**

**Streaming de áudio e vídeo a pedido e em tempo real**

Ano Letivo 2021/2022

**Grupo 6 - PL1**

Ana Filipa Pereira PG46978  
Carolina Santejo PG47102  
Raquel Costa PG47600

## Conteúdo

<b>1</b>	<b>Questões e Respostas</b>	<b>3</b>
1.1	Etapa 1 . . . . .	3
	Questão 1 . . . . .	3
1.2	Etapa 2 . . . . .	12
	Questão 2 . . . . .	12
	Questão 3 . . . . .	12
	Questão 4 . . . . .	13
1.3	Etapa 3 . . . . .	13
	Questão 5 . . . . .	13
<b>2</b>	<b>Conclusões</b>	<b>14</b>

# 1 Questões e Respostas

## 1.1 Etapa 1

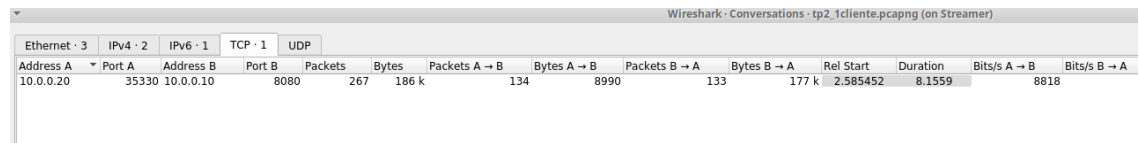
### Questão 1

Capture três pequenas amostras de tráfego no link de saída do servidor, respectivamente com 1 cliente (VLC), com 2 clientes (VLC e Firefox) e com 3 clientes (VLC, Firefox e ffmpeg). Identifique a taxa em bps necessária (usando o ffmpeg -i video1.mp4 e/ou o próprio wireshark), o encapsulamento usado e o número total de fluxos gerados. Comente a escalabilidade da solução. Ilustre com evidências da realização prática do exercício (ex: capturas de ecrã).

**R:** Nesta etapa foi feito *streaming* por HTTP apenas, sem adaptação do débito. Verificou-se, usando o comando referido no enunciado, que a taxa necessária para a transmissão do vídeo capturado é de 49 Kbps.

De seguida são acrescentados mais dois clientes (firefox e ffmpeg) e é estudado o impacto dos mesmos no tráfego da rede. Para tal, capturou-se uma pequena amostra de tráfego no link de saída do servidor para cada um dos seguintes casos:

- No caso de **um cliente (VLC)**, verifica-se que existe apenas um fluxo TCP.



Wireshark - Conversations - tp2_1cliente.pcapng (on Streamer)													
Ethernet · 3		IPv4 · 2		IPv6 · 1		TCP · 1		UDP					
Address A	Port A	Address B	Port B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration	Bits/s A → B	Bits/s B → A
10.0.0.20	35330	10.0.0.10	8080	267	186 k	134	8990	133	177 k	2.585452	8.1559		8818

Figura 1: 1 Cliente - Fluxo

Apply a display filter ... <Ctrl-/>					
No.	Time	Source	Destination	Protocol	Length Info
1	0.000000000	10.0.0.1	224.0.0.5	OSPF	78 Hello Packet
2	2.001441328	10.0.0.1	224.0.0.5	OSPF	78 Hello Packet
3	2.585452247	10.0.0.20	10.0.0.10	TCP	78 35330 → 8080 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1
4	2.585464500	10.0.0.10	10.0.0.20	TCP	74 8080 → 35330 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460
5	2.585475413	10.0.0.20	10.0.0.10	TCP	66 35330 → 8080 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=97429777
6	2.585530178	10.0.0.20	10.0.0.10	HTTP	200 GET / HTTP/1.1
7	2.585535284	10.0.0.10	10.0.0.20	TCP	66 8080 → 35330 [ACK] Seq=1 Ack=135 Win=65152 Len=0 TSval=307864
8	2.605732491	10.0.0.10	10.0.0.20	TCP	169 8080 → 35330 [PSH, ACK] Seq=1 Ack=135 Win=65152 Len=103 TSval=
9	2.605752411	10.0.0.20	10.0.0.10	TCP	66 35330 → 8080 [ACK] Seq=135 Ack=104 Win=64256 Len=0 TSval=9742
10	2.605768027	10.0.0.10	10.0.0.20	TCP	1514 8080 → 35330 [ACK] Seq=104 Ack=135 Win=65152 Len=1448 TSval=5
11	2.605768155	10.0.0.10	10.0.0.20	TCP	1514 8080 → 35330 [ACK] Seq=1552 Ack=135 Win=65152 Len=1448 TSval=
12	2.605768254	10.0.0.10	10.0.0.20	TCP	543 8080 → 35330 [PSH, ACK] Seq=3090 Ack=135 Win=65152 Len=477 TS
13	2.605776818	10.0.0.10	10.0.0.10	TCP	66 35330 → 8080 [ACK] Seq=135 Ack=1552 Win=64128 Len=0 TSval=974
14	2.605777529	10.0.0.20	10.0.0.10	TCP	66 35330 → 8080 [ACK] Seq=135 Ack=3090 Win=63488 Len=0 TSval=974
15	2.605778095	10.0.0.20	10.0.0.10	TCP	66 35330 → 8080 [ACK] Seq=135 Ack=3477 Win=63232 Len=0 TSval=974
16	2.605792604	10.0.0.10	10.0.0.20	TCP	1514 8080 → 35330 [ACK] Seq=3477 Ack=135 Win=65152 Len=1448 TSval=
17	2.605792737	10.0.0.10	10.0.0.20	TCP	1514 8080 → 35330 [ACK] Seq=4925 Ack=135 Win=65152 Len=1448 TSval=
18	2.605792837	10.0.0.10	10.0.0.20	TCP	1514 8080 → 35330 [ACK] Seq=6373 Ack=135 Win=65152 Len=1448 TSval=
19	2.605792937	10.0.0.10	10.0.0.20	TCP	1514 8080 → 35330 [ACK] Seq=7821 Ack=135 Win=65152 Len=1448 TSval=
20	2.605793026	10.0.0.10	10.0.0.20	TCP	1514 8080 → 35330 [PSH, ACK] Seq=9269 Ack=135 Win=65152 Len=1448 T
21	2.605804939	10.0.0.20	10.0.0.10	TCP	66 35330 → 8080 [ACK] Seq=135 Ack=4925 Win=62464 Len=0 TSval=974
22	2.605805609	10.0.0.20	10.0.0.10	TCP	66 35330 → 8080 [ACK] Seq=135 Ack=6373 Win=61824 Len=0 TSval=974
23	2.605806261	10.0.0.20	10.0.0.10	TCP	66 35330 → 8080 [ACK] Seq=135 Ack=7821 Win=61056 Len=0 TSval=974
24	2.605806758	10.0.0.20	10.0.0.10	TCP	66 35330 → 8080 [ACK] Seq=135 Ack=9269 Win=60288 Len=0 TSval=974
25	2.605807317	10.0.0.20	10.0.0.10	TCP	66 35330 → 8080 [ACK] Seq=135 Ack=10717 Win=59648 Len=0 TSval=97
26	2.605811226	10.0.0.10	10.0.0.20	TCP	1514 8080 → 35330 [ACK] Seq=10717 Ack=135 Win=65152 Len=1448 TSval=
27	2.605811331	10.0.0.10	10.0.0.20	TCP	1378 8080 → 35330 [PSH, ACK] Seq=12165 Ack=135 Win=65152 Len=1312
28	2.605815710	10.0.0.20	10.0.0.10	TCP	66 35330 → 8080 [ACK] Seq=135 Ack=12165 Win=58880 Len=0 TSval=97
29	2.605816309	10.0.0.20	10.0.0.10	TCP	66 35330 → 8080 [ACK] Seq=135 Ack=13477 Win=58240 Len=0 TSval=97
30	2.605823951	10.0.0.10	10.0.0.20	TCP	1514 8080 → 35330 [ACK] Seq=13477 Ack=135 Win=65152 Len=1448 TSval=
31	2.605824048	10.0.0.10	10.0.0.20	TCP	163 8080 → 35330 [PSH, ACK] Seq=14925 Ack=135 Win=65152 Len=97 TS
32	2.605828578	10.0.0.20	10.0.0.10	TCP	66 35330 → 8080 [ACK] Seq=135 Ack=14925 Win=57472 Len=0 TSval=97
33	2.605829191	10.0.0.20	10.0.0.10	TCP	66 35330 → 8080 [ACK] Seq=135 Ack=15022 Win=57472 Len=0 TSval=97
34	2.849562299	10.0.0.10	10.0.0.20	TCP	1514 8080 → 35330 [ACK] Seq=15022 Ack=135 Win=65152 Len=1448 TSval=
35	2.849562722	10.0.0.10	10.0.0.20	TCP	1514 8080 → 35330 [ACK] Seq=16478 Ack=135 Win=65152 Len=1448 TSval=
36	2.849562840	10.0.0.10	10.0.0.20	TCP	1514 8080 → 35330 [ACK] Seq=17918 Ack=135 Win=65152 Len=1448 TSval=
37	2.849562930	10.0.0.10	10.0.0.20	TCP	1514 8080 → 35330 [ACK] Seq=19366 Ack=135 Win=65152 Len=1448 TSval=
38	2.849563020	10.0.0.10	10.0.0.20	TCP	1514 8080 → 35330 [PSH, ACK] Seq=20814 Ack=135 Win=65152 Len=1448
39	2.849591362	10.0.0.20	10.0.0.10	TCP	66 35330 → 8080 [ACK] Seq=135 Ack=16470 Win=64128 Len=0 TSval=97
40	2.849592530	10.0.0.20	10.0.0.10	TCP	66 35330 → 8080 [ACK] Seq=135 Ack=17918 Win=63488 Len=0 TSval=97
41	2.849593116	10.0.0.20	10.0.0.10	TCP	66 35330 → 8080 [ACK] Seq=135 Ack=19366 Win=62848 Len=0 TSval=97
42	2.849593668	10.0.0.20	10.0.0.10	TCP	66 35330 → 8080 [ACK] Seq=135 Ack=20814 Win=62080 Len=0 TSval=97
43	2.849594225	10.0.0.10	10.0.0.10	TCP	66 35330 → 8080 [ACK] Seq=135 Ack=22262 Win=61312 Len=0 TSval=97
44	2.849600237	10.0.0.10	10.0.0.20	TCP	1514 8080 → 35330 [ACK] Seq=22262 Ack=135 Win=65152 Len=1448 TSval=
45	2.849600363	10.0.0.10	10.0.0.20	TCP	1378 8080 → 35330 [PSH, ACK] Seq=23710 Ack=135 Win=65152 Len=1312

Figura 2: 1 Cliente - VLC

No.	Time	Source	Destination	Protocol	Length	Info
234	8.105424036	10.0.0.20	10.0.0.10	TCP	66	35330 → 8080 [ACK] Seq=135 Ack=144374 Win=63488 Len=0 TSval=9...
235	8.105424816	10.0.0.20	10.0.0.10	TCP	66	35330 → 8080 [ACK] Seq=135 Ack=145822 Win=62848 Len=0 TSval=9...
236	8.105425546	10.0.0.20	10.0.0.10	TCP	66	35330 → 8080 [ACK] Seq=135 Ack=147270 Win=62080 Len=0 TSval=9...
237	8.105426288	10.0.0.20	10.0.0.10	TCP	66	35330 → 8080 [ACK] Seq=135 Ack=148718 Win=61312 Len=0 TSval=9...
238	8.105434373	10.0.0.10	10.0.0.20	TCP	1514	8080 → 35330 [ACK] Seq=148718 Ack=135 Win=65152 Len=1448 TSva...
239	8.105434537	10.0.0.10	10.0.0.20	TCP	1378	8080 → 35330 [PSH, ACK] Seq=150166 Ack=135 Win=65152 Len=1312...
240	8.105441076	10.0.0.20	10.0.0.10	TCP	66	35330 → 8080 [ACK] Seq=135 Ack=150166 Win=60672 Len=0 TSval=9...
241	8.105442058	10.0.0.20	10.0.0.10	TCP	66	35330 → 8080 [ACK] Seq=135 Ack=151478 Win=59984 Len=0 TSval=9...
242	8.106796417	10.0.0.10	10.0.0.20	TCP	1514	8080 → 35330 [ACK] Seq=151478 Ack=135 Win=65152 Len=1448 TSva...
243	8.106796835	10.0.0.10	10.0.0.20	TCP	1514	8080 → 35330 [ACK] Seq=152926 Ack=135 Win=65152 Len=1448 TSva...
244	8.106796935	10.0.0.10	10.0.0.20	TCP	1514	8080 → 35330 [ACK] Seq=154374 Ack=135 Win=65152 Len=1448 TSva...
245	8.106797024	10.0.0.10	10.0.0.20	TCP	1514	8080 → 35330 [ACK] Seq=155822 Ack=135 Win=65152 Len=1448 TSva...
246	8.106797116	10.0.0.10	10.0.0.20	TCP	1514	8080 → 35330 [ACK] Seq=157270 Ack=135 Win=65152 Len=1448 TSva...
247	8.106797286	10.0.0.10	10.0.0.20	TCP	71	8080 → 35330 [PSH, ACK] Seq=158718 Ack=135 Win=65152 Len=5 TS...
248	8.106821628	10.0.0.20	10.0.0.10	TCP	66	35330 → 8080 [ACK] Seq=135 Ack=152926 Win=64128 Len=0 TSval=9...
249	8.106822595	10.0.0.20	10.0.0.10	TCP	66	35330 → 8080 [ACK] Seq=135 Ack=154374 Win=63488 Len=0 TSval=9...
250	8.106823194	10.0.0.20	10.0.0.10	TCP	66	35330 → 8080 [ACK] Seq=135 Ack=155822 Win=62848 Len=0 TSval=9...
251	8.106823745	10.0.0.20	10.0.0.10	TCP	66	35330 → 8080 [ACK] Seq=135 Ack=157270 Win=62080 Len=0 TSval=9...
252	8.106824296	10.0.0.20	10.0.0.10	TCP	66	35330 → 8080 [ACK] Seq=135 Ack=158718 Win=61312 Len=0 TSval=9...
253	8.106824856	10.0.0.20	10.0.0.10	TCP	66	35330 → 8080 [ACK] Seq=135 Ack=158723 Win=61312 Len=0 TSval=9...
254	8.613139777	10.0.0.10	10.0.0.20	TCP	1514	8080 → 35330 [ACK] Seq=158723 Ack=135 Win=65152 Len=1448 TSva...
255	8.613140159	10.0.0.10	10.0.0.20	TCP	1514	8080 → 35330 [ACK] Seq=160171 Ack=135 Win=65152 Len=1448 TSva...
256	8.613140257	10.0.0.10	10.0.0.20	TCP	1514	8080 → 35330 [ACK] Seq=161619 Ack=135 Win=65152 Len=1448 TSva...
257	8.613140345	10.0.0.10	10.0.0.20	TCP	120	8080 → 35330 [PSH, ACK] Seq=163067 Ack=135 Win=65152 Len=54 T...
258	8.613160850	10.0.0.20	10.0.0.10	TCP	66	35330 → 8080 [ACK] Seq=135 Ack=160171 Win=64128 Len=0 TSval=9...
259	8.613160940	10.0.0.20	10.0.0.10	TCP	66	35330 → 8080 [ACK] Seq=135 Ack=161619 Win=63488 Len=0 TSval=9...
260	8.6131609718	10.0.0.20	10.0.0.10	TCP	66	35330 → 8080 [ACK] Seq=135 Ack=163067 Win=62848 Len=0 TSval=9...
261	8.613170264	10.0.0.20	10.0.0.10	TCP	66	35330 → 8080 [ACK] Seq=135 Ack=163121 Win=62848 Len=0 TSval=9...
262	8.854956225	10.0.0.10	10.0.0.20	TCP	1514	8080 → 35330 [ACK] Seq=163121 Ack=135 Win=65152 Len=1448 TSva...
263	8.854956596	10.0.0.10	10.0.0.20	TCP	1514	8080 → 35330 [ACK] Seq=164569 Ack=135 Win=65152 Len=1448 TSva...
264	8.854956695	10.0.0.10	10.0.0.20	TCP	1514	8080 → 35330 [ACK] Seq=166017 Ack=135 Win=65152 Len=1448 TSva...
265	8.854956787	10.0.0.10	10.0.0.20	TCP	1121	8080 → 35330 [PSH, ACK] Seq=167465 Ack=135 Win=65152 Len=1055...
266	8.854983439	10.0.0.20	10.0.0.10	TCP	66	35330 → 8080 [ACK] Seq=135 Ack=164569 Win=67072 Len=0 TSval=9...
267	8.854984622	10.0.0.20	10.0.0.10	TCP	66	35330 → 8080 [ACK] Seq=135 Ack=166017 Win=60888 Len=0 TSval=9...
268	8.854985193	10.0.0.20	10.0.0.10	TCP	66	35330 → 8080 [ACK] Seq=135 Ack=167465 Win=72832 Len=0 TSval=9...
269	8.854985778	10.0.0.20	10.0.0.10	TCP	66	35330 → 8080 [ACK] Seq=135 Ack=168520 Win=75776 Len=0 TSval=9...
270	9.832755318	10.0.0.10	10.0.0.20	HTTP	66	HTTP/1.0 200 OK
271	9.873175969	10.0.0.20	10.0.0.10	TCP	66	35330 → 8080 [ACK] Seq=135 Ack=168521 Win=75776 Len=0 TSval=9...
272	10.005173848	10.0.0.1	224.0.0.5	OSPF	78	Hello Packet
273	10.741358421	10.0.0.20	10.0.0.10	TCP	66	35330 → 8080 [FIN, ACK] Seq=135 Ack=168521 Win=75776 Len=0 TS...
274	10.741370990	10.0.0.10	10.0.0.20	TCP	66	8080 → 35330 [ACK] Seq=168521 Ack=136 Win=65152 Len=0 TSval=3...
275	12.006169606	10.0.0.1	224.0.0.5	OSPF	78	Hello Packet
276	13.932080956	fe80::200:ff:feaa:2	ff02::5	OSPF	90	Hello Packet
277	14.006957396	10.0.0.1	224.0.0.5	OSPF	78	Hello Packet
278	16.008526242	10.0.0.1	224.0.0.5	OSPF	78	Hello Packet

Figura 3: 1 Cliente - VLC

- No caso de dois clientes (VLC e Firefox), verifica-se que existem dois fluxos TCP

Wireshark - Conversations - tp2_2clientes.pcapng (on Streamer)													
Ethernet · 3			IPv4 · 3		IPv6 · 1		TCP · 2		UDP				
Address A	Port A	Address B	Port B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration	Bits/s A → B	Bits/s B → A
10.0.2.20	45580	10.0.0.10	8080	387	276 k	194	12 k	193	263 k	13.367675	9.4027	11 k	
10.0.2.21	51386	10.0.0.10	8080	591	432 k	293	19 k	298	412 k	8.861002	13.8625	11 k	

Figura 4: 2 Clientes - Fluxo

Apply a display filter ... <Ctrl-F>					
No.	Time	Source	Destination	Protocol	Length Info
10	16.064500325	10.0.0.1	224.0.0.5	OSPF	78 Hello Packet
11	18.065824483	10.0.0.1	224.0.0.5	OSPF	78 Hello Packet
12	18.022757236	fe80::200:ff:feaa:2	ff02::5	OSPF	90 Hello Packet
13	20.066481381	10.0.0.1	224.0.0.5	OSPF	78 Hello Packet
14	22.067821704	10.0.0.1	224.0.0.5	OSPF	78 Hello Packet
15	24.068325705	10.0.0.1	224.0.0.5	OSPF	78 Hello Packet
16	26.069039987	10.0.0.1	224.0.0.5	OSPF	78 Hello Packet
17	27.678769163	10.0.0.20	10.0.0.10	TCP	78 52838 → 8080 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1
18	27.678763061	10.0.0.10	10.0.0.20	TCP	74 8080 → 52838 [SYN, ACK] Seq=9 Ack=1 Win=65152 Len=0 MSS=1460
19	27.678795494	10.0.0.20	10.0.0.10	TCP	66 52838 → 8080 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=14750731
20	27.678837309	10.0.0.20	10.0.0.10	HTTP	200 GET / HTTP/1.1
21	27.678849570	10.0.0.10	10.0.0.20	TCP	66 8080 → 52838 [ACK] Seq=1 Ack=135 Win=65152 Len=0 TSval=313608
22	27.699717676	10.0.0.10	10.0.0.20	TCP	169 8080 → 52838 [PSH, ACK] Seq=1 Ack=135 Win=65152 Len=103 TSval=
23	27.699737832	10.0.0.20	10.0.0.10	TCP	66 52838 → 8080 [ACK] Seq=135 Ack=104 Win=64256 Len=0 TSval=1475
24	27.699749358	10.0.0.10	10.0.0.20	TCP	1514 8080 → 52838 [ACK] Seq=104 Ack=135 Win=65152 Len=1448 TSval=3
25	27.699749482	10.0.0.10	10.0.0.20	TCP	1514 8080 → 52838 [ACK] Seq=1552 Ack=135 Win=65152 Len=1448 TSval=
26	27.699749582	10.0.0.10	10.0.0.20	TCP	543 8080 → 52838 [PSH, ACK] Seq=3000 Ack=135 Win=65152 Len=477 TS
27	27.699757082	10.0.0.20	10.0.0.10	TCP	66 52838 → 8080 [ACK] Seq=135 Ack=1552 Win=64128 Len=0 TSval=147
28	27.699757700	10.0.0.20	10.0.0.10	TCP	66 52838 → 8080 [ACK] Seq=135 Ack=3000 Win=63488 Len=0 TSval=147
29	27.699758198	10.0.0.20	10.0.0.10	TCP	66 52838 → 8080 [ACK] Seq=135 Ack=3477 Win=63232 Len=0 TSval=147
30	27.699766419	10.0.0.10	10.0.0.20	TCP	1514 8080 → 52838 [ACK] Seq=3477 Ack=135 Win=65152 Len=1448 TSval=
31	27.699766540	10.0.0.10	10.0.0.20	TCP	1514 8080 → 52838 [ACK] Seq=4925 Ack=135 Win=65152 Len=1448 TSval=
32	27.699766640	10.0.0.10	10.0.0.20	TCP	1514 8080 → 52838 [ACK] Seq=6373 Ack=135 Win=65152 Len=1448 TSval=
33	27.699766734	10.0.0.10	10.0.0.20	TCP	348 8080 → 52838 [PSH, ACK] Seq=7821 Ack=135 Win=65152 Len=282 TS
34	27.699774490	10.0.0.20	10.0.0.10	TCP	66 52838 → 8080 [ACK] Seq=135 Ack=4925 Win=62464 Len=0 TSval=147
35	27.699775034	10.0.0.20	10.0.0.10	TCP	66 52838 → 8080 [ACK] Seq=135 Ack=6373 Win=61824 Len=0 TSval=147
36	27.699775533	10.0.0.20	10.0.0.10	TCP	66 52838 → 8080 [ACK] Seq=135 Ack=7821 Win=61056 Len=0 TSval=147
37	27.699776013	10.0.0.20	10.0.0.10	TCP	66 52838 → 8080 [ACK] Seq=135 Ack=8103 Win=60928 Len=0 TSval=147
38	27.991781891	fe80::200:ff:feaa:2	ff02::5	OSPF	90 Hello Packet
39	28.062934854	10.0.0.10	10.0.0.20	TCP	1514 8080 → 52838 [ACK] Seq=8103 Ack=135 Win=65152 Len=1448 TSval=
40	28.062935191	10.0.0.10	10.0.0.20	TCP	1514 8080 → 52838 [ACK] Seq=9551 Ack=135 Win=65152 Len=1448 TSval=
41	28.062935334	10.0.0.10	10.0.0.20	TCP	1514 8080 → 52838 [ACK] Seq=10999 Ack=135 Win=65152 Len=1448 TSval=
42	28.062935467	10.0.0.10	10.0.0.20	TCP	118 8080 → 52838 [PSH, ACK] Seq=12447 Ack=135 Win=65152 Len=52 TS
43	28.062959324	10.0.0.20	10.0.0.10	TCP	66 52838 → 8080 [ACK] Seq=135 Ack=9551 Win=64128 Len=0 TSval=147
44	28.062960310	10.0.0.20	10.0.0.10	TCP	66 52838 → 8080 [ACK] Seq=135 Ack=10999 Win=63488 Len=0 TSval=14
45	28.062961035	10.0.0.20	10.0.0.10	TCP	66 52838 → 8080 [ACK] Seq=135 Ack=12447 Win=62848 Len=0 TSval=14
46	28.062961728	10.0.0.20	10.0.0.10	TCP	66 52838 → 8080 [ACK] Seq=135 Ack=12499 Win=62848 Len=0 TSval=14
47	28.069841994	10.0.0.1	224.0.0.5	OSPF	78 Hello Packet
48	28.754561119	10.0.0.10	10.0.0.20	TCP	1514 8080 → 52838 [ACK] Seq=12499 Ack=135 Win=65152 Len=1448 TSval=
49	28.754561514	10.0.0.10	10.0.0.20	TCP	1514 8080 → 52838 [ACK] Seq=13947 Ack=135 Win=65152 Len=1448 TSval=
50	28.754561655	10.0.0.10	10.0.0.20	TCP	1323 8080 → 52838 [PSH, ACK] Seq=15395 Ack=135 Win=65152 Len=1257
51	28.754565457	10.0.0.20	10.0.0.10	TCP	66 52838 → 8080 [ACK] Seq=135 Ack=13947 Win=64128 Len=0 TSval=14
52	28.754565468	10.0.0.20	10.0.0.10	TCP	66 52838 → 8080 [ACK] Seq=135 Ack=15395 Win=63488 Len=0 TSval=14
Frame 1: 78 bytes on wire (624 bits), 78 bytes captured (624 bits) on interface veth8.0.5f, id 0					
0000	01 00 5e 00 00 05 00 00	00 aa 00 02 08 00 45 c0	--A-----E-		
0010	00 40 10 ca 00 00 01 59	05 d5 0a 00 00 01 e0 00	0-----Y-----		
0020	00 05 02 01 00 2c 0a 00	00 01 00 00 00 00 e8 c6	-----		
0030	00 00 00 00 00 00 00 00	00 00 ff ff 00 00 00 02	-----		
0040	02 01 00 00 00 06 0a 00	00 01 00 00 00 00 00	-----		

Figura 5: 2 Clientes - VLC e Firefox

No.	Time	Source	Destination	Protocol	Length	Info
55	29.259363949	10.0.0.10	10.0.0.20	TCP	1514	8080 → 52838 [ACK] Seq=18100 Ack=135 Win=65152 Len=1448 TSval=...
56	29.259364047	10.0.0.10	10.0.0.20	TCP	1514	8080 → 52838 [ACK] Seq=18548 Ack=135 Win=65152 Len=1448 TSval=...
57	29.259364137	10.0.0.10	10.0.0.20	TCP	1514	8080 → 52838 [ACK] Seq=20996 Ack=135 Win=65152 Len=1448 TSval=...
58	29.259364227	10.0.0.10	10.0.0.20	TCP	1514	8080 → 52838 [PSH, ACK] Seq=22444 Ack=135 Win=65152 Len=1448 TSval=...
59	29.259398707	10.0.0.20	10.0.0.10	TCP	66	52838 → 8080 [ACK] Seq=135 Ack=18100 Win=64128 Len=0 TSval=14...
60	29.259399578	10.0.0.20	10.0.0.10	TCP	66	52838 → 8080 [ACK] Seq=135 Ack=19548 Win=63488 Len=0 TSval=14...
61	29.259400066	10.0.0.20	10.0.0.10	TCP	66	52838 → 8080 [ACK] Seq=135 Ack=20996 Win=62948 Len=0 TSval=14...
62	29.259400591	10.0.0.20	10.0.0.10	TCP	66	52838 → 8080 [ACK] Seq=135 Ack=22444 Win=62080 Len=0 TSval=14...
63	29.259401074	10.0.0.20	10.0.0.10	TCP	66	52838 → 8080 [ACK] Seq=135 Ack=23892 Win=61312 Len=0 TSval=14...
64	29.259423123	10.0.0.10	10.0.0.20	TCP	1514	8080 → 52838 [ACK] Seq=23892 Ack=135 Win=65152 Len=1448 TSval=...
65	29.259423247	10.0.0.10	10.0.0.20	TCP	408	8080 → 52838 [PSH, ACK] Seq=25340 Ack=135 Win=65152 Len=342 T...
66	29.259427548	10.0.0.20	10.0.0.10	TCP	66	52838 → 8080 [ACK] Seq=135 Ack=25340 Win=60672 Len=0 TSval=14...
67	29.259427917	10.0.0.20	10.0.0.10	TCP	66	52838 → 8080 [ACK] Seq=135 Ack=25682 Win=60416 Len=0 TSval=14...
68	30.010378736	10.0.0.1	224.0.0.5	OSPF	78	Hello Packet
69	31.1090612498	10.0.0.20	10.0.0.10	TCP	74	36840 → 8080 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SACK PERM=...
70	31.1090627848	10.0.0.10	10.0.0.20	TCP	74	8080 → 36840 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SACK PERM=...
71	31.1090652039	10.0.0.20	10.0.0.10	TCP	66	36840 → 8080 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=24780115...
72	31.109085806	10.0.0.10	10.0.0.10	HTTP	360	GET / HTTP/1.1
73	31.109812441	10.0.0.10	10.0.0.20	TCP	66	8080 → 36840 [ACK] Seq=1 Ack=295 Win=64896 Len=0 TSval=307022...
74	31.130184031	10.0.0.10	10.0.0.20	TCP	169	8080 → 36840 [PSH, ACK] Seq=1 Ack=295 Win=64896 Len=103 TSval=...
75	31.130222309	10.0.0.20	10.0.0.10	TCP	66	36840 → 8080 [ACK] Seq=295 Ack=104 Win=64256 Len=0 TSval=2478...
76	31.130238198	10.0.0.10	10.0.0.20	TCP	1514	8080 → 36840 [ACK] Seq=104 Ack=295 Win=64896 Len=1448 TSval=3...
77	31.130238493	10.0.0.10	10.0.0.20	TCP	1514	8080 → 36840 [ACK] Seq=1552 Ack=295 Win=64896 Len=1448 TSval=3...
78	31.130238547	10.0.0.10	10.0.0.20	TCP	543	8080 → 36840 [PSH, ACK] Seq=3090 Ack=295 Win=64896 Len=477 TS...
79	31.130266903	10.0.0.20	10.0.0.10	TCP	66	36840 → 8080 [ACK] Seq=295 Ack=1552 Win=64128 Len=0 TSval=247...
80	31.130267876	10.0.0.20	10.0.0.10	TCP	66	36840 → 8080 [ACK] Seq=295 Ack=3090 Win=63488 Len=0 TSval=247...
81	31.130268724	10.0.0.10	10.0.0.10	TCP	66	36840 → 8080 [ACK] Seq=295 Ack=3477 Win=63232 Len=0 TSval=247...
82	31.130283544	10.0.0.10	10.0.0.20	TCP	1514	8080 → 36840 [ACK] Seq=3477 Ack=295 Win=64896 Len=1448 TSval=...
83	31.130283736	10.0.0.10	10.0.0.20	TCP	1514	8080 → 36840 [ACK] Seq=4925 Ack=295 Win=64896 Len=1448 TSval=...
84	31.130283907	10.0.0.10	10.0.0.20	TCP	1514	8080 → 36840 [ACK] Seq=6373 Ack=295 Win=64896 Len=1448 TSval=...
85	31.130284082	10.0.0.10	10.0.0.20	TCP	1514	8080 → 36840 [ACK] Seq=7821 Ack=295 Win=64896 Len=1448 TSval=...
86	31.130284231	10.0.0.10	10.0.0.20	TCP	1514	8080 → 36840 [PSH, ACK] Seq=9269 Ack=295 Win=64896 Len=1448 T...
87	31.130328951	10.0.0.20	10.0.0.10	TCP	66	36840 → 8080 [ACK] Seq=295 Ack=4925 Win=62464 Len=0 TSval=247...
88	31.130329892	10.0.0.20	10.0.0.10	TCP	66	36840 → 8080 [ACK] Seq=295 Ack=6373 Win=61824 Len=0 TSval=247...
89	31.130330814	10.0.0.20	10.0.0.10	TCP	66	36840 → 8080 [ACK] Seq=295 Ack=7821 Win=61056 Len=0 TSval=247...
90	31.130331667	10.0.0.20	10.0.0.10	TCP	66	36840 → 8080 [ACK] Seq=295 Ack=9269 Win=60288 Len=0 TSval=247...
91	31.130332501	10.0.0.20	10.0.0.10	TCP	66	36840 → 8080 [ACK] Seq=295 Ack=10717 Win=59648 Len=0 TSval=24...
92	31.130338031	10.0.0.10	10.0.0.20	TCP	1514	8080 → 36840 [ACK] Seq=10717 Ack=295 Win=64896 Len=1448 TSval=...
93	31.130338219	10.0.0.10	10.0.0.20	TCP	408	8080 → 36840 [PSH, ACK] Seq=12165 Ack=295 Win=64896 Len=342 T...
94	31.130356358	10.0.0.20	10.0.0.10	TCP	66	36840 → 8080 [ACK] Seq=295 Ack=12165 Win=58080 Len=0 TSval=24...
95	31.130357280	10.0.0.20	10.0.0.10	TCP	66	36840 → 8080 [ACK] Seq=295 Ack=12507 Win=58752 Len=0 TSval=24...
96	32.011555778	10.0.0.1	224.0.0.5	OSPF	78	Hello Packet
97	32.513303204	10.0.0.10	10.0.0.20	TCP	1514	8080 → 52838 [ACK] Seq=25682 Ack=135 Win=65152 Len=1448 TSval=...
Frame 1: 78 bytes on wire (624 bits), 78 bytes captured (624 bits) on interface veth8.0.5f, id 0						
0000	01 00 5e 00 00 05 00 00 00 aa 00 02 08 00 45 c0	.....E				
0010	00 40 18 ca 00 00 01 59 b5 d5 0a 00 00 01 e0 00	.....Y				
0020	00 05 02 01 00 2c 0a 00 00 01 00 00 00 e8 c6	.....				
0030	00 00 00 00 00 00 00 00 00 ff ff ff 00 00 02	.....				
0040	02 01 00 00 00 06 0a 00 00 01 00 00 00 00	.....				

Figura 6: 2 Clientes - VLC e Firefox

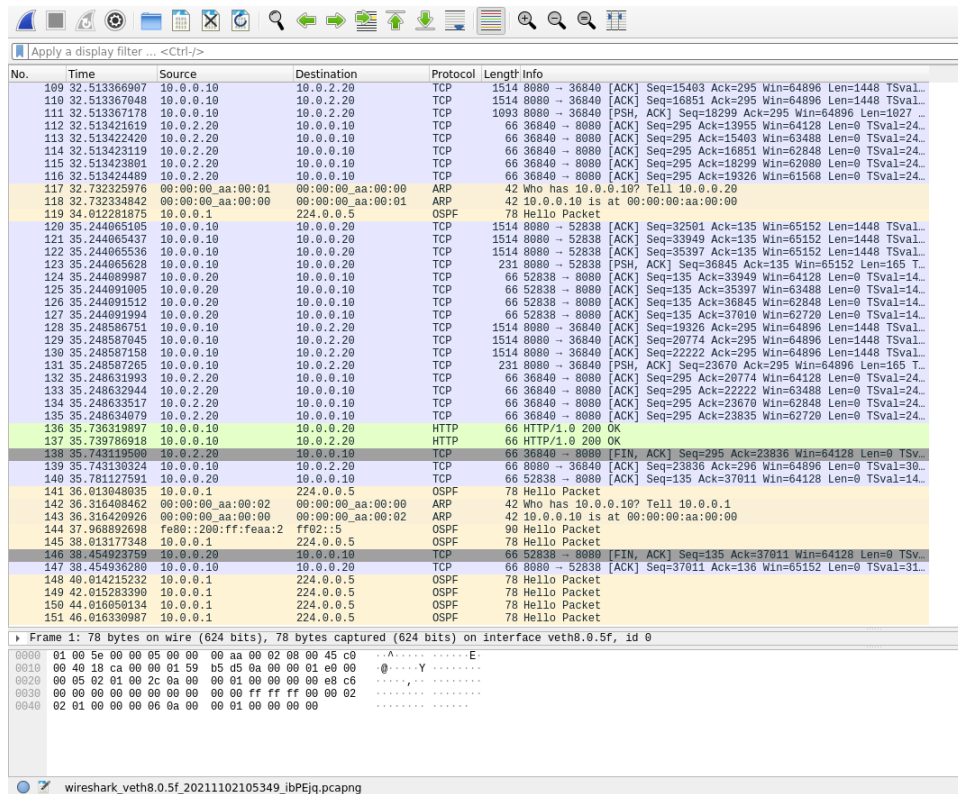


Figura 7: 2 Clientes - VLC e Firefox

- No caso de três clientes (VLC, Firefox e ffmpeg), é verificada a existência de três fluxos TCP

Ethernet · 4		IPv4 · 4		IPv6 · 1		TCP · 3		UDP						
Address A	Port A	Address B		Port B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration	Bits/s A → B	Bits/s B → A
10.0.0.20	37452	10.0.0.10		8080	3,034	2255 k	1,497	98 k	1,537	2156 k	10.315036	86.8082	9118	
10.0.2.20	47690	10.0.0.10		8080	2,704	2013 k	1,332	88 k	1,372	1925 k	17.341070	79.7137	8840	
10.0.2.21	53500	10.0.0.10		8080	2,609	1942 k	1,287	85 k	1,322	1857 k	21.756979	75.2572	9061	

Figura 8: 3 Clientes - Fluxo

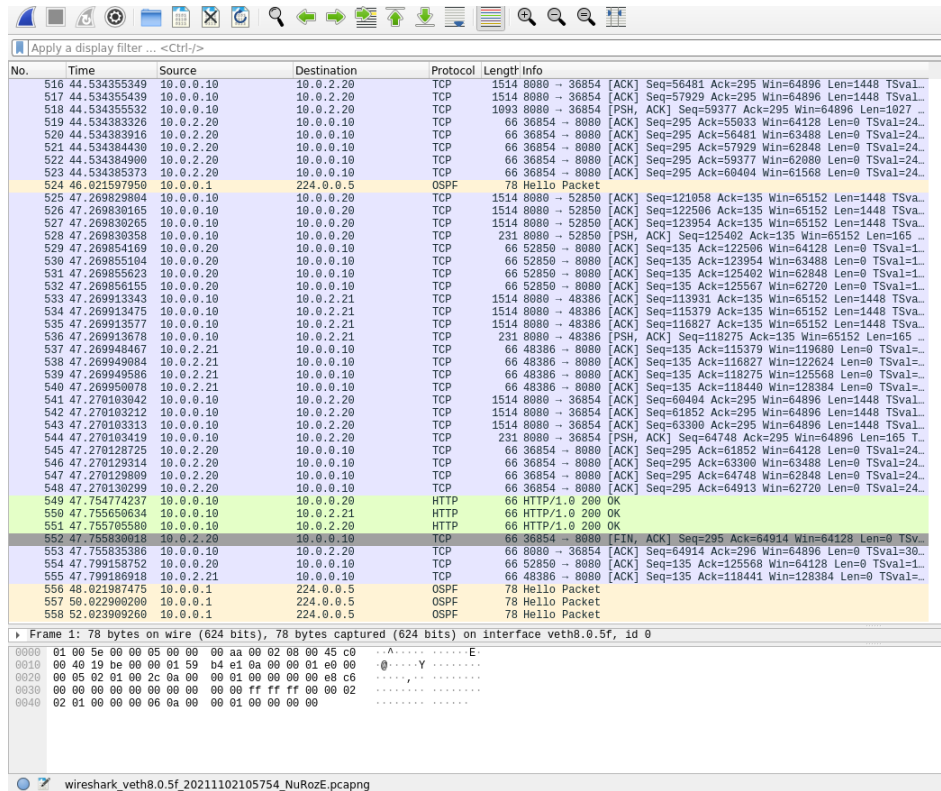


Apply a display filter ... <Ctrl-F>					
No.	Time	Source	Destination	Protocol	Length Info
25	31.178166577	10.0.0.10	10.0.0.20	TCP	74 8080 → 52850 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460
26	31.178175637	10.0.0.20	10.0.0.10	TCP	66 52850 → 8080 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=14759207...
27	31.178220053	10.0.0.20	10.0.0.10	HTTP	200 GET / HTTP/1.1
28	31.178222856	10.0.0.10	10.0.0.20	TCP	66 8080 → 52850 [ACK] Seq=1 Ack=135 Win=65152 Len=0 TSval=313633...
29	31.198470773	10.0.0.10	10.0.0.20	TCP	169 8080 → 52850 [PSH, ACK] Seq=1 Ack=135 Win=65152 Len=103 TSval=...
30	31.198490354	10.0.0.20	10.0.0.10	TCP	66 52850 → 8080 [ACK] Seq=135 Ack=104 Win=64256 Len=0 TSval=1475...
31	31.198503171	10.0.0.10	10.0.0.20	TCP	1514 8080 → 52850 [ACK] Seq=104 Ack=135 Win=65152 Len=1448 TSval=3...
32	31.198503297	10.0.0.10	10.0.0.20	TCP	1514 8080 → 52850 [ACK] Seq=1552 Ack=135 Win=65152 Len=1448 TSval=...
33	31.198503397	10.0.0.10	10.0.0.20	TCP	545 8080 → 52850 [PSH, ACK] Seq=3000 Ack=135 Win=65152 Len=477 TS...
34	31.198510900	10.0.0.20	10.0.0.10	TCP	66 52850 → 8080 [ACK] Seq=135 Ack=1552 Win=64120 Len=0 TSval=147...
35	31.198511573	10.0.0.20	10.0.0.10	TCP	66 52850 → 8080 [ACK] Seq=135 Ack=3000 Win=63488 Len=0 TSval=147...
36	31.198512081	10.0.0.20	10.0.0.10	TCP	66 52850 → 8080 [ACK] Seq=135 Ack=3477 Win=63232 Len=0 TSval=147...
37	31.198521661	10.0.0.10	10.0.0.20	TCP	1514 8080 → 52850 [ACK] Seq=3477 Ack=135 Win=65152 Len=1448 TSval=...
38	31.198521789	10.0.0.10	10.0.0.20	TCP	1514 8080 → 52850 [ACK] Seq=4925 Ack=135 Win=65152 Len=1448 TSval=...
39	31.198521893	10.0.0.10	10.0.0.20	TCP	1514 8080 → 52850 [ACK] Seq=6373 Ack=135 Win=65152 Len=1448 TSval=...
40	31.198521990	10.0.0.10	10.0.0.20	TCP	1514 8080 → 52850 [ACK] Seq=7821 Ack=135 Win=65152 Len=1448 TSval=...
41	31.198522081	10.0.0.10	10.0.0.20	TCP	1401 8080 → 52850 [PSH, ACK] Seq=9269 Ack=135 Win=65152 Len=1335 T...
42	31.198532175	10.0.0.20	10.0.0.10	TCP	66 52850 → 8080 [ACK] Seq=135 Ack=4925 Win=62464 Len=0 TSval=147...
43	31.198532717	10.0.0.20	10.0.0.10	TCP	66 52850 → 8080 [ACK] Seq=135 Ack=6373 Win=61924 Len=0 TSval=147...
44	31.198533215	10.0.0.20	10.0.0.10	TCP	66 52850 → 8080 [ACK] Seq=135 Ack=7821 Win=61056 Len=0 TSval=147...
45	31.198533696	10.0.0.20	10.0.0.10	TCP	66 52850 → 8080 [ACK] Seq=135 Ack=9269 Win=60288 Len=0 TSval=147...
46	31.198534179	10.0.0.10	10.0.0.10	TCP	66 52850 → 8080 [ACK] Seq=135 Ack=10604 Win=59648 Len=0 TSval=14...
47	31.1783136022	10.0.0.10	10.0.0.20	TCP	1514 8080 → 52850 [ACK] Seq=10604 Ack=135 Win=65152 Len=1448 TSval=...
48	31.1783136776	10.0.0.10	10.0.0.20	TCP	1514 8080 → 52850 [ACK] Seq=12052 Ack=135 Win=65152 Len=1448 TSval=...
49	31.1783136917	10.0.0.10	10.0.0.20	TCP	1514 8080 → 52850 [ACK] Seq=13500 Ack=135 Win=65152 Len=1448 TSval=...
50	31.1783137048	10.0.0.10	10.0.0.20	TCP	1514 8080 → 52850 [ACK] Seq=14948 Ack=135 Win=65152 Len=1448 TSval=...
51	31.1783137175	10.0.0.10	10.0.0.20	TCP	1514 8080 → 52850 [PSH, ACK] Seq=16396 Ack=135 Win=65152 Len=1448 ...
52	31.1783167884	10.0.0.10	10.0.0.10	TCP	66 52850 → 8080 [ACK] Seq=135 Ack=12652 Win=64128 Len=0 TSval=14...
53	31.1783168812	10.0.0.20	10.0.0.10	TCP	66 52850 → 8080 [ACK] Seq=135 Ack=13500 Win=63488 Len=0 TSval=14...
54	31.1783169532	10.0.0.20	10.0.0.10	TCP	66 52850 → 8080 [ACK] Seq=135 Ack=14948 Win=62848 Len=0 TSval=14...
55	31.1783170221	10.0.0.20	10.0.0.10	TCP	66 52850 → 8080 [ACK] Seq=135 Ack=16396 Win=62080 Len=0 TSval=14...
56	31.1783170906	10.0.0.20	10.0.0.10	TCP	66 52850 → 8080 [ACK] Seq=135 Ack=17844 Win=61312 Len=0 TSval=14...
57	31.1783176600	10.0.0.10	10.0.0.20	TCP	73 8080 → 52850 [PSH, ACK] Seq=17844 Ack=135 Win=65152 Len=7 TSv...
58	31.1783179950	10.0.0.20	10.0.0.10	TCP	66 52850 → 8080 [ACK] Seq=135 Ack=17851 Win=61312 Len=0 TSval=14...
59	32.013821890	10.0.0.1	224.0.0.5	OSPF	78 Hello Packet
60	33.285100923	10.0.2.21	10.0.0.10	TCP	74 48386 → 8080 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1...
61	33.285134829	10.0.0.10	10.0.2.21	TCP	74 8080 → 48386 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 ...
62	33.285153214	10.0.2.21	10.0.0.10	TCP	66 48386 → 8080 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=16713376...
63	33.285200977	10.0.2.21	10.0.0.10	HTTP	200 GET / HTTP/1.1
64	33.285205150	10.0.0.10	10.0.2.21	TCP	66 8080 → 48386 [ACK] Seq=1 Ack=135 Win=65152 Len=0 TSval=410382...
65	33.305435031	10.0.0.10	10.0.2.21	TCP	169 8080 → 48386 [PSH, ACK] Seq=1 Ack=135 Win=65152 Len=103 TSval=...
66	33.305469895	10.0.2.21	10.0.0.10	TCP	66 48386 → 8080 [ACK] Seq=135 Ack=104 Win=64256 Len=0 TSval=1671...
67	33.305483020	10.0.0.10	10.0.2.21	TCP	1514 8080 → 48386 [ACK] Seq=104 Ack=135 Win=65152 Len=1448 TSval=4...
Frame 1: 78 bytes on wire (624 bits), 78 bytes captured (624 bits) on interface veth8.0.5f, id 0					
0000	01 00 5e 00 00 05 00 00	00 aa 00 02 08 00 45 c0	..A.....E		
0010	00 40 19 be 00 00 01 59	d4 e1 0a 00 00 01 e0 00	@.....Y		
0020	00 05 02 01 00 7c 0a 00	00 01 00 00 00 00 c6	.....,.....		
0030	00 00 00 00 00 00 00 00	00 00 ff ff 00 00 02	.....		
0040	02 01 00 00 00 06 0a 00	00 01 00 00 00 00	.....		

Figura 9: 3 Clientes - VLC, Firefox e ffmpeg

No.	Time	Source	Destination	Protocol	Length	Info
235	37.023397568	10.0.0.21	10.0.0.10	TCP	66	48386 → 8080 [ACK] Seq=135 Ack=57004 Win=52480 Len=0 TSval=16...
236	37.289021526	10.0.0.10	10.0.0.20	TCP	1514	8080 → 52850 [ACK] Seq=64131 Ack=135 Win=65152 Len=1448 TSval=...
237	37.289021987	10.0.0.10	10.0.0.20	TCP	1514	8080 → 52850 [ACK] Seq=65579 Ack=135 Win=65152 Len=1448 TSval=...
238	37.289022148	10.0.0.10	10.0.0.20	TCP	1514	8080 → 52850 [ACK] Seq=67027 Ack=135 Win=65152 Len=1448 TSval=...
239	37.289022290	10.0.0.10	10.0.0.20	TCP	527	8080 → 52850 [PSH, ACK] Seq=68475 Ack=135 Win=65152 Len=461 T...
240	37.289051736	10.0.0.20	10.0.0.10	TCP	66	52850 → 8080 [ACK] Seq=135 Ack=65579 Win=64128 Len=0 TSval=14...
241	37.289053099	10.0.0.20	10.0.0.10	TCP	66	52850 → 8080 [ACK] Seq=135 Ack=67027 Win=63488 Len=0 TSval=14...
242	37.289053886	10.0.0.20	10.0.0.10	TCP	66	52850 → 8080 [ACK] Seq=135 Ack=68475 Win=62848 Len=0 TSval=14...
243	37.289054605	10.0.0.20	10.0.0.10	TCP	66	52850 → 8080 [ACK] Seq=135 Ack=68936 Win=62592 Len=0 TSval=14...
244	37.289068470	10.0.0.10	10.0.2.21	TCP	1514	8080 → 48386 [ACK] Seq=57004 Ack=135 Win=65152 Len=1448 TSval=...
245	37.289068666	10.0.0.10	10.0.2.21	TCP	1514	8080 → 48386 [ACK] Seq=58452 Ack=135 Win=65152 Len=1448 TSval=...
246	37.289068819	10.0.0.10	10.0.2.21	TCP	1514	8080 → 48386 [ACK] Seq=59900 Ack=135 Win=65152 Len=1448 TSval=...
247	37.289068977	10.0.0.10	10.0.2.21	TCP	527	8080 → 48386 [PSH, ACK] Seq=61348 Ack=135 Win=65152 Len=461 T...
248	37.289152539	10.0.2.21	10.0.0.10	TCP	66	48386 → 8080 [ACK] Seq=135 Ack=58452 Win=51840 Len=0 TSval=16...
249	37.289153457	10.0.2.21	10.0.0.10	TCP	66	48386 → 8080 [ACK] Seq=135 Ack=59900 Win=51072 Len=0 TSval=16...
250	37.289154203	10.0.2.21	10.0.0.10	TCP	66	48386 → 8080 [ACK] Seq=135 Ack=61348 Win=50304 Len=0 TSval=16...
251	37.289154926	10.0.2.21	10.0.0.10	TCP	66	48386 → 8080 [ACK] Seq=135 Ack=61809 Win=50176 Len=0 TSval=16...
252	37.290653367	10.0.2.20	10.0.0.10	TCP	74	8080 → 36854 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460...
253	37.290654964	10.0.0.10	10.0.2.20	TCP	74	8080 → 36854 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460...
254	37.290657306	10.0.2.20	10.0.0.10	TCP	66	36854 → 8080 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=24782617...
255	37.296690780	10.0.0.10	10.0.0.10	HTTP	360	GET / HTTP/1.1
256	37.296696705	10.0.0.10	10.0.2.20	TCP	66	8080 → 36854 [ACK] Seq=1 Ack=295 Win=64896 Len=0 TSval=307272...
257	37.317232861	10.0.0.10	10.0.2.20	TCP	169	8080 → 36854 [PSH, ACK] Seq=1 Ack=295 Win=64896 Len=103 TSval=...
258	37.317236877	10.0.2.20	10.0.0.10	TCP	66	36854 → 8080 [ACK] Seq=295 Ack=104 Win=64256 Len=0 TSval=2478...
259	37.317284537	10.0.0.10	10.0.2.20	TCP	1514	8080 → 36854 [ACK] Seq=104 Ack=295 Win=64896 Len=1448 TSval=3...
260	37.317284713	10.0.0.10	10.0.2.20	TCP	1514	8080 → 36854 [ACK] Seq=1552 Ack=295 Win=64896 Len=1448 TSval=...
261	37.317284858	10.0.0.10	10.0.2.20	TCP	543	8080 → 36854 [PSH, ACK] Seq=3000 Ack=295 Win=64896 Len=477 TS...
262	37.317309275	10.0.2.20	10.0.0.10	TCP	66	36854 → 8080 [ACK] Seq=295 Ack=1552 Win=64128 Len=0 TSval=247...
263	37.317310056	10.0.2.20	10.0.0.10	TCP	66	36854 → 8080 [ACK] Seq=295 Ack=3000 Win=63488 Len=0 TSval=247...
264	37.317310753	10.0.2.20	10.0.0.10	TCP	66	36854 → 8080 [ACK] Seq=295 Ack=3477 Win=63232 Len=0 TSval=247...
265	37.317321831	10.0.0.10	10.0.2.20	TCP	1514	8080 → 36854 [ACK] Seq=3477 Ack=295 Win=64896 Len=1448 TSval=...
266	37.317321997	10.0.0.10	10.0.2.20	TCP	1514	8080 → 36854 [ACK] Seq=4925 Ack=295 Win=64896 Len=1448 TSval=...
267	37.317322145	10.0.0.10	10.0.2.20	TCP	1514	8080 → 36854 [ACK] Seq=6373 Ack=295 Win=64896 Len=1448 TSval=...
268	37.317322286	10.0.0.10	10.0.2.20	TCP	527	8080 → 36854 [PSH, ACK] Seq=7821 Ack=295 Win=64896 Len=461 TS...
269	37.317350916	10.0.2.20	10.0.0.10	TCP	66	36854 → 8080 [ACK] Seq=295 Ack=4925 Win=62464 Len=0 TSval=247...
270	37.317351721	10.0.0.10	10.0.0.10	TCP	66	36854 → 8080 [ACK] Seq=295 Ack=6373 Win=61824 Len=0 TSval=247...
271	37.317352433	10.0.2.20	10.0.0.10	TCP	66	36854 → 8080 [ACK] Seq=295 Ack=7821 Win=61056 Len=0 TSval=247...
272	37.317353130	10.0.2.20	10.0.0.10	TCP	66	36854 → 8080 [ACK] Seq=295 Ack=8282 Win=60800 Len=0 TSval=247...
273	37.787383525	10.0.0.10	10.0.0.20	TCP	1514	8080 → 52850 [ACK] Seq=68936 Ack=135 Win=65152 Len=1448 TSval=...
274	37.787383870	10.0.0.10	10.0.0.20	TCP	1514	8080 → 52850 [ACK] Seq=70384 Ack=135 Win=65152 Len=1448 TSval=...
275	37.787384024	10.0.0.10	10.0.0.20	TCP	1514	8080 → 52850 [ACK] Seq=71832 Ack=135 Win=65152 Len=1448 TSval=...
276	37.787384163	10.0.0.10	10.0.0.20	TCP	183	8080 → 52850 [PSH, ACK] Seq=73280 Ack=135 Win=65152 Len=117 T...
277	37.787413829	10.0.0.20	10.0.0.10	TCP	66	52850 → 8080 [ACK] Seq=135 Ack=70384 Win=64128 Len=0 TSval=14...
Frame 1: 78 bytes on wire (624 bits), 78 bytes captured (624 bits) on interface veth8.0.5f, id 0						
0000	01 00 5e 00 00 05 00 00	00 aa 00 02 08 00 45 c0	--A-----E-			
0010	00 40 10 be 00 00 01 59	04 e1 0a 00 00 01 e0 00	-Y-----			
0020	00 05 02 01 00 2c 0a 00	00 01 00 00 00 e8 c6	-----			
0030	00 00 00 00 00 00 00 00	00 00 ff ff 00 00 02	-----			
0040	02 01 00 00 00 06 0a 00	00 01 00 00 00 00	-----			

Figura 10: 3 Clientes - VLC, Firefox e ffmpeg



No.	Time	Source	Destination	Protocol	Length	Info
516	44.534355349	10.0.0.10	10.0.2.20	TCP	1514	8080 → 36854 [ACK] Seq=56481 Ack=295 Win=64896 Len=1448 TSval=...
517	44.534355349	10.0.0.10	10.0.2.20	TCP	1514	8080 → 36854 [ACK] Seq=57929 Ack=295 Win=64896 Len=1448 TSval=...
518	44.534355352	10.0.0.10	10.0.2.20	TCP	1093	8080 → 36854 [PSH, ACK] Seq=59377 Ack=295 Win=64896 Len=1927 ...
519	44.534383326	10.0.2.20	10.0.0.10	TCP	66	36854 → 8080 [ACK] Seq=295 Ack=55033 Win=64128 Len=0 TSval=24...
520	44.534383916	10.0.2.20	10.0.0.10	TCP	66	36854 → 8080 [ACK] Seq=295 Ack=56481 Win=63488 Len=0 TSval=24...
521	44.534384430	10.0.2.20	10.0.0.10	TCP	66	36854 → 8080 [ACK] Seq=295 Ack=57929 Win=62848 Len=0 TSval=24...
522	44.534384900	10.0.2.20	10.0.0.10	TCP	66	36854 → 8080 [ACK] Seq=295 Ack=59377 Win=62800 Len=0 TSval=24...
523	44.534385373	10.0.2.20	10.0.0.10	TCP	66	36854 → 8080 [ACK] Seq=295 Ack=60404 Win=61568 Len=0 TSval=24...
524	46.021597950	10.0.0.1	224.0.0.5	OSPF	78	Hello Packet
525	47.269829804	10.0.0.10	10.0.0.20	TCP	1514	8080 → 52850 [ACK] Seq=121058 Ack=135 Win=65152 Len=1448 TSva...
526	47.269830165	10.0.0.10	10.0.0.20	TCP	1514	8080 → 52850 [ACK] Seq=122506 Ack=135 Win=65152 Len=1448 TSva...
527	47.269830265	10.0.0.10	10.0.0.20	TCP	1514	8080 → 52850 [ACK] Seq=123954 Ack=135 Win=65152 Len=1448 TSva...
528	47.269830358	10.0.0.10	10.0.0.20	TCP	231	8080 → 52850 [PSH, ACK] Seq=125402 Ack=135 Win=65152 Len=165 ...
529	47.269854169	10.0.0.20	10.0.0.10	TCP	66	52850 → 8080 [ACK] Seq=135 Ack=122506 Win=64128 Len=0 TSval=1...
530	47.269855104	10.0.0.20	10.0.0.10	TCP	66	52850 → 8080 [ACK] Seq=135 Ack=123954 Win=63488 Len=0 TSval=1...
531	47.269855623	10.0.0.20	10.0.0.10	TCP	66	52850 → 8080 [ACK] Seq=135 Ack=125402 Win=62848 Len=0 TSval=1...
532	47.269856155	10.0.0.20	10.0.0.10	TCP	66	52850 → 8080 [ACK] Seq=135 Ack=125567 Win=62720 Len=0 TSval=1...
533	47.269913343	10.0.0.10	10.0.2.21	TCP	1514	8080 → 48386 [ACK] Seq=113931 Ack=135 Win=65152 Len=1448 TSva...
534	47.269913475	10.0.0.10	10.0.2.21	TCP	1514	8080 → 48386 [ACK] Seq=115379 Ack=135 Win=65152 Len=1448 TSva...
535	47.269913577	10.0.0.10	10.0.2.21	TCP	1514	8080 → 48386 [ACK] Seq=116827 Ack=135 Win=65152 Len=1448 TSva...
536	47.269913678	10.0.0.10	10.0.2.21	TCP	231	8080 → 48386 [PSH, ACK] Seq=118275 Ack=135 Win=65152 Len=165 ...
537	47.269948467	10.0.2.21	10.0.0.10	TCP	66	48386 → 8080 [ACK] Seq=135 Ack=115379 Win=119680 Len=0 TSval=...
538	47.269949084	10.0.2.21	10.0.0.10	TCP	66	48386 → 8080 [ACK] Seq=135 Ack=116827 Win=122624 Len=0 TSval=...
539	47.269949586	10.0.2.21	10.0.0.10	TCP	66	48386 → 8080 [ACK] Seq=135 Ack=118275 Win=125568 Len=0 TSval=...
540	47.269950078	10.0.2.21	10.0.0.10	TCP	66	48386 → 8080 [ACK] Seq=135 Ack=118440 Win=128384 Len=0 TSval=...
541	47.270103042	10.0.0.10	10.0.2.20	TCP	1514	8080 → 36854 [ACK] Seq=60404 Ack=295 Win=64896 Len=1448 TSval=...
542	47.270103212	10.0.0.10	10.0.2.20	TCP	1514	8080 → 36854 [ACK] Seq=61852 Ack=295 Win=64896 Len=1448 TSval=...
543	47.270103313	10.0.0.10	10.0.2.20	TCP	1514	8080 → 36854 [ACK] Seq=63300 Ack=295 Win=64896 Len=1448 TSval=...
544	47.270103419	10.0.0.10	10.0.2.20	TCP	231	8080 → 36854 [PSH, ACK] Seq=64748 Ack=295 Win=64896 Len=165 T...
545	47.270128725	10.0.2.20	10.0.0.10	TCP	66	36854 → 8080 [ACK] Seq=295 Ack=61852 Win=64128 Len=0 TSval=24...
546	47.270129314	10.0.2.20	10.0.0.10	TCP	66	36854 → 8080 [ACK] Seq=295 Ack=63300 Win=63488 Len=0 TSval=24...
547	47.270129809	10.0.2.20	10.0.0.10	TCP	66	36854 → 8080 [ACK] Seq=295 Ack=64748 Win=62848 Len=0 TSval=24...
548	47.270130299	10.0.2.20	10.0.0.10	TCP	66	36854 → 8080 [ACK] Seq=295 Ack=64913 Win=62720 Len=0 TSval=24...
549	47.754774237	10.0.0.10	10.0.0.20	HTTP	66	HTTP/1.0 200 OK
550	47.755650634	10.0.0.10	10.0.2.21	HTTP	66	HTTP/1.0 200 OK
551	47.755705580	10.0.0.10	10.0.2.20	HTTP	66	HTTP/1.0 200 OK
552	47.755830018	10.0.2.20	10.0.0.10	TCP	66	36854 → 8080 [FIN, ACK] Seq=295 Ack=64914 Win=64128 Len=0 TSv...
553	47.755835386	10.0.0.10	10.0.2.20	TCP	66	8080 → 36854 [ACK] Seq=64914 Ack=296 Win=64896 Len=0 TSval=30...
554	47.799158752	10.0.0.10	10.0.0.10	TCP	66	52850 → 8080 [ACK] Seq=135 Ack=125568 Win=64128 Len=0 TSval=1...
555	47.799186918	10.0.2.21	10.0.0.10	TCP	66	48386 → 8080 [ACK] Seq=135 Ack=118441 Win=128384 Len=0 TSval=...
556	48.021987475	10.0.0.1	224.0.0.5	OSPF	78	Hello Packet
557	50.022900200	10.0.0.1	224.0.0.5	OSPF	78	Hello Packet
558	52.023909260	10.0.0.1	224.0.0.5	OSPF	78	Hello Packet

Frame 1: 78 bytes on wire (624 bits), 78 bytes captured (624 bits) on interface veth8.0.5f, id 0

```

0000 01 00 5e 00 00 05 00 00 00 aa 00 02 00 00 45 c0  ..A....E
0010 00 40 10 1e 00 00 01 59 04 e1 0a 00 00 01 e0 00  0....Y.....
0020 00 05 02 01 00 2c 0a 00 00 01 00 00 00 e8 c6  .....
0030 00 00 00 00 00 00 00 00 00 00 ff ff ff 00 00 02  .....
0040 02 01 00 00 00 06 0a 00 00 01 00 00 00 00  .....

```

wireshark\_veth8.0.5f\_20211102105754\_NuRoZe.pcapng

Figura 11: 3 Clientes - VLC, Firefox e fmeig

Através da análise, tanto do tráfego capturado pelo *Wireshark*, como dos dados estatísticos do fluxo, concluímos que a solução não é escalável. As capturas do *Wireshark* permitiram verificar que o servidor responde aos pedidos do clientes, individualmente, ou seja, um de cada vez. Desta forma, se o número de clientes na rede aumentar de forma significativa, haverá perda de qualidade de serviço, sendo que este ficará mais lento, uma vez que o servidor responde a um pedido de cada vez. Esta diminuição da qualidade de serviço vê-se nos dados estatísticos do fluxo (da aplicação Wireshark), através da diminuição do *bitrate*.

## 1.2 Etapa 2

### Questão 2

Diga qual a largura de banda necessária, em bits por segundo, para que o cliente de streaming consiga receber o vídeo no firefox e qual a pilha protocolar usada neste cenário.

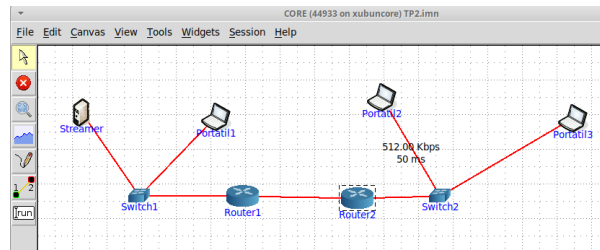


Figura 12: Topologia core utilizada para débito igual a 512Kbps

**R:** Após serem efetuados alguns testes variando a largura de banda (64, 256 e 512Kbps) da ligação entre o portátil 2 e o switch 2 foi possível verificar que para débitos inferiores a 512Kbps não era possível receber o vídeo no firefox. Desta forma pode-se concluir que, neste caso, a largura de banda mínima necessária para transmitir o vídeo é aproximadamente 512Kbps. A pilha protocolar utilizada neste cenário é TCP/IP.

### Questão 3

Ajuste o débito dos links da topologia de modo que o cliente no portátil 2 exiba o vídeo de menor resolução e o cliente no portátil 1 exiba o vídeo com mais resolução. Mostre evidências.

122622	15.945159047	10.0.0.20	10.0.0.10	HTTP	405 GET /video2_540_360_1000k_dash.mp4 HTTP/1.1
124638	16.026635044	10.0.0.10	10.0.0.20	MP4	534
127056	50.911963746	10.0.0.10	10.0.2.21	HTTP	716 HTTP/1.1 200 Ok (application/x-javascript)
127066	51.268836339	10.0.2.21	10.0.0.10	HTTP	379 GET /favicon.ico HTTP/1.1
127068	51.268950694	10.0.0.10	10.0.2.21	HTTP	741 HTTP/1.1 404 Not Found (text/html)
127070	51.277216732	10.0.2.21	10.0.0.10	HTTP	426 GET /video_manifest.mpd HTTP/1.1
127082	51.277960673	10.0.0.10	10.0.2.21	HTTP/XML	717 HTTP/1.1 200 OK
127094	51.459119905	10.0.2.21	10.0.0.10	HTTP	364 GET /video_manifest.mpd HTTP/1.1
127106	51.459447215	10.0.0.10	10.0.2.21	HTTP/XML	717 HTTP/1.1 200 OK
127132	52.295555539	10.0.2.21	10.0.0.10	HTTP	369 GET /video_manifest_init.mp4 HTTP/1.1
127134	52.296006758	10.0.0.10	10.0.2.21	MP4	1060
127141	52.446791931	10.0.2.21	10.0.0.10	HTTP	399 GET /video2_540_360_1000k_dash.mp4 HTTP/1.1
130347	98.053887355	10.0.0.10	10.0.2.21	MP4	534
130367	115.779705311	10.0.2.21	10.0.0.10	HTTP	399 GET /video2_180_120_200k_dash.mp4 HTTP/1.1
131010	124.740803305	10.0.0.10	10.0.2.21	MP4	152
131028	125.422610679	10.0.2.21	10.0.0.10	HTTP	400 GET /video2_180_120_200k_dash.mp4 HTTP/1.1

Figura 13: Filtragem de pacotes HTTP da captura

**R:** Ajustando o débito da ligação entre o portátil 2 e o switch 2 para 512Kbps podemos observar que inicialmente o portátil 2 fez um pedido do vídeo com a resolução máxima (pacote nº127141). Ao longo da transmissão, e após algum

tempo verifica-se que é feito um pedido novamente, no entanto desta vez é utilizada uma menor resolução como o esperado (pacote nº 130367). Isto acontece porque não foi possível reproduzir o vídeo inicial de forma correta.

#### Questão 4

Descreva o funcionamento do DASH neste caso concreto, referindo o papel do ficheiro MPD criado.

**R:** Como se verificou na questão anterior houve um ajuste dinâmico da resolução do vídeo transmitido ao portátil 2. Como inicialmente foi definida uma resolução alta, devido à fraca qualidade da rede verificou-se delay na chegada dos pacotes, ou seja, não estava a ser possível reproduzir o vídeo de forma contínua. Desta forma, o DASH foi o responsável por analisar a largura de banda disponível e com recurso ao ficheiro MPD selecionar a resolução do vídeo mais adequada à situação. Ao reduzir a qualidade de imagem é necessário uma largura de banda mínima menor para reproduzir o vídeo, logo já foi possível exibí-lo de forma ininterrupta.

### 1.3 Etapa 3

#### Questão 5

Compare o cenário unicast aplicado com o cenário multicast. Mostre vantagens e desvantagens na solução multicast ao nível da rede, no que diz respeito a escalabilidade (aumento do nº de clientes) e tráfego na rede. Tire as suas conclusões.

**R:** No **cenário aplicado (Unicast)**, como estamos perante um servidor centralizado, é alocado um canal de transmissão de dados unicast para atender os pedidos de cada cliente. No entanto, como a banda do servidor é um recurso limitado, esta forma de implementação torna-se inviável quando muitos clientes precisam de obter respostas aos seus pedidos simultaneamente, pois a demanda por banda cresce linearmente com a taxa de chegada de pedidos. Por outro lado, o cenário Unicast é mais fácil de configurar.

Em relação ao **cenário Multicast**, a transmissão de dados é feita para múltiplos clientes simultaneamente, isto é, os pacotes são transmitidos eficientemente para múltiplos pontos distintos ao mesmo tempo. Além disso, as mensagens só passam por um link uma única vez e somente são duplicadas quando o link para os destinatários se divide em duas direções. Isto permite o aumento da eficiência das comunicações de um para muitos, sendo, portanto, esta solução escalável. Além disso, como os switches vão mandar o vídeo apenas para os clientes que o peçam, a qualidade do mesmo irá ser preservada e o vídeo continuará a ser visualizável. Outra vantagem desta solução, é que permite o menor uso da largura de banda, uma vez que, os pacotes são enviados de acordo com as necessidades dos recetores, permitindo assim que a mesma não seja limitada pela largura de banda do *access end* do cliente. Além disso, com base

no descrito, podemos concluir que permite a minimização do tráfego na rede.

Por outro lado, existem também desvantagens no que toca ao cenário Multicast. Primeiramente, comparativamente com o cenário Unicast, não existe um mecanismo de correção de erros. Portanto, é uma solução que não é fiável, para contrariar isto, deve-se responsabilizar a camada aplicacional para a resolução deste problema. Além disso é de notar que como não existem mecanismos de controlo de congestão como no TCP, uma vez que o protocolo da camada de transporte utilizado neste tipo de cenário é o UDP, existe a possibilidade de haver congestão na rede.

## 2 Conclusões

Neste segundo trabalho prático, foi possível consolidar o conhecimento adquirido ao longo das aulas teóricas da unidade curricular.

Na primeira etapa, foi feito Streaming HTTP simples, no entanto, verificou-se que apesar de ser uma opção popular, não é a mais viável pois não permite que a solução seja escalável.

Já na segunda etapa, foi feito Streaming adaptativo sobre HTTP, tendo sido analisado o papel do DASH no ajustamento da resolução dos vídeos de acordo com a qualidade da rede.

Por último, na etapa 3, foi feito Streaming sobre UDP, tanto *unicast* como *multicast*, tendo-se comparando estas duas opções. Esta análise permitiu verificar que o cenário multicast permite a escalabilidade da rede, além de reduzir o seu tráfego.

Concluindo, o grupo considera que realizou o trabalho com sucesso tendo aprofundado os seus conhecimentos em *video streaming*.