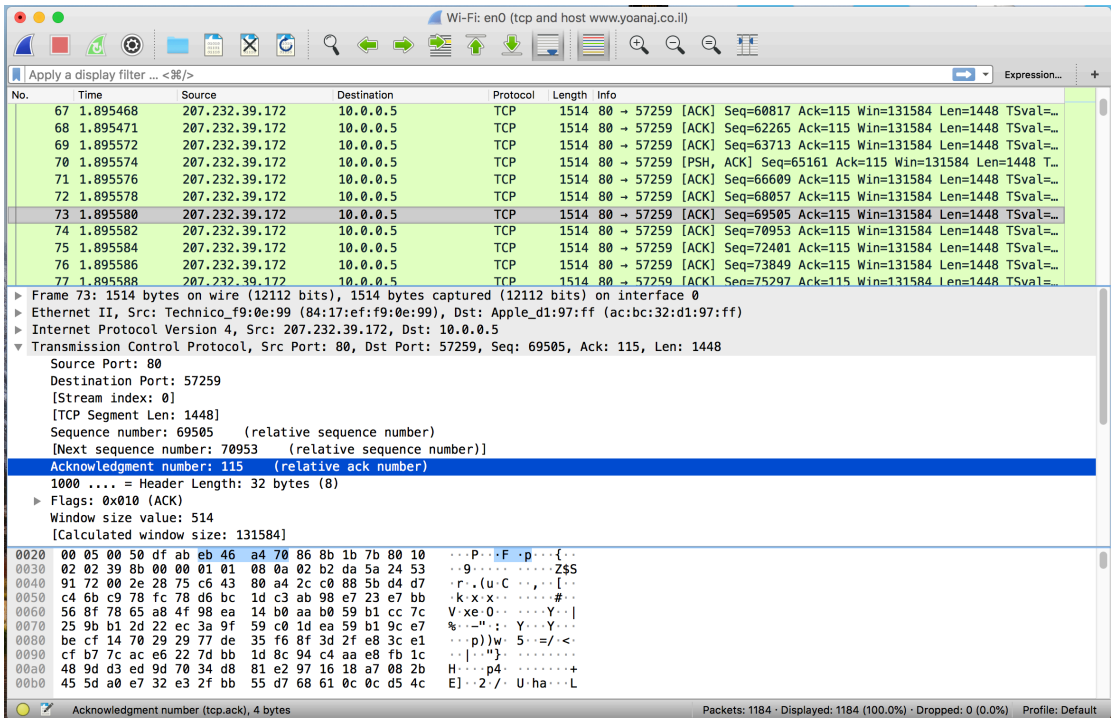
Step1：

```
lijiatongdeMacBook-Pro:~ Jessielee$ curl http://www.yoanaj.co.il/uploadimages/The_Little_Prince.pdf
> fetch
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 1057k  100 1057k    0     0   197k      0  0:00:05  0:00:05 --:--:--  243k
lijiatongdeMacBook-Pro:~ Jessielee$ curl http://www.yoanaj.co.il/uploadimages/The_Little_Prince.pdf
> fetch
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 1057k  100 1057k    0     0   247k      0  0:00:04  0:00:04 --:--:--  247k
lijiatongdeMacBook-Pro:~ Jessielee$
```



Step2：

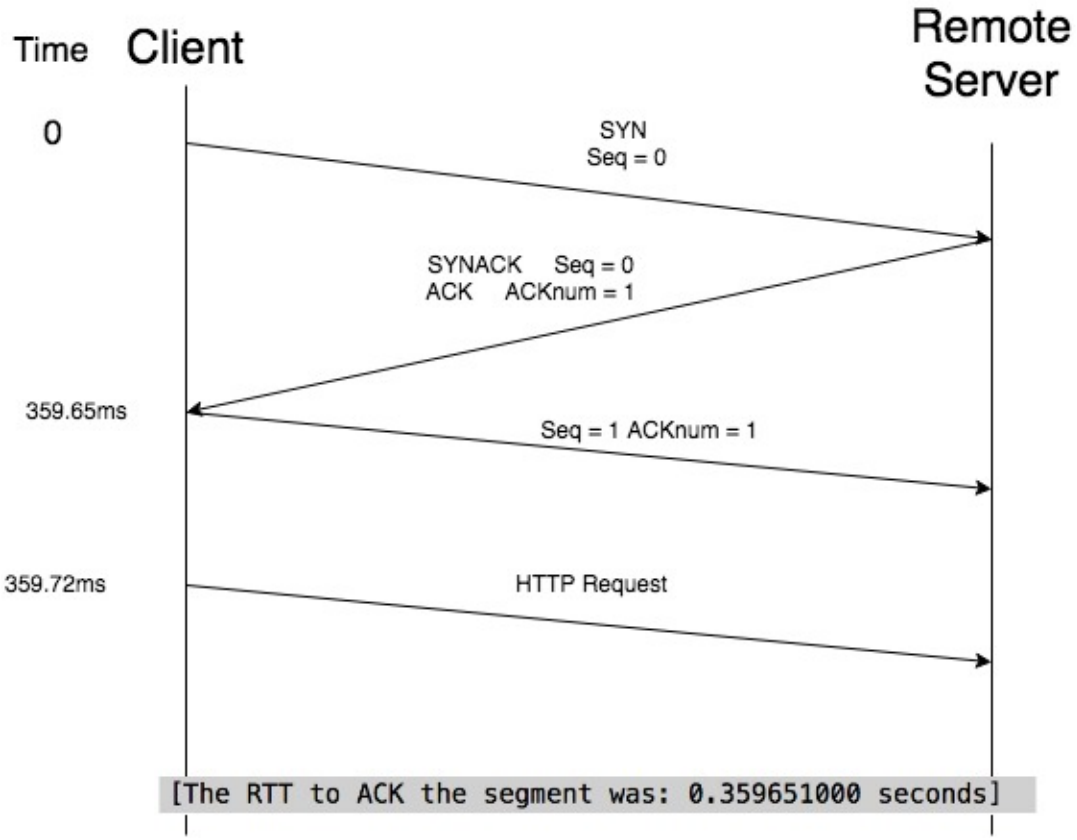
```
▼ Transmission Control Protocol, Src Port: 80, Dst Port: 57259, Seq: 69505, Ack: 115, Len: 1448
    Source Port: 80
    Destination Port: 57259
    [Stream index: 0]
    [TCP Segment Len: 1448]
    Sequence number: 69505    (relative sequence number)
    [Next sequence number: 70953    (relative sequence number)]
    Acknowledgment number: 115    (relative ack number)
    1000 .... = Header Length: 32 bytes (8)
  ▶ Flags: 0x010 (ACK)
    Window size value: 514
    [Calculated window size: 131584]
    [Window size scaling factor: 256]
    Checksum: 0x398b [unverified]
    [Checksum Status: Unverified]
    Urgent pointer: 0
  ▶ Options: (12 bytes), No-Operation (NOP), No-Operation (NOP), Timestamps
  ▶ [SEQ/ACK analysis]
  ▶ [Timestamps]
    TCP payload (1448 bytes)
    [Reassembled PDU in frame: 1168]
    TCP segment data (1448 bytes)
```

Step3：

<table>
<tr><td colspan="3">TCP Segment</td></tr>
<tr><td colspan="2">Source Port: 80</td><td>Destination Port: 57259</td></tr>
<tr><td colspan="3">Sequence Number : 69505</td></tr>
<tr><td colspan="3">Acknowledgement Number : 115</td></tr>
<tr><td>Header Length</td><td>Flag</td><td>Window Size Value</td></tr>
<tr><td colspan="2">Checksum：0x398b</td><td>Urgent</td></tr>
<tr><td colspan="3">Options</td></tr>
<tr><td colspan="3">Payload: 1448bytes</td></tr>
</table>

Step4:

Three-Way Handshake



Connection Options:

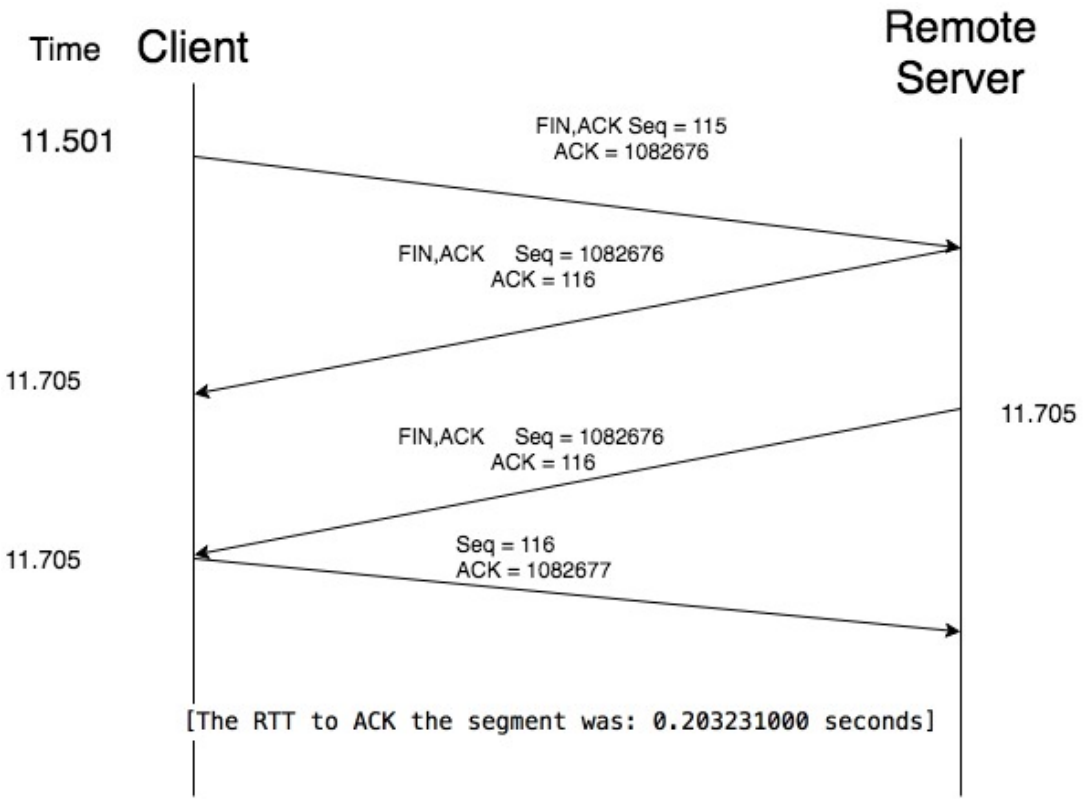
SYN packets has Maximum segment size, Window scale, Timestamps, SACK permitted.

```
▼ Options: (24 bytes), Maximum segment size, No-Operation (NOP), Window scale, No-Operation (NOP), No-Operation (NOP), Timestamps, SACK permitt
  ▶ TCP Option - Maximum segment size: 1460 bytes
  ▼ TCP Option - No-Operation (NOP)
      Kind: No-Operation (1)
  ▶ TCP Option - Window scale: 5 (multiply by 32)
  ▼ TCP Option - No-Operation (NOP)
      Kind: No-Operation (1)
  ▼ TCP Option - No-Operation (NOP)
      Kind: No-Operation (1)
  ▼ TCP Option - Timestamps: TSval 609454914, TSecr 0
      Kind: Time Stamp Option (8)
      Length: 10
      Timestamp value: 609454914
      Timestamp echo reply: 0
  ▶ TCP Option - SACK permitted
  ▼ TCP Option - End of Option List (EOL)
      Kind: End of Option List (0)
```

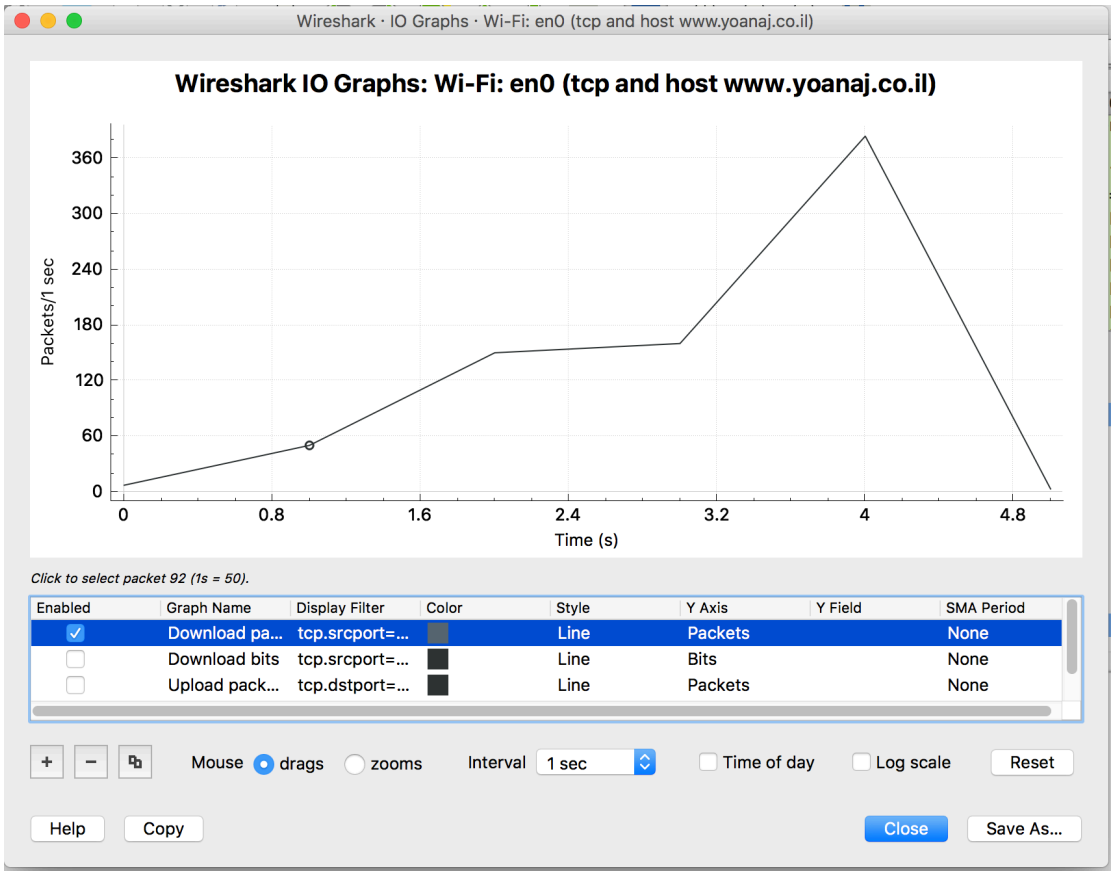
FIN/RST Teardown

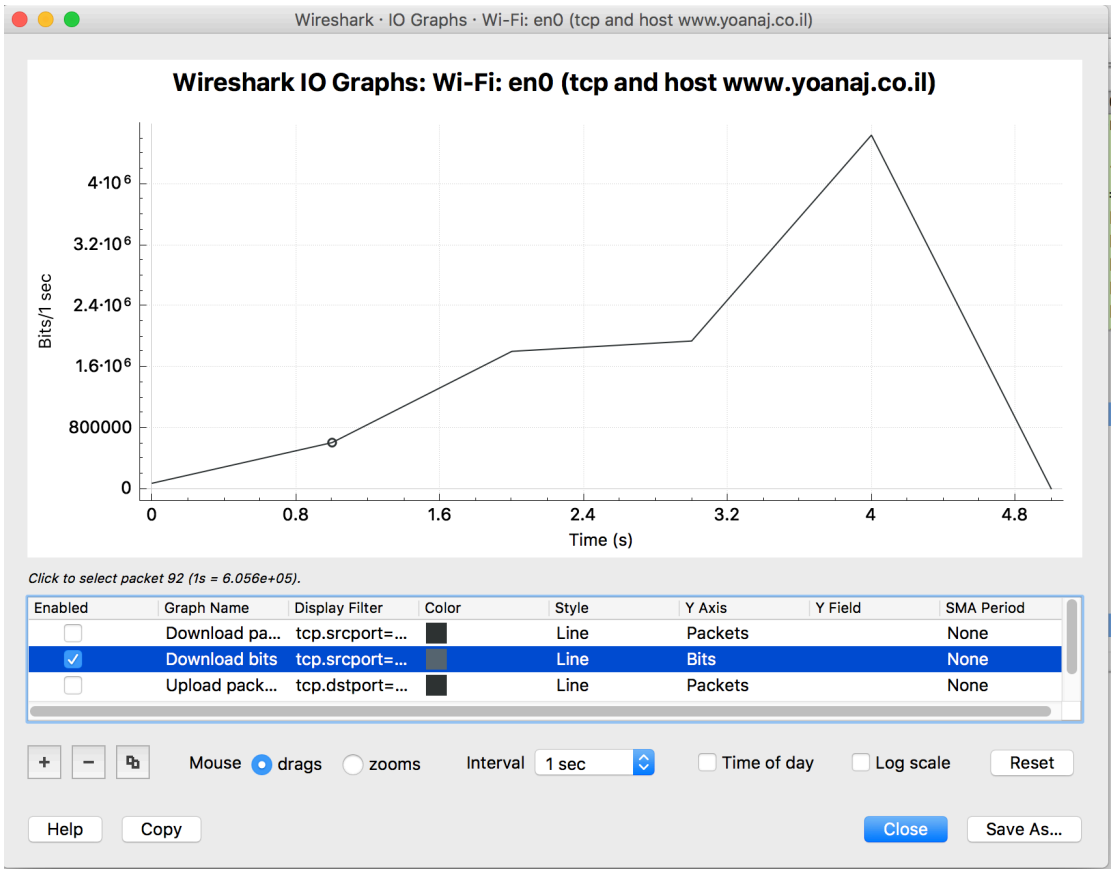
```
  1398 11.501969    10.15.71.179      207.232.39.172    TCP      66 58265 → 80 [FIN, ACK] Seq=115 Ack=1082676 Win=131072 Len=0 TSval=…
  1399 11.705200    207.232.39.172    10.15.71.179      TCP      66 80 → 58265 [ACK] Seq=1082676 Ack=116 Win=131840 Len=0 TSval=46735…
  1400 11.705205    207.232.39.172    10.15.71.179      TCP      66 80 → 58265 [FIN, ACK] Seq=1082676 Ack=116 Win=131840 Len=0 TSval=…
  1401 11.705321    10.15.71.179      207.232.39.172    TCP      66 58265 → 80 [ACK] Seq=116 Ack=1082677 Win=131072 Len=0 TSval=62122…
```

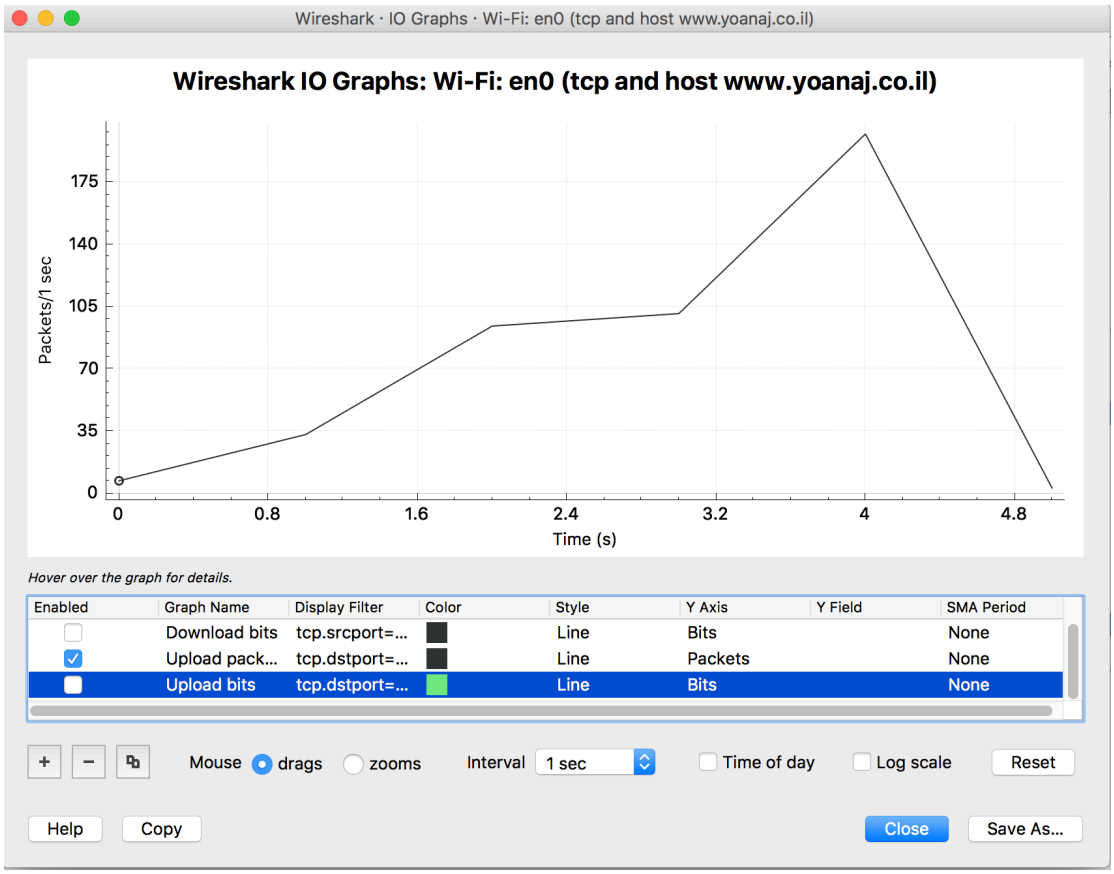
```
▶ Frame 1398: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface 0
▶ Ethernet II, Src: Apple_d1:97:ff (ac:bc:32:d1:97:ff), Dst: PaloAlto_04:b7:10 (00:86:9c:04:b7:10)
▶ Internet Protocol Version 4, Src: 10.15.71.179, Dst: 207.232.39.172
▼ Transmission Control Protocol, Src Port: 58265, Dst Port: 80, Seq: 115, Ack: 1082676, Len: 0
    Source Port: 58265
    Destination Port: 80
    [Stream index: 0]
    [TCP Segment Len: 0]
    Sequence number: 115    (relative sequence number)
    [Next sequence number: 115    (relative sequence number)]
    Acknowledgment number: 1082676    (relative ack number)
    1000 .... = Header Length: 32 bytes (8)
  ▶ Flags: 0x011 (FIN, ACK)
    Window size value: 4096
    [Calculated window size: 131072]
    [Window size scaling factor: 32]
    Checksum: 0x1c10 [unverified]
    [Checksum Status: Unverified]
    Urgent pointer: 0
▼ Options: (12 bytes), No-Operation (NOP), No-Operation (NOP), Timestamps
```

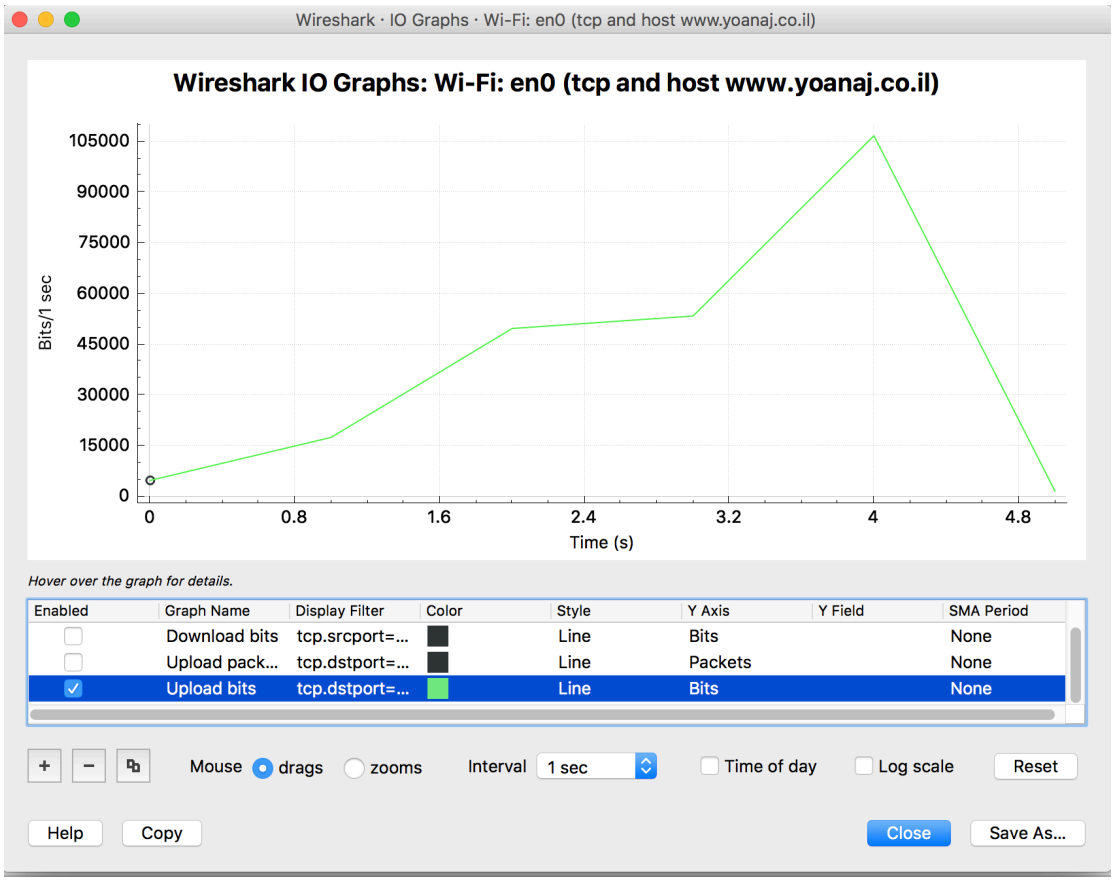


Step5:


Wireshark · IO Graphs · Wi-Fi: en0 (tcp and host www.yoanaj.co.il)
Wireshark IO Graphs: Wi-Fi: en0 (tcp and host www.yoanaj.co.il)
Packets/1 sec
360
300
240
180
120
60
0
0
0.8
1.6
2.4
3.2
4
4.8
Time (s)
Click to select packet 92 (1s = 50).
Enabled
Graph Name
Display Filter
Color
Style
Y Axis
Y Field
SMA Period
Download pa...
tcp.srcport=...
Line
Packets
None
Download bits
tcp.srcport=...
Line
Bits
None
Upload pack...
tcp.dstport=...
Line
Packets
None
Mouse
drags
zooms
Interval
1 sec
Time of day
Log scale
Reset
Help
Copy
Close
Save As...


Wireshark · IO Graphs · Wi-Fi: en0 (tcp and host www.yoanaj.co.il)
Wireshark IO Graphs: Wi-Fi: en0 (tcp and host www.yoanaj.co.il)
Bits/1 sec
4·10⁶
3.2·10⁶
2.4·10⁶
1.6·10⁶
800000
0
0
0.8
1.6
2.4
3.2
4
4.8
Time (s)
Click to select packet 92 (1s = 6.056e+05).
Enabled
Graph Name
Display Filter
Color
Style
Y Axis
Y Field
SMA Period
Download pa...
tcp.srcport=...
Line
Packets
None
Download bits
tcp.srcport=...
Line
Bits
None
Upload pack...
tcp.dstport=...
Line
Packets
None
Mouse
drags
zooms
Interval
1 sec
Time of day
Log scale
Reset
Help
Copy
Close
Save As...





1. Answer the following questions to show your understanding of the data transfer: 1. What is the rough

data rate in the download direction in packets/second and bits/second once the TCP connection is running well?

300 packtes/second && 3200000 bits/second

2. What percentage of this download rate is content? Show your calculation. To find out, look at a typical download packet; there should be many similar, large download packets. You can see how long it is, and how many bytes of TCP payload it contains.

The download packages are 1440 bytes long and 1374 bytes are the TCP payload carring contents.

The percentage would be 1374/1440 = 95.41%.

3. What is the rough data rate in the upload direction in packets/second and bits/second due to the ACK packets?

100packets/second && 105000bits/second

4. If the most recently received TCP segment from the server has a sequence number of X, then what ACK number does the next transmitted TCP segment carry?

The ACK number tells the next expected sequence number. Thus it will be X plus the number of TCP payload bytes in the data segment.