

3- Wireshark

Apachee Server :

1	0.000000000	127.0.0.1	127.0.0.1	TCP	76	42128 → 80 [SYN] Seq=0 Win=65495 Len=0 MSS=65495 SACK_PERM=1 TSval=3046309511 TSecr=0 WS=128
2	0.000010599	127.0.0.1	127.0.0.1	TCP	76	80 → 42128 [SYN, ACK] Seq=0 Ack=1 Win=65483 Len=0 MSS=65495 SACK_PERM=1 TSval=3046309511 TSecr=3046309511 WS=128
3	0.000017665	127.0.0.1	127.0.0.1	TCP	68	42128 → 80 [ACK] Seq=1 Ack=1 Win=65536 Len=0 TSval=3046309512 TSecr=3046309511
4	0.000032333	127.0.0.1	127.0.0.1	HTTP	208	GET / HTTP/1.1
5	0.000074246	127.0.0.1	127.0.0.1	TCP	68	80 → 42128 [ACK] Seq=1 Ack=141 Win=65408 Len=0 TSval=3046309512 TSecr=3046309512
6	0.001109484	127.0.0.1	127.0.0.1	HTTP	1203	HTTP/1.1 200 OK (text/html)
7	0.001129740	127.0.0.1	127.0.0.1	TCP	68	42128 → 80 [ACK] Seq=141 Ack=1136 Win=64512 Len=0 TSval=3046309513 TSecr=3046309513
8	0.001552695	127.0.0.1	127.0.0.1	TCP	68	42128 → 80 [FIN, ACK] Seq=141 Ack=1136 Win=65536 Len=0 TSval=3046309513 TSecr=3046309513
9	0.001658821	127.0.0.1	127.0.0.1	TCP	68	80 → 42128 [FIN, ACK] Seq=1136 Ack=142 Win=65536 Len=0 TSval=3046309513 TSecr=3046309513
10	0.001670775	127.0.0.1	127.0.0.1	TCP	68	42128 → 80 [ACK] Seq=142 Ack=1137 Win=65536 Len=0 TSval=3046309513 TSecr=3046309513

In our code scrap.py , we accessed to the http server 127.0.0.1 to analyse the website and get the list of files.

- These packets 1 to 3 , it's a TCP 3-Way Handshake Process.
- The packet 4 : GET we are asking for the html code and all that makes up the site
- The packet 5 : ACK to respond that the server get our demand GET
- The packet 6 : This packet is the server response to the previous HTTP request. "HTTP/1.1" indicates the protocol version used, "200" is the response status code: means that the request was processed successfully. The content of the requested resource (for example, the HTML code for the home page) is also included in this package.
- The packet 7 : ACK to respond that we get the reponse from the server
- These packet 8 to 10 : To close the TCP connection

DHCP Server :

11	14.117431325	0.0.0.0	255.255.255.255	DHCP	288	DHCP Discover - Transaction ID 0x0
12	15.171060568	10.20.30.40	255.255.255.255	DHCP	318	DHCP Offer - Transaction ID 0x0
13	15.221812265	0.0.0.0	255.255.255.255	DHCP	300	DHCP Request - Transaction ID 0x0
14	15.222239394	192.168.214.254	255.255.255.255	DHCP	344	DHCP NAK - Transaction ID 0x0
15	16.261611942	10.20.30.40	255.255.255.255	DHCP	318	DHCP ACK - Transaction ID 0x0

These packets represent the communication between the client and the DHCP server.

- The packet 11 : DHCP Discover: This packet is sent by the DHCP client to discover the DHCP servers available on the local network. The client essentially requests that any DHCP server respond with an IP address offer and other configuration information.

- The packet 12 : DHCP Offer: This packet is sent by a DHCP server in response to a DHCP Discover packet. It contains an IP address offer for the client, along with other configuration information, such as subnet mask, default gateway, and DNS servers.

11	14.117431325	0.0.0.0	255.255.255.255	DHCP	288 DHCP Discover	- Transaction ID 0x0
12	15.171060568	10.20.30.40	255.255.255.255	DHCP	318 DHCP Offer	- Transaction ID 0x0
13	15.221812265	0.0.0.0	255.255.255.255	DHCP	300 DHCP Request	- Transaction ID 0x0

Wireshark - Packet 12 - All-New.pcapng	
12	Transaction ID: 0x00000000 Seconds elapsed: 0 Bootp flags: 0x0000 (Unicast) 0... .. = Broadcast flag: Unicast .000 0000 0000 0000 = Reserved flags: 0x0000 Client IP address: 0.0.0.0 Your (client) IP address: 10.20.30.51 Next server IP address: 10.20.30.40 Relay agent IP address: 0.0.0.0 Client MAC address: 30:30:3a:30:63:3a (30:30:3a:30:63:3a) Client hardware address padding: 32393a37393a32633a66 Server host name not given Boot file name not given Magic cookie: DHCP Option: (53) DHCP Message Type (Offer) Option: (54) DHCP Server Identifier (10.20.30.40) Option: (1) Subnet Mask (255.255.255.0) Option: (3) Router Option: (6) Domain Name Server Option: (51) IP Address Lease Time Option: (255) End

- This is the packet 12 : DHCP Offer , as we can see the DHCP Server (10.20.30.40) send the IP : 10.20.30.51 and also different option like Subnet Mask.
- The packet 13 : DHCP Request: This packet is sent by the DHCP client to accept the IP address offer and configuration information received from the DHCP server. It also informs other DHCP servers on the network that the client has accepted a particular server's offer, and that other offers (if any) are no longer needed. As we can see the request from the client , ask for the IP 10.20.30.51 to the Server 10.20.30.40.

11	14.117431325	0.0.0.0	255.255.255.255	DHCP	288 DHCP Discover	- Transaction ID 0x0
12	15.171060568	10.20.30.40	255.255.255.255	DHCP	318 DHCP Offer	- Transaction ID 0x0
13	15.221812265	0.0.0.0	255.255.255.255	DHCP	300 DHCP Request	- Transaction ID 0x0

Wireshark - Packet 13	
13	Hardware type: Ethernet (0x01) Hardware address length: 6 Hops: 0 Transaction ID: 0x00000000 Seconds elapsed: 0 Bootp flags: 0x0000 (Unicast) 0... .. = Broadcast flag: Unicast .000 0000 0000 0000 = Reserved flags: 0x0000 Client IP address: 0.0.0.0 Your (client) IP address: 0.0.0.0 Next server IP address: 0.0.0.0 Relay agent IP address: 0.0.0.0 Client MAC address: VMware_79:2c:f9 (00:0c:29:79:2c:f9) Client hardware address padding: 00000000000000000000 Server host name not given Boot file name not given Magic cookie: DHCP Option: (53) DHCP Message Type (Request) Option: (50) Requested IP Address (10.20.30.51) Option: (54) DHCP Server Identifier (10.20.30.40) Option: (255) End

- The packet 15 : DHCP ACK: This packet is sent by the DHCP server to

Transaction ID: 0x00000000
 Seconds elapsed: 0
 Bootp flags: 0x0000 (Unicast)
 = Broadcast flag: Unicast
 = Reserved flag: 0x0000
 Client IP address: 0.0.0.0
 Your (client) IP address: 0.0.0.0
 Next server IP address: 0.0.0.0
 Relay agent IP address: 0.0.0.0
 Client MAC address: 08:00:20:08:00:00 (08:00:20:08:00:00)
 Client hardware address padding: 00:00:00:00:00:00:00:00
 Server host name not given
 Boot file name not given
 Magic cookie: DHCP
 Options:
 Option (55) DHCP Message Type (ACK)
 Option (54) DHCP Server Identifier (0.0.0.0)
 Option (1) Subnet Mask (255.255.255.0)
 Option (3) Router
 Option (6) Domain Name Server
 Option (51) IP Address Lease Time
 Option (255) End

confirm that the client has received the IP address offer and configuration information. This completes the IP address assignment process and the client can begin using the assigned IP address and configuration information to communicate on the network.

DNS Server :

No.	Time	Source	Destination	Protocol	Length	Info
15	16.261611942	10.20.30.40	255.255.255.255	DHCP	318	DHCP ACK - Transaction ID 0x0
16	16.293353905	10.20.30.51	127.0.0.1	DNS	73	Standard query 0xf121 A example.com
17	16.293525392	127.0.0.1	10.20.30.51	DNS	89	Standard query response 0xf121 A example.com A 127.0.0.10
18	16.295680954	10.20.30.51	127.0.0.10	UDP	54	49152 → 49152 Len=10
19	16.295812159	127.0.0.10	10.20.30.51	UDP	51	49152 → 49152 Len=7
20	16.221466437	10.20.30.51	127.0.0.10	UDP	75	49152 → 49152 Len=31

Wireshark · Packet 16 · All-New.pcapng

Frame 16: 73 bytes on wire (584 bits), 73 bytes captured (584 bits) on interface any, id 0
 Linux cooked capture v1
 Internet Protocol Version 4, Src: 10.20.30.51, Dst: 127.0.0.1
 User Datagram Protocol, Src Port: 52299, Dst Port: 53
 Domain Name System (query)
 Transaction ID: 0xf121
 Flags: 0x0100 Standard query
 Questions: 1
 Answer RRs: 0
 Authority RRs: 0
 Additional RRs: 0
 Queries
 example.com: type A, class IN
 Name: example.com
 [Name Length: 11]
 [Label Count: 2]
 Type: A (Host Address) (1)
 Class: IN (0x0001)
[\[Response In: 17\]](#)

The packet 16 and 17 is the communication between the client and the server DNS.

- Packet 16 : The client send a Querie to our DNS Server with the domain name "example.com" when the Type is : A. As we can see the reponse of the Querie is in Packet 17.
- Packet 17 : The DNS Server sent to the client the IP of the "example.com" Type A when the adress of this domain is : 127.0.0.10

15	16.261611942	10.20.30.40	255.255.255.255	DHCP	318 DHCP ACK	- Transaction ID 0x0
16	16.293353905	10.20.30.51	127.0.0.1	DNS	73 Standard query 0xf121 A example.com	
17	16.293525392	127.0.0.1	10.20.30.51	DNS	89 Standard query response 0xf121 A example.com A 127.0.0.10	
18	16.295680954	10.20.30.51	127.0.0.10	UDP	54 49152 - 49152 Len=10	

Wireshark · Packet 17 · All-New.pcapng

- Linux cooked capture v1
- Internet Protocol Version 4, Src: 127.0.0.1, Dst: 10.20.30.51
- User Datagram Protocol, Src Port: 53, Dst Port: 52299
- Domain Name System (response)
 - Transaction ID: 0xf121
 - Flags: 0x8400 Standard query response, No error
 - Questions: 1
 - Answer RRs: 1
 - Authority RRs: 0
 - Additional RRs: 0
- Queries
 - example.com: type A, class IN
 - Name: example.com
 - [Name Length: 11]
 - [Label Count: 2]
 - Type: A (Host Address) (1)
 - Class: IN (0x0001)
- Answers
 - example.com: type A, class IN, addr 127.0.0.10

[Request In: 16]
[Time: 0.000171487 seconds]

RUDP Connection : Client ↔ Rudproxy

Now we are going to analyse the different packets between the Client and the rudproxy. The communication is going to be in RUDP.

Client : 10.20.30.51

RUDP Server : 127.0.0.10

15	16.261611942	10.20.30.40	255.255.255.255	DHCP	318 DHCP ACK	- Transaction ID 0x0
16	16.293353905	10.20.30.51	127.0.0.1	DNS	73 Standard query 0xf121 A example.com	
17	16.293525392	127.0.0.1	10.20.30.51	DNS	89 Standard query response 0xf121 A example.com A 127.0.0.10	
18	16.295680954	10.20.30.51	127.0.0.10	UDP	54 49152 - 49152 Len=10	

Wireshark · Packet 18 · All-New.pcapng

- Frame 18: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface any, id 0
- Linux cooked capture v1
- Internet Protocol Version 4, Src: 10.20.30.51, Dst: 127.0.0.10
- User Datagram Protocol, Src Port: 49152, Dst Port: 49152
- Data (10 bytes)
 - Data: 5349474e45573a54616c
 - [Length: 10]

```

0000  00 00 03 04 00 06 00 00 00 00 00 00 08 00  .....
0010  45 00 00 26 3d fd 40 00 40 11 55 79 0a 14 1e 33  E: &= @. @. Uy ... 3
0020  7f 00 00 0a c0 00 c0 00 00 12 a7 74 53 49 47 4e  ....tSIGN
0030  45 57 3a 54 61 6c  ....EW: Tal

```

- The packet 18 : The client start a new connection with the Proxy Server.
As we can see in the Data the client sent a SIGNEW + name of user to start the discussion.

17	16.293525392	127.0.0.1	10.20.30.51	DNS	89 Standard query response 0xf121 A example.com A 127.0.0.10
18	16.295680954	10.20.30.51	127.0.0.10	UDP	54 49152 → 49152 Len=10
19	16.295812159	127.0.0.10	10.20.30.51	UDP	51 49152 → 49152 Len=7

Wireshark · Packet 19 · All-New.pcapng					
▶ Frame 19: 51 bytes on wire (408 bits), 51 bytes captured (408 bits) on interface any, id 0 ▶ Linux cooked capture v1 ▶ Internet Protocol Version 4, Src: 127.0.0.10, Dst: 10.20.30.51 ▶ User Datagram Protocol, Src Port: 49152, Dst Port: 49152 ▶ Data (7 bytes)					
Data: 5349474e41434b [Length: 7]					

0000	00 00 03 04 00 06 00 00 00 00 00 00 00 08 00
0010	45 00 00 23 53 b8 40 00 40 11 3f c1 7f 00 00 0a	E...#S...@...?.....
0020	0a 14 1e 33 c0 00 c0 00 00 0f a7 71 53 49 47 4e	...3...qSIGN
0030	41 43 4b	ACK

- The packet 19 : This is the response of the Server with the SIGNACK that he accept the connection with the Client.

17	16.293525392	127.0.0.1	10.20.30.51	DNS	89 Standard query response 0xf121 A example.com A 127.0.0.10
18	16.295680954	10.20.30.51	127.0.0.10	UDP	54 49152 → 49152 Len=10
19	16.295812159	127.0.0.10	10.20.30.51	UDP	51 49152 → 49152 Len=7
20	26.221466437	10.20.30.51	127.0.0.10	UDP	75 49152 → 49152 Len=31

Wireshark · Packet 20 · All-New.pcapng					
▶ Frame 20: 75 bytes on wire (600 bits), 75 bytes captured (600 bits) on interface any, id 0 ▶ Linux cooked capture v1 ▶ Internet Protocol Version 4, Src: 10.20.30.51, Dst: 127.0.0.10 ▶ User Datagram Protocol, Src Port: 49152, Dst Port: 49152 ▶ Data (31 bytes)					
Data: 5349474e4543484f3a3239303b313638323637323334322e38303433313837 [Length: 31]					

0000	00 00 03 04 00 06 00 00 00 00 00 00 00 08 00
0010	45 00 00 3b 46 b5 40 00 40 11 4c ac 0a 14 1e 33	E...;F...@...L...3
0020	7f 00 00 0a c0 00 c0 00 00 27 a7 89 53 49 47 4eSIGN
0030	45 43 48 4f 3a 32 39 30 3b 31 36 38 32 36 37 32	ECHO:290 ;1682672
0040	33 34 32 2e 38 30 34 33 31 38 37	342.8043 187

- Packet 20 : The client sent to the Server a new SIGNAL : SIGNECHO to know the ping between him and the Server. Packet ID : 290

18	16.295680954	10.20.30.51	127.0.0.10	UDP	54	49152 → 49152	Len=10
19	16.295812159	127.0.0.10	10.20.30.51	UDP	51	49152 → 49152	Len=7
20	26.221466437	10.20.30.51	127.0.0.10	UDP	75	49152 → 49152	Len=31
21	26.221611721	127.0.0.10	10.20.30.51	UDP	76	49152 → 49152	Len=32

Wireshark · Packet 21 · All-New.pcapng

▶ Frame 21: 76 bytes on wire (608 bits), 76 bytes captured (608 bits) on interface any, id 0
 ▶ Linux cooked capture v1
 ▶ Internet Protocol Version 4, Src: 127.0.0.10, Dst: 10.20.30.51
 ▶ User Datagram Protocol, Src Port: 49152, Dst Port: 49152
 ▼ Data (32 bytes)
 Data: 4543484f5245504c593a3239303b313638323637323334322e38303433313837
 [Length: 32]

0000	00 00 03 04 00 06 00 00	00 00 00 00 ca 7f 08 00
0010	45 00 00 3c 5d 42 40 00	40 11 36 1e 7f 00 00 0a	E...<]B@. @.6.....
0020	0a 14 1e 33 c0 00 c0 00	00 28 a7 8a 45 43 48 4f	...3.....(.ECHO
0030	52 45 50 4c 59 3a 32 39	30 3b 31 36 38 32 36 37	REPLY:29 0;168267
0040	32 33 34 32 2e 38 30 34	33 31 38 37	2342.804 3187

- The packet 21 : This is the response of the SIGNECHO , the server sent the time and also the SIGN : ECHOREPLY to the client.

20	26.221466437	10.20.30.51	127.0.0.10	UDP	75	49152 → 49152	Len=31
21	26.221611721	127.0.0.10	10.20.30.51	UDP	76	49152 → 49152	Len=32
22	35.061271569	10.20.30.51	127.0.0.10	UDP	56	49152 → 49152	Len=12

Wireshark · Packet

▶ Frame 22: 56 bytes on wire (448 bits), 56 bytes captured (448 bits) on interface any, id 0
 ▶ Linux cooked capture v1
 ▶ Internet Protocol Version 4, Src: 10.20.30.51, Dst: 127.0.0.10
 ▶ User Datagram Protocol, Src Port: 49152, Dst Port: 49152
 ▼ Data (12 bytes)
 Data: 5349474e535441543a323931
 [Length: 12]

0000	00 00 03 04 00 06 00 00	00 00 00 00 01 00 08 00
0010	45 00 00 28 4b ca 40 00	40 11 47 aa 0a 14 1e 33	E...(K@. @.G...3
0020	7f 00 00 0a c0 00 c0 00	00 14 a7 76 53 49 47 4evSIGN
0030	53 54 41 54 3a 32 39 31		STAT:291

- The packet 22 : The client sent to the Proxy Server a new SIGNAL : SIGNSTAT to know some information about the connection (packet sent , packet lost , time ...). Packet ID : 291.

20	26.221466437	10.20.30.51	127.0.0.10	UDP	75	49152 → 49152	Len=31
21	26.221611721	127.0.0.10	10.20.30.51	UDP	76	49152 → 49152	Len=32
22	35.061271569	10.20.30.51	127.0.0.10	UDP	56	49152 → 49152	Len=12
23	35.061394934	127.0.0.10	10.20.30.51	UDP	119	49152 → 49152	Len=75

Wireshark · Packet 23 · All-New.pcapng							
▶ Frame 23: 119 bytes on wire (952 bits), 119 bytes captured (952 bits) on interface any, id 0 ▶ Linux cooked capture v1 ▶ Internet Protocol Version 4, Src: 127.0.0.10, Dst: 10.20.30.51 ▶ User Datagram Protocol, Src Port: 49152, Dst Port: 49152 ▶ Data (75 bytes) Data: 535441545245504c593a3239313b5061636b6574732073656e743a20320a5061636b6574... [Length: 75]							

0000	00 00 03 04 00 06 00 00	00 00 00 00 25 12 08 00 %..
0010	45 00 00 67 60 57 40 00	40 11 32 de 7f 00 00 0a	E-g`w@. @.2.....
0020	0a 14 1e 33 c0 00 c0 00	00 53 a7 b5 53 54 41 54	...3.... S..STAT
0030	52 45 50 4c 59 3a 32 39	31 3b 50 61 63 6b 65 74	REPLY:29 1;Packet
0040	73 20 73 65 6e 74 3a 20	32 0a 50 61 63 6b 65 74	s sent: 2;Packet
0050	73 20 6c 6f 73 74 3a 20	30 0a 43 6f 6e 6e 65 63	s lost: 0;Connec
0060	74 65 64 20 74 69 6d 65	3a 20 31 38 2e 37 37 20	ted time : 18.77
0070	73 65 63 6f 6e 64 73		seconds

- The packet 23 : The Server sent a reponse after the SIGNSTAT with a the flag : STATREPLY , as we can see in the Data the server sent him different information.

23	35.061394934	127.0.0.10	10.20.30.51	UDP	119	49152 → 49152	Len=75
24	47.572725118	10.20.30.51	127.0.0.10	UDP	63	49152 → 49152	Len=19

Wireshark · Packet 24 · All-New.pcapng							
▶ Frame 24: 63 bytes on wire (504 bits), 63 bytes captured (504 bits) on interface any, id 0 ▶ Linux cooked capture v1 ▶ Internet Protocol Version 4, Src: 10.20.30.51, Dst: 127.0.0.10 ▶ User Datagram Protocol, Src Port: 49152, Dst Port: 49152 ▶ Data (19 bytes) Data: 5349474e4745543a3239323b5349474e474554 [Length: 19]							

0000	00 00 03 04 00 06 00 00	00 00 00 00 13 54 08 00 T..
0010	45 00 00 2f 56 2c 40 00	40 11 3d 41 0a 14 1e 33	E-/V,@. @=A...3
0020	7f 00 00 0a c0 00 c0 00	00 1b a7 7d 53 49 47 4e }SIGN
0030	47 45 54 3a 32 39 32 3b	53 49 47 4e 47 45 54	GET:292; SIGNGET

- The packet 24 : The client sent to the server a new SIGNAL : SIGNGET to get the list of files that he can download. Packet ID : 292

Connection TCP : Proxy Server ↔ Scrap Server / Redirect

25	47.572948692	127.0.0.10	127.0.0.5	TCP	76	49153 → 49153 [SYN] Seq=0 Win=65495 Len=0 MSS=65495 SACK_PERM=1 TSval=1618098717 TSecr=0 WS=128
26	47.572959384	127.0.0.5	127.0.0.10	TCP	76	49153 → 49153 [SYN, ACK] Seq=0 Ack=1 Win=65483 Len=0 MSS=65495 SACK_PERM=1 TSval=1825027792 TSecr=1618098717 WS=128
27	47.572967258	127.0.0.10	127.0.0.5	TCP	68	49153 → 49153 [ACK] Seq=1 Ack=1 Win=65536 Len=0 TSval=1618098717 TSecr=1825027792
28	47.573040217	127.0.0.10	127.0.0.5	TCP	75	49153 → 49153 [PSH, ACK] Seq=1 Ack=1 Win=65536 Len=7 TSval=1618098718 TSecr=1825027792
29	47.573045393	127.0.0.5	127.0.0.10	TCP	68	49153 → 49153 [ACK] Seq=1 Ack=8 Win=65536 Len=0 TSval=1825027793 TSecr=1618098718
30	47.573118524	127.0.0.5	127.0.0.10	TCP	122	49153 → 49153 [PSH, ACK] Seq=1 Ack=8 Win=65536 Len=54 TSval=1825027793 TSecr=1618098718
31	47.573134875	127.0.0.10	127.0.0.5	TCP	68	49153 → 49153 [ACK] Seq=8 Ack=55 Win=65536 Len=0 TSval=1618098718 TSecr=1825027793
32	47.573201453	127.0.0.5	127.0.0.10	TCP	68	49153 → 49153 [FIN, ACK] Seq=55 Ack=8 Win=65536 Len=0 TSval=1825027793 TSecr=1618098718
33	47.573206052	127.0.0.10	127.0.0.5	TCP	68	49153 → 49153 [FIN, ACK] Seq=8 Ack=55 Win=65536 Len=0 TSval=1618098718 TSecr=1825027793
34	47.573211222	127.0.0.5	127.0.0.10	TCP	68	49153 → 49153 [ACK] Seq=56 Ack=9 Win=65536 Len=0 TSval=1825027793 TSecr=1618098718
35	47.573213997	127.0.0.10	127.0.0.5	TCP	68	49153 → 49153 [ACK] Seq=9 Ack=56 Win=65536 Len=0 TSval=1618098718 TSecr=1825027793

The client ask the list of differents files that he can download. The proxy Server perform a redirect to the Scrap Server with a connection TCP.

- The packets 25 to 27 : it's a TCP 3-Way Handshake Process. As we can see the connection it's between 127.0.0.10 and 127.0.0.5 (Proxy Server and Scrap Server)
- The packet 28 : [PSH,ACK] 127.0.0.10 → 127.0.0.5 The Proxy Server sent the request to the Scrap Server to get the list of files.
- The packet 29 : The scrap Server sent to the Proxy an ACK to assure to him that he get his request. We can see the list in Data (54 bytes)
- The packet 30 : [PSH,ACK] 127.0.0.5 → 127.0.0.10 The Scrap Server sent the list of files to the Proxy Server.

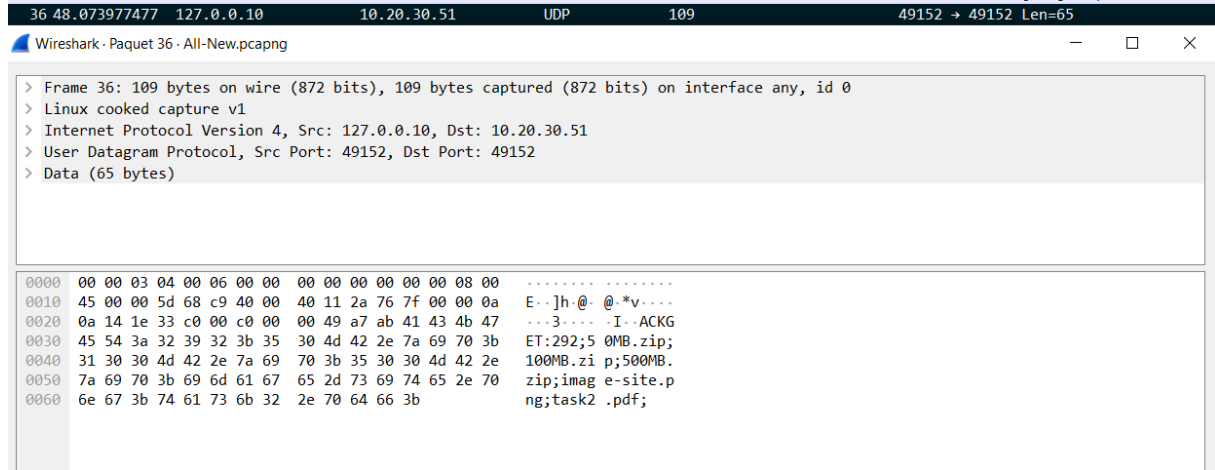
29	47.573045393	127.0.0.5	127.0.0.10	TCP	68	49153 → 49153 [ACK] Seq=1 Ack=8 Win=65536 Len=0 TSval=1825027793 TSecr=1618098718
30	47.573118524	127.0.0.5	127.0.0.10	TCP	122	49153 → 49153 [PSH, ACK] Seq=1 Ack=8 Win=65536 Len=54 TSval=1825027793 TSecr=1618098718
31	47.573134875	127.0.0.10	127.0.0.5	TCP	68	49153 → 49153 [ACK] Seq=8 Ack=55 Win=65536 Len=0 TSval=1618098718 TSecr=1825027793

Wireshark - Packet 30 - All-New.pcapng	
[TCP Segment Len: 54]	
Sequence Number: 1 (relative sequence number)	
Sequence Number (raw): 1912163157	
[Next Sequence Number: 55 (relative sequence number)]	
Acknowledgment Number: 8 (relative ack number)	
Acknowledgment number (raw): 152600330	
1000 = Header Length: 32 bytes (8)	
* Flags: 0x018 (PSH, ACK)	
Window: 512	
[Calculated window size: 65536]	
[Window size scaling factor: 128]	
Checksum: 0xafeb [unverified]	
[Checksum Status: Unverified]	
Urgent Pointer: 0	
* Options: (12 bytes), No-Operation (NOP), No-Operation (NOP), Timestamps	
* [Timestamps]	
* [SEQ/ACK analysis]	
TCP payload (54 bytes)	
Data (54 bytes)	
Data: 353040422e7a69703b3130304d422e7a69703b3530304d422e7a69703b606d6167652d73...	
[Length: 54]	

0000	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	E . j
0010	45 00 00 0a 00 42 40 00 40 00 8c 3c 7f 00 00 05	... j
0020	7f 00 00 0a c0 01 c0 01 71 f9 4b 55 09 10 06 7a	... q
0030	80 18 02 00 fe 0b 00 00 01 01 08 0a 6c c7 b6 d1	... k
0040	60 72 3a 1e 35 30 4d 42 2e 7a 69 70 3b 31 30 30	"r: 50MB .zip;100
0050	4d 42 2e 7a 69 70 3b 35 30 30 4d 42 2e 7a 69 70	MB.zip;5 00MB.zip
0060	3b 69 0d 61 07 05 2d 73 69 7a 65 2e 70 6e 07 3b	;image:s lte.png;
0070	74 61 73 6b 32 2e 70 64 66 3b	task2.pd f;

- The packet 31 : The proxy Server sent an ACK to the Scrap Server to assure that he got the list of files from him.
- The packets 32 to 35 : To close the connection between the Proxy Server and Scrap Server.

Now the Proxy Server going to send the list to the Client.



The packet 36 : 127.0.0.10 → 10.20.30.51 , the Proxy Server sent now the list of files to the Client. He used the SIGNAL : ACKGET and as we can see in the Data (65 bytes) the different files to download.

The list is :

1. 50MB.zip
2. 100MB.zip
3. 500MB.zip
4. Image-site.png
5. Task2.pdf

For the explanation the client going to download every files.

The first file : 50MB.zip

37	57.116893320	10.20.30.51	127.0.0.1	TCP	76	59369 → 80 [SYN]	Seq=0 Win=65495 Len=0 MSS=65495 SACK_PERM=1 TSval=2209864025 TSecr=0 WS=128
38	57.116906548	127.0.0.1	10.20.30.51	TCP	76	80 → 59369 [SYN, ACK]	Seq=0 Ack=1 Win=65483 Len=0 MSS=65495 SACK_PERM=1 TSval=3138025102 TSecr=2209864025 WS=128
39	57.116915746	10.20.30.51	127.0.0.1	TCP	68	59369 → 80 [ACK]	Seq=1 Ack=1 Win=65536 Len=0 TSval=2209864025 TSecr=3138025102
40	57.116941107	10.20.30.51	127.0.0.1	HTTP	200	GET /50MB.zip HTTP/1.1	
41	57.116987721	127.0.0.1	10.20.30.51	TCP	68	80 → 59369 [ACK]	Seq=1 Ack=133 Win=65408 Len=0 TSval=3138025102 TSecr=2209864025
42	57.119074973	127.0.0.1	10.20.30.51	TCP	32836	80 → 59369 [ACK]	Seq=1 Ack=133 Win=65536 Len=32768 TSval=3138025105 TSecr=2209864025 [TCP segment of a reassembled PDU]
43	57.119102332	10.20.30.51	127.0.0.1	TCP	68	59369 → 80 [ACK]	Seq=133 Ack=32769 Win=48512 Len=0 TSval=2209864028 TSecr=3138025105
44	57.119117719	127.0.0.1	10.20.30.51	TCP	32836	80 → 59369 [PSH, ACK]	Seq=32769 Ack=133 Win=65536 Len=32768 TSval=3138025105 TSecr=2209864025 [TCP segment of a reassembled PDU]
45	57.119502109	10.20.30.51	127.0.0.1	TCP	68	59369 → 80 [ACK]	Seq=133 Ack=65537 Win=48512 Len=0 TSval=2209864028 TSecr=3138025105
46	57.121192484	127.0.0.1	10.20.30.51	TCP	32836	80 → 59369 [ACK]	Seq=65537 Ack=133 Win=65536 Len=32768 TSval=3138025107 TSecr=2209864028 [TCP segment of a reassembled PDU]
47	57.121225005	10.20.30.51	127.0.0.1	TCP	68	59369 → 80 [ACK]	Seq=133 Ack=98305 Win=65536 Len=0 TSval=2209864030 TSecr=3138025107
48	57.123044087	127.0.0.1	10.20.30.51	TCP	32836	80 → 59369 [PSH, ACK]	Seq=98305 Ack=133 Win=65536 Len=32768 TSval=3138025108 TSecr=2209864030 [TCP segment of a reassembled PDU]
49	57.123062775	10.20.30.51	127.0.0.1	TCP	68	59369 → 80 [ACK]	Seq=133 Ack=131073 Win=196480 Len=0 TSval=2209864032 TSecr=3138025108
50	57.123071990	127.0.0.1	10.20.30.51	TCP	32836	80 → 59369 [ACK]	Seq=131073 Ack=133 Win=65536 Len=32768 TSval=3138025108 TSecr=2209864030 [TCP segment of a reassembled PDU]
51	57.123074362	10.20.30.51	127.0.0.1	TCP	68	59369 → 80 [ACK]	Seq=133 Ack=103841 Win=327552 Len=0 TSval=2209864032 TSecr=3138025108
52	57.123083930	127.0.0.1	10.20.30.51	TCP	32836	80 → 59369 [PSH, ACK]	Seq=103841 Ack=133 Win=65536 Len=32768 TSval=3138025109 TSecr=2209864032 [TCP segment of a reassembled PDU]
53	57.123087011	10.20.30.51	127.0.0.1	TCP	68	59369 → 80 [ACK]	Seq=133 Ack=196609 Win=458496 Len=0 TSval=2209864032 TSecr=3138025109
54	57.123097500	127.0.0.1	10.20.30.51	TCP	32836	80 → 59369 [ACK]	Seq=196609 Ack=133 Win=65536 Len=32768 TSval=3138025109 TSecr=2209864032 [TCP segment of a reassembled PDU]
55	57.123099214	10.20.30.51	127.0.0.1	TCP	68	59369 → 80 [ACK]	Seq=133 Ack=229377 Win=589440 Len=0 TSval=2209864032 TSecr=3138025109
56	57.123111628	127.0.0.1	10.20.30.51	TCP	32836	80 → 59369 [PSH, ACK]	Seq=229377 Ack=133 Win=65536 Len=32768 TSval=3138025109 TSecr=2209864032 [TCP segment of a reassembled PDU]
57	57.123113233	10.20.30.51	127.0.0.1	TCP	68	59369 → 80 [ACK]	Seq=133 Ack=262145 Win=720384 Len=0 TSval=2209864032 TSecr=3138025109
58	57.123124554	127.0.0.1	10.20.30.51	TCP	32836	80 → 59369 [ACK]	Seq=262145 Ack=133 Win=65536 Len=32768 TSval=3138025109 TSecr=2209864032 [TCP segment of a reassembled PDU]
59	57.123126225	10.20.30.51	127.0.0.1	TCP	68	59369 → 80 [ACK]	Seq=133 Ack=294913 Win=851328 Len=0 TSval=2209864032 TSecr=3138025109
60	57.123136632	127.0.0.1	10.20.30.51	TCP	32836	80 → 59369 [PSH, ACK]	Seq=294913 Ack=133 Win=65536 Len=32768 TSval=3138025109 TSecr=2209864032 [TCP segment of a reassembled PDU]
61	57.123138278	10.20.30.51	127.0.0.1	TCP	68	59369 → 80 [ACK]	Seq=133 Ack=327681 Win=982272 Len=0 TSval=2209864032 TSecr=3138025109
62	57.123147710	127.0.0.1	10.20.30.51	TCP	32836	80 → 59369 [ACK]	Seq=327681 Ack=133 Win=65536 Len=32768 TSval=3138025109 TSecr=2209864032 [TCP segment of a reassembled PDU]

- The packet 37 to 39 : it's a TCP 3-Way Handshake Process.
- The packet 40 : [200 GET /50MB.zip HTTP/1.1] The client asking the file : 50MB.zip .
- The next packets going to be the download of this file , as we can see many packets of [ACK] and [PSH,ACK].

1273	57.290518245	10.20.30.51	127.0.0.1	TCP	68	59369 → 80 [FIN, ACK]	Seq=133 Ack=52429101 Win=3112448 Len=0 TSval=2209864199 TSecr=3138025275
1274	57.290733825	127.0.0.1	10.20.30.51	TCP	68	80 → 59369 [FIN, ACK]	Seq=52429101 Ack=134 Win=65536 Len=0 TSval=3138025276 TSecr=2209864199

- The packet 1273 and 1274 : The client got the file 50MB.zip , so now he can close the connection with the flag [FIN, ACK].

The second file : 100MB.zip

1276	59.644947108	10.20.30.51	127.0.0.1	TCP	76	47475	-	80	[SYN]	Seq=0	Win=65495	Len=0	MSS=65495	SACK_PERM=1	TSval=2209866553	TSecr=0	WS=128	
1277	59.644961581	127.0.0.1	10.20.30.51	TCP	76	80	-	47475	[SYN, ACK]	Seq=0	Ack=1	Win=65483	Len=0	MSS=65495	SACK_PERM=1	TSval=3138027630	TSecr=2209866553	WS=128
1278	59.644973497	10.20.30.51	127.0.0.1	TCP	68	47475	-	80	[ACK]	Seq=1	Ack=1	Win=65536	Len=0	TSval=2209866553	TSecr=3138027630			
1279	59.645039600	10.20.30.51	127.0.0.1	HTTP	201	GET	/100MB.zip	HTTP/1.1										
1280	59.645112060	127.0.0.1	10.20.30.51	TCP	68	80	-	47475	[ACK]	Seq=1	Ack=134	Win=65408	Len=0	TSval=3138027631	TSecr=2209866554			
1281	59.649327507	127.0.0.1	10.20.30.51	TCP	32836	80	-	47475	[ACK]	Seq=1	Ack=134	Win=65536	Len=32768	TSval=3138027635	TSecr=2209866554	[TCP segment of a reassembled PDU]		
1282	59.649351847	10.20.30.51	127.0.0.1	TCP	68	47475	-	80	[ACK]	Seq=134	Ack=32769	Win=48512	Len=0	TSval=2209866558	TSecr=3138027635			
1283	59.649366791	127.0.0.1	10.20.30.51	TCP	32836	80	-	47475	[PSH, ACK]	Seq=32769	Ack=134	Win=65536	Len=32768	TSval=3138027635	TSecr=2209866554	[TCP segment of a reassembled PDU]		
1284	59.649616026	10.20.30.51	127.0.0.1	TCP	68	47475	-	80	[ACK]	Seq=134	Ack=65537	Win=48512	Len=0	TSval=2209866558	TSecr=3138027635			
1285	59.651601469	127.0.0.1	10.20.30.51	TCP	32836	80	-	47475	[ACK]	Seq=65537	Ack=134	Win=65536	Len=32768	TSval=3138027637	TSecr=2209866558	[TCP segment of a reassembled PDU]		
1286	59.651675258	10.20.30.51	127.0.0.1	TCP	68	47475	-	80	[ACK]	Seq=134	Ack=98305	Win=65536	Len=0	TSval=2209866560	TSecr=3138027637			
1287	59.653489110	127.0.0.1	10.20.30.51	TCP	32836	80	-	47475	[PSH, ACK]	Seq=98305	Ack=134	Win=65536	Len=32768	TSval=3138027639	TSecr=2209866560	[TCP segment of a reassembled PDU]		
1288	59.653507722	10.20.30.51	127.0.0.1	TCP	68	47475	-	80	[ACK]	Seq=134	Ack=131073	Win=196480	Len=0	TSval=2209866562	TSecr=3138027639			
1289	59.653517297	127.0.0.1	10.20.30.51	TCP	32836	80	-	47475	[ACK]	Seq=131073	Ack=134	Win=65536	Len=32768	TSval=3138027639	TSecr=2209866560	[TCP segment of a reassembled PDU]		
1290	59.653519801	10.20.30.51	127.0.0.1	TCP	68	47475	-	80	[ACK]	Seq=134	Ack=163841	Win=327552	Len=0	TSval=2209866562	TSecr=3138027639			
1291	59.653529172	127.0.0.1	10.20.30.51	TCP	32836	80	-	47475	[PSH, ACK]	Seq=163841	Ack=134	Win=65536	Len=32768	TSval=3138027639	TSecr=2209866562	[TCP segment of a reassembled PDU]		
1292	59.653531960	10.20.30.51	127.0.0.1	TCP	68	47475	-	80	[ACK]	Seq=134	Ack=196609	Win=458496	Len=0	TSval=2209866562	TSecr=3138027639			
1293	59.653540630	127.0.0.1	10.20.30.51	TCP	32836	80	-	47475	[ACK]	Seq=196609	Ack=134	Win=65536	Len=32768	TSval=3138027639	TSecr=2209866562	[TCP segment of a reassembled PDU]		
1294	59.653542432	10.20.30.51	127.0.0.1	TCP	68	47475	-	80	[ACK]	Seq=134	Ack=229377	Win=589440	Len=0	TSval=2209866562	TSecr=3138027639			
1295	59.653552736	127.0.0.1	10.20.30.51	TCP	32836	80	-	47475	[PSH, ACK]	Seq=229377	Ack=134	Win=65536	Len=32768	TSval=3138027639	TSecr=2209866562	[TCP segment of a reassembled PDU]		
1296	59.653554337	10.20.30.51	127.0.0.1	TCP	68	47475	-	80	[ACK]	Seq=134	Ack=262145	Win=726384	Len=0	TSval=2209866562	TSecr=3138027639			
1297	59.653572740	127.0.0.1	10.20.30.51	TCP	32836	80	-	47475	[ACK]	Seq=262145	Ack=134	Win=65536	Len=32768	TSval=3138027639	TSecr=2209866562	[TCP segment of a reassembled PDU]		
1298	59.653574366	10.20.30.51	127.0.0.1	TCP	68	47475	-	80	[ACK]	Seq=134	Ack=294913	Win=851328	Len=0	TSval=2209866562	TSecr=3138027639			
1299	59.653587580	127.0.0.1	10.20.30.51	TCP	32836	80	-	47475	[PSH, ACK]	Seq=294913	Ack=134	Win=65536	Len=32768	TSval=3138027639	TSecr=2209866562	[TCP segment of a reassembled PDU]		
1300	59.653589142	10.20.30.51	127.0.0.1	TCP	68	47475	-	80	[ACK]	Seq=134	Ack=327681	Win=982272	Len=0	TSval=2209866562	TSecr=3138027639			

- The packets 1276 to 1278 : it's a TCP 3-Way Handshake Process.
- The packet 1279 : [200 GET /100MB.zip HTTP/1.1] The client asking the file : 100MB.zip .
- The next packets going to be the download of this file , as we can see many packets of [ACK] and [PSH,ACK].

2509	59.813077308	10.20.30.51	127.0.0.1	TCP	68	47475	-	80	[FIN, ACK]	Seq=134	Ack=52429101	Win=3112448	Len=0	TSval=2209866722	TSecr=3138027798		
2510	59.813233747	127.0.0.1	10.20.30.51	TCP	68	80	-	47475	[FIN, ACK]	Seq=52429101	Ack=135	Win=65536	Len=0	TSval=3138027799	TSecr=2209866722		

- The packet 2509 and 2510 : The client got the file 100MB.zip , so now he can close the connection with the flag [FIN, ACK].

The third file : 500MB.zip

2512	62.145964491	10.20.30.51	127.0.0.1	TCP	76	44261 → 80	[SYN]	Seq=0 Win=65495 Len=0 MSS=65495 SACK_PERM=1 TSval=2209869054 TSecr=0 WS=128
2513	62.145985669	127.0.0.1	10.20.30.51	TCP	76	80 → 44261	[SYN, ACK]	Seq=0 Ack=1 Win=65483 Len=0 MSS=65495 SACK_PERM=1 TSval=3138030131 TSecr=2209869054 WS=128
2514	62.145990556	10.20.30.51	127.0.0.1	TCP	68	44261 → 80	[ACK]	Seq=1 Ack=1 Win=65536 Len=0 TSval=2209869054 TSecr=3138030131
2515	62.146067300	10.20.30.51	127.0.0.1	HTTP	201	GET /500MB.zip	HTTP/1.1	
2516	62.146115938	127.0.0.1	10.20.30.51	TCP	68	80 → 44261	[ACK]	Seq=1 Ack=134 Win=65408 Len=0 TSval=3138030132 TSecr=2209869055
2517	62.158497727	127.0.0.1	10.20.30.51	TCP	32836	80 → 44261	[ACK]	Seq=1 Ack=134 Win=65536 Len=32768 TSval=3138030144 TSecr=2209869055 [TCP segment of a reassembled PDU]
2518	62.158545965	10.20.30.51	127.0.0.1	TCP	68	44261 → 80	[ACK]	Seq=134 Ack=32769 Win=48512 Len=0 TSval=2209869067 TSecr=3138030144
2519	62.158573409	127.0.0.1	10.20.30.51	TCP	32836	80 → 44261	[PSH, ACK]	Seq=32769 Ack=134 Win=65536 Len=32768 TSval=3138030144 TSecr=2209869055 [TCP segment of a reassembled PDU]
2520	62.159839069	10.20.30.51	127.0.0.1	TCP	68	44261 → 80	[ACK]	Seq=134 Ack=65537 Win=48512 Len=0 TSval=2209869068 TSecr=3138030144
2521	62.174577611	127.0.0.1	10.20.30.51	TCP	32836	80 → 44261	[ACK]	Seq=65537 Ack=134 Win=65536 Len=32768 TSval=3138030160 TSecr=2209869068 [TCP segment of a reassembled PDU]
2522	62.174678939	10.20.30.51	127.0.0.1	TCP	68	44261 → 80	[ACK]	Seq=134 Ack=98305 Win=65536 Len=0 TSval=2209869083 TSecr=3138030160
2523	62.184676456	127.0.0.1	10.20.30.51	TCP	32836	80 → 44261	[PSH, ACK]	Seq=98305 Ack=134 Win=65536 Len=32768 TSval=3138030170 TSecr=2209869083 [TCP segment of a reassembled PDU]
2524	62.184918823	10.20.30.51	127.0.0.1	TCP	68	44261 → 80	[ACK]	Seq=134 Ack=131073 Win=196480 Len=0 TSval=2209869093 TSecr=3138030170
2525	62.184963663	127.0.0.1	10.20.30.51	TCP	32836	80 → 44261	[ACK]	Seq=131073 Ack=134 Win=65536 Len=32768 TSval=3138030170 TSecr=2209869083 [TCP segment of a reassembled PDU]
2526	62.184973136	10.20.30.51	127.0.0.1	TCP	68	44261 → 80	[ACK]	Seq=134 Ack=163841 Win=327552 Len=0 TSval=2209869093 TSecr=3138030170
2527	62.184997912	127.0.0.1	10.20.30.51	TCP	32836	80 → 44261	[PSH, ACK]	Seq=163841 Ack=134 Win=65536 Len=32768 TSval=3138030170 TSecr=2209869093 [TCP segment of a reassembled PDU]
2528	62.185003743	10.20.30.51	127.0.0.1	TCP	68	44261 → 80	[ACK]	Seq=134 Ack=196609 Win=458496 Len=0 TSval=2209869093 TSecr=3138030170
2529	62.185021303	127.0.0.1	10.20.30.51	TCP	32836	80 → 44261	[ACK]	Seq=196609 Ack=134 Win=65536 Len=32768 TSval=3138030170 TSecr=2209869093 [TCP segment of a reassembled PDU]
2530	62.185023722	10.20.30.51	127.0.0.1	TCP	68	44261 → 80	[ACK]	Seq=134 Ack=229377 Win=589440 Len=0 TSval=2209869094 TSecr=3138030170
2531	62.185044900	127.0.0.1	10.20.30.51	TCP	32836	80 → 44261	[PSH, ACK]	Seq=229377 Ack=134 Win=65536 Len=32768 TSval=3138030170 TSecr=2209869093 [TCP segment of a reassembled PDU]
2532	62.185047073	10.20.30.51	127.0.0.1	TCP	68	44261 → 80	[ACK]	Seq=134 Ack=262145 Win=720384 Len=0 TSval=2209869094 TSecr=3138030170
2533	62.185065410	127.0.0.1	10.20.30.51	TCP	32836	80 → 44261	[ACK]	Seq=262145 Ack=134 Win=65536 Len=32768 TSval=3138030170 TSecr=2209869093 [TCP segment of a reassembled PDU]
2534	62.185067410	10.20.30.51	127.0.0.1	TCP	68	44261 → 80	[ACK]	Seq=134 Ack=294913 Win=851328 Len=0 TSval=2209869094 TSecr=3138030170
2535	62.185088602	127.0.0.1	10.20.30.51	TCP	32836	80 → 44261	[PSH, ACK]	Seq=294913 Ack=134 Win=65536 Len=32768 TSval=3138030170 TSecr=2209869093 [TCP segment of a reassembled PDU]
2536	62.185091079	10.20.30.51	127.0.0.1	TCP	68	44261 → 80	[ACK]	Seq=134 Ack=327681 Win=982272 Len=0 TSval=2209869094 TSecr=3138030170

- The packets 2512 to 2514 : it's a TCP 3-Way Handshake Process.
- The packet 2515 : [200 GET /500MB.zip HTTP/1.1] The client asking the file : 500MB.zip .
- The next packets going to be the download of this file , as we can see many packets of [ACK] and [PSH,ACK].

3746	62.401869951	10.20.30.51	127.0.0.1	TCP	68	44261 → 80	[FIN, ACK]	Seq=134 Ack=52429101 Win=3112448 Len=0 TSval=2209869310 TSecr=3138030387
3747	62.401991778	127.0.0.1	10.20.30.51	TCP	68	80 → 44261	[FIN, ACK]	Seq=52429101 Ack=135 Win=65536 Len=0 TSval=3138030387 TSecr=2209869310

- The packet 3746 and 3747 : The client got the file 500MB.zip , so now he can close the connection with the flag [FIN, ACK].

The fourth file : image-site.png

3749	64.431540207	10.20.30.51	127.0.0.1	TCP	76	39031	-	80	[SYN]	Seq=0	Win=65495	Len=0	MSS=65495	SACK_PERM=1	TSval=2209871340	TSecr=0	WS=128	
3750	64.431560358	127.0.0.1	10.20.30.51	TCP	76	80	-	39031	[ACK]	Seq=0	Ack=1	Win=65483	Len=0	MSS=65495	SACK_PERM=1	TSval=3138032417	TSecr=2209871340	WS=128
3751	64.431560805	10.20.30.51	127.0.0.1	TCP	68	39031	-	80	[ACK]	Seq=1	Ack=1	Win=65536	Len=0	TSval=2209871340	TSecr=3138032417			
3752	64.431602223	10.20.30.51	127.0.0.1	HTTP	206	GET /image-site.png HTTP/1.1												
3753	64.431680643	127.0.0.1	10.20.30.51	TCP	68	80	-	39031	[ACK]	Seq=1	Ack=139	Win=65408	Len=0	TSval=3138032417	TSecr=2209871340	[TCP segment of a reassembled PDU]		
3754	64.434406360	127.0.0.1	10.20.30.51	TCP	32836	80	-	39031	[ACK]	Seq=1	Ack=139	Win=65536	Len=32768	TSval=3138032420	TSecr=2209871340	[TCP segment of a reassembled PDU]		
3755	64.434524933	10.20.30.51	127.0.0.1	TCP	68	39031	-	80	[ACK]	Seq=139	Ack=32769	Win=48512	Len=0	TSval=2209871343	TSecr=3138032420	[TCP segment of a reassembled PDU]		
3756	64.434540066	127.0.0.1	10.20.30.51	TCP	32836	80	-	39031	[PSH, ACK]	Seq=32769	Ack=139	Win=65536	Len=32768	TSval=3138032420	TSecr=2209871340	[TCP segment of a reassembled PDU]		
3757	64.435106449	10.20.30.51	127.0.0.1	TCP	68	39031	-	80	[ACK]	Seq=139	Ack=65537	Win=48512	Len=0	TSval=2209871344	TSecr=3138032420	[TCP segment of a reassembled PDU]		
3758	64.437102611	127.0.0.1	10.20.30.51	TCP	32836	80	-	39031	[ACK]	Seq=65537	Ack=139	Win=65536	Len=32768	TSval=3138032423	TSecr=2209871344	[TCP segment of a reassembled PDU]		
3759	64.437173790	10.20.30.51	127.0.0.1	TCP	68	39031	-	80	[ACK]	Seq=139	Ack=98305	Win=65536	Len=0	TSval=2209871346	TSecr=3138032423	[TCP segment of a reassembled PDU]		
3760	64.439662400	127.0.0.1	10.20.30.51	TCP	32836	80	-	39031	[PSH, ACK]	Seq=98305	Ack=139	Win=65536	Len=32768	TSval=3138032425	TSecr=2209871346	[TCP segment of a reassembled PDU]		
3761	64.439687538	10.20.30.51	127.0.0.1	TCP	68	39031	-	80	[ACK]	Seq=139	Ack=115073	Win=196480	Len=0	TSval=2209871348	TSecr=3138032425	[TCP segment of a reassembled PDU]		
3762	64.439698436	127.0.0.1	10.20.30.51	TCP	32836	80	-	39031	[ACK]	Seq=131073	Ack=139	Win=65536	Len=32768	TSval=3138032425	TSecr=2209871346	[TCP segment of a reassembled PDU]		
3763	64.439701638	10.20.30.51	127.0.0.1	TCP	68	39031	-	80	[ACK]	Seq=139	Ack=163841	Win=327552	Len=0	TSval=2209871348	TSecr=3138032425	[TCP segment of a reassembled PDU]		
3764	64.439712815	127.0.0.1	10.20.30.51	TCP	32836	80	-	39031	[PSH, ACK]	Seq=163841	Ack=139	Win=65536	Len=32768	TSval=3138032425	TSecr=2209871348	[TCP segment of a reassembled PDU]		
3765	64.439716082	10.20.30.51	127.0.0.1	TCP	68	39031	-	80	[ACK]	Seq=139	Ack=196609	Win=486496	Len=0	TSval=2209871348	TSecr=3138032425	[TCP segment of a reassembled PDU]		
3766	64.439724992	127.0.0.1	10.20.30.51	TCP	32836	80	-	39031	[ACK]	Seq=196609	Ack=139	Win=65536	Len=32768	TSval=3138032425	TSecr=2209871348	[TCP segment of a reassembled PDU]		
3767	64.439727066	10.20.30.51	127.0.0.1	TCP	68	39031	-	80	[ACK]	Seq=139	Ack=229377	Win=509440	Len=0	TSval=2209871348	TSecr=3138032425	[TCP segment of a reassembled PDU]		
3768	64.439737660	127.0.0.1	10.20.30.51	TCP	32836	80	-	39031	[PSH, ACK]	Seq=229377	Ack=139	Win=65536	Len=32768	TSval=3138032425	TSecr=2209871348	[TCP segment of a reassembled PDU]		
3769	64.439739888	10.20.30.51	127.0.0.1	TCP	68	39031	-	80	[ACK]	Seq=139	Ack=262145	Win=720384	Len=0	TSval=2209871348	TSecr=3138032425	[TCP segment of a reassembled PDU]		
3770	64.439750558	127.0.0.1	10.20.30.51	TCP	32836	80	-	39031	[ACK]	Seq=262145	Ack=139	Win=65536	Len=32768	TSval=3138032425	TSecr=2209871348	[TCP segment of a reassembled PDU]		
3771	64.439752289	10.20.30.51	127.0.0.1	TCP	68	39031	-	80	[ACK]	Seq=139	Ack=294913	Win=851328	Len=0	TSval=2209871348	TSecr=3138032425	[TCP segment of a reassembled PDU]		
3772	64.439762967	127.0.0.1	10.20.30.51	TCP	32836	80	-	39031	[PSH, ACK]	Seq=294913	Ack=139	Win=65536	Len=32768	TSval=3138032425	TSecr=2209871348	[TCP segment of a reassembled PDU]		
3773	64.439764979	10.20.30.51	127.0.0.1	TCP	68	39031	-	80	[ACK]	Seq=139	Ack=327681	Win=982272	Len=0	TSval=2209871348	TSecr=3138032425	[TCP segment of a reassembled PDU]		

- The packets 3749 to 3751 : it's a TCP 3-Way Handshake Process.
- The packet 3752 : [200 GET /image-site.png HTTP/1.1] The client asking the file : image-site.png.
- The next packets going to be the download of this file , as we can see many packets of [ACK] and [PSH,ACK].

3809	64.443336020	10.20.30.51	127.0.0.1	TCP	68	39031	-	80	[FIN, ACK]	Seq=139	Ack=1360876	Win=1712384	Len=0	TSval=2209871352	TSecr=3138032427			
3810	64.443553196	127.0.0.1	10.20.30.51	TCP	68	80	-	39031	[FIN, ACK]	Seq=1360876	Ack=140	Win=65536	Len=0	TSval=3138032429	TSecr=2209871352			

- The packet 3809 and 3810 : The client got the file 500MB.zip , so now he can close the connection with the flag [FIN, ACK].

The last file : task2.pdf

3812	66.552747427	10.20.30.51	127.0.0.1	TCP	76	35195 → 80	[SYN] Seq=0 Win=65495 Len=0 MSS=65495 SACK_PERM=1 TSval=2209873461 TSecr=0 WS=128
3813	66.552756299	127.0.0.1	10.20.30.51	TCP	76	80 → 35195	[SYN, ACK] Seq=0 Ack=1 Win=65483 Len=0 MSS=65495 SACK_PERM=1 TSval=3138034538 TSecr=2209873461 WS=128
3814	66.552777076	10.20.30.51	127.0.0.1	TCP	68	35195 → 80	[ACK] Seq=1 Ack=1 Win=65536 Len=0 TSval=2209873461 TSecr=3138034538
3815	66.552089537	10.20.30.51	127.0.0.1	HTTP	201	GET /task2.pdf HTTP/1.1	
3816	66.552844640	127.0.0.1	10.20.30.51	TCP	68	80 → 35195	[ACK] Seq=1 Ack=134 Win=65488 Len=0 TSval=3138034538 TSecr=2209873461
3817	66.555100978	127.0.0.1	10.20.30.51	HTTP	32663	HTTP/1.1 200 OK	(application/pdf)
3818	66.555124461	10.20.30.51	127.0.0.1	TCP	68	35195 → 80	[ACK] Seq=134 Ack=32596 Win=48640 Len=0 TSval=2209873464 TSecr=3138034541
3819	66.555779037	10.20.30.51	127.0.0.1	TCP	68	35195 → 80	[FIN, ACK] Seq=134 Ack=32596 Win=65536 Len=0 TSval=2209873464 TSecr=3138034541
3820	66.555966921	127.0.0.1	10.20.30.51	TCP	68	80 → 35195	[FIN, ACK] Seq=32596 Ack=135 Win=65536 Len=0 TSval=3138034541 TSecr=2209873464
3821	66.555976365	10.20.30.51	127.0.0.1	TCP	68	35195 → 80	[ACK] Seq=135 Ack=32597 Win=65536 Len=0 TSval=2209873464 TSecr=3138034541

- The packets 3812 to 3814 : it's a TCP 3-Way Handshake Process.
- The packet 3815 : [200 GET /task2.pdf HTTP/1.1] The client asking the file : task2.pdf.
- The packet 3816 : The http Server send an ACK to the Client that he got his request.
- The packet 3817 : [HTTP/1.1 200 OK ...] This packet is the server response to the previous HTTP request. "HTTP/1.1" indicates the protocol version used, "200" is the response status code: means that the request was processed successfully. The content of the requested resource (task2.pdf) is included in this package
- The packet 3818 : The Client send an ACK to the HTTP , because he got with success the answer of his request.
- The packets 3819 to 3821 : To close the TCP connection