Entrega: 23/11/2023

Alumnes: Francesco Oncins Spedo
Mariona Farré Tapias
Pau Alcázar Perdomo

INTERNET:
P2- Qüestionari sessió 2 - Anàlisis dels protocols Telnet i FTP

Preguntes telnet

1.
Un servidor permet més d'una connexió de clients, mentre aquest tingui multiples ports oberts i actius per establir més d'una connexió de diferents clients.
Si fem un ss -na per veure quins ports poden tenir connexions LISTEN o els que no estan activats amb UNCONN (els protocols UDP) en la MV:

```
udp    UNCONN     0      0                *:631                            *:*
udp    UNCONN     0      0                *:53221                          *:*
udp    UNCONN     0      0                *:5353                           *:*
udp    UNCONN     0      0      192.168.60.201:53                          *:*
udp    UNCONN     0      0           127.0.0.1:53                          *:*
udp    UNCONN     0      0                *:68                             *:*
udp    UNCONN     0      0               :::47839                         :::*
udp    UNCONN     0      0               :::5353                          :::*
udp    UNCONN     0      0               :::53                            :::*
tcp    LISTEN     0      10     192.168.60.201:53                          *:*
tcp    LISTEN     0      10          127.0.0.1:53                          *:*
tcp    LISTEN     0      128              *:22                             *:*
tcp    LISTEN     0      5           127.0.0.1:631                         *:*
tcp    LISTEN     0      128              *:23                             *:*
tcp    LISTEN     0      100              *:25                             *:*
tcp    LISTEN     0      128         127.0.0.1:953                         *:*
tcp    ESTAB      0      0      192.168.60.201:23          192.168.60.197:60534
tcp    LISTEN     0      128             :::80                            :::*
tcp    LISTEN     0      10              :::53                            :::*
tcp    LISTEN     0      32              :::21                            :::*
tcp    LISTEN     0      128             :::22                            :::*
tcp    LISTEN     0      5              ::1:631                           :::*
tcp    LISTEN     0      100             :::25                            :::*
tcp    LISTEN     0      128            ::1:953                           :::*
entel@debian95-INTE-server:~$
```

Hi han varies connexions ESTAB(ESTABLISHED) i LISTENING verificant que aquest servidor pot acceptar múltiples clients alhora.

2. tcp.flags.syn==1
Des de wireshark es poden posar diferents filtres per només mostrar els paquets que creiem necessaris.
Per l'establiment d'una connexió TCP podem buscar-ho a través de filtrar les ip d'origen i destí:
ip.addr==192.168.60.201
ip.addr==192.168.60.197
Posar al apartat de filtres amb l'operació lògica ||
Si es vol filtrar també per protocol, posar el nom d'aquest seguit de l'operació lògica &&
Nosaltres hem filtrat fent:

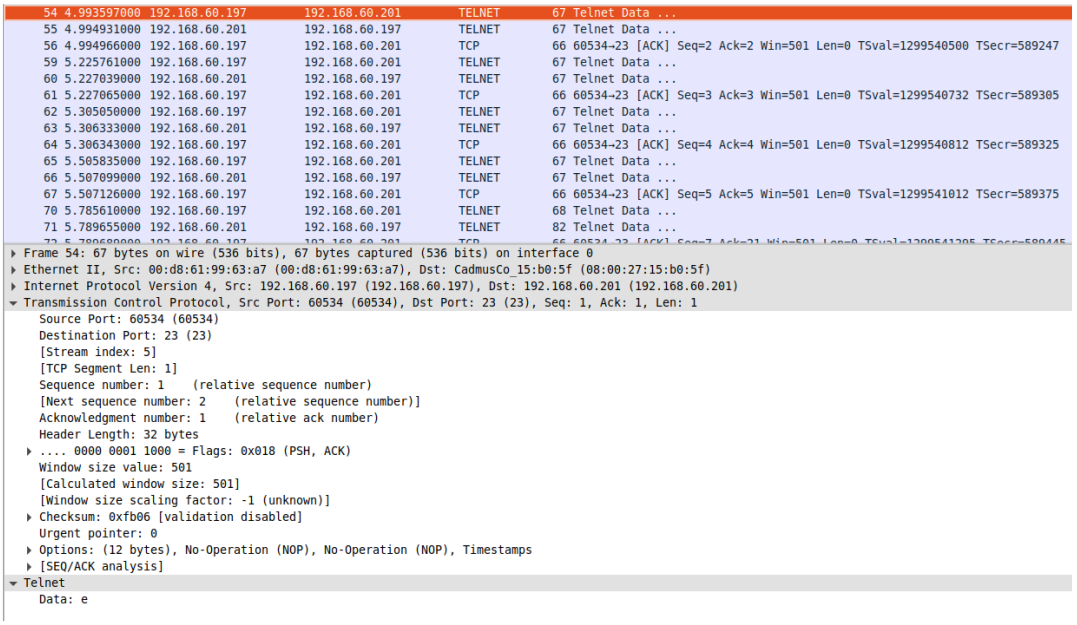
telnet && (ip.addr==192.168.60.201 || ip.addr==192.168.60.197)

3.

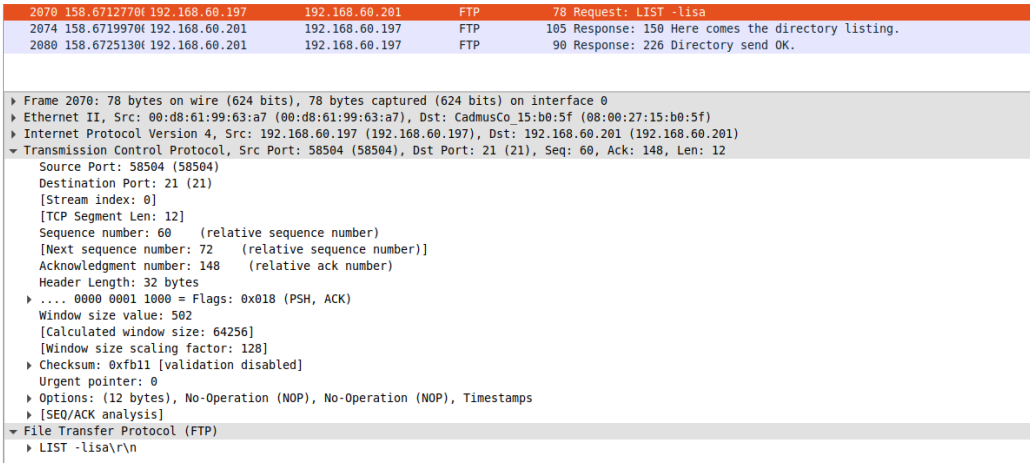
La utilització de la mida màxima del camp de dades en una connexió de xarxa depèn de diversos factors, incloent el protocol utilitzat, la configuració de la xarxa i el tipus de dades que s'estan transmetent.

- En el cas del Telnet, que és un protocol orientat a text, és poc probable que s'aprofiti la mida màxima del camp de dades en cada paquet, ja que les transmissions de text solen ser relativament petites.
- En canvi, amb el FTP, especialment durant la transferència de fitxers, és més probable que s'aprofitin els camps de dades més grans per optimitzar l'eficiència de la transferència.

Paquets TELNET: transferint un sol caràcter “e”

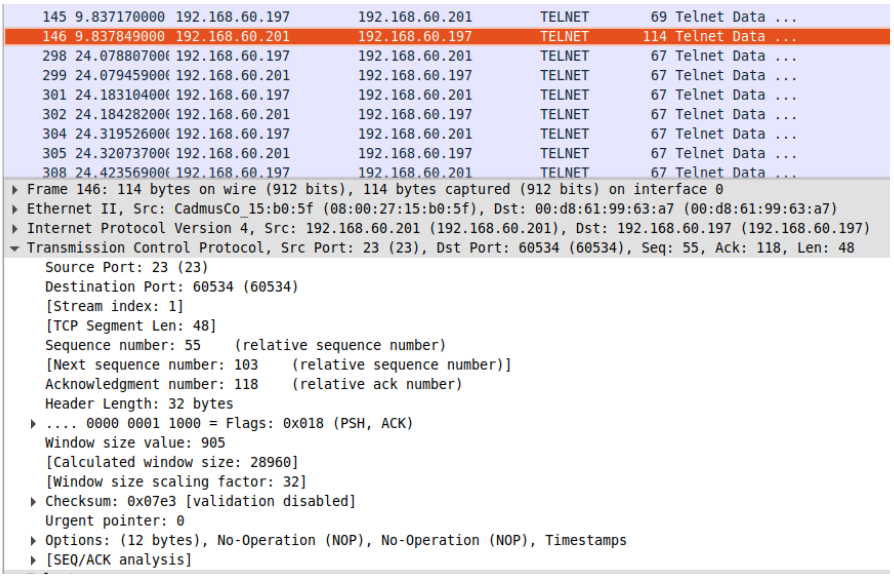


Paquets FTP: transferint la frase “-lisa\r\m” (instrucció d’ensenyar un llistat)



4.

El Wireshark és una aplicació d'anàlisi de la xarxa on es poden visualitzar tots els paquets IP que són enviats i rebuts. Aquesta aplicació per simplificar els números de seqüència associats a cada paquet inicialitza la seqüència de numeració a cada inicialització del programa, ja que si s'utilitzéssim els números de seqüència dels paquets originals IP, seria molt més difícil de comprovar el seu correcte funcionament.



Es poden veure els números de seqüència associats per l'aplicació Wireshark, dins de cada paquet a l'apartat dels paràmetres de TCP.

En aquest cas els números són:

Sequence number: 55 (el nombre de seqüència donat per Wireshark [1…55])

Next sequence number: 103 (el següent nombre de seqüència calculat 55+48 (longitud del paquet) =103)

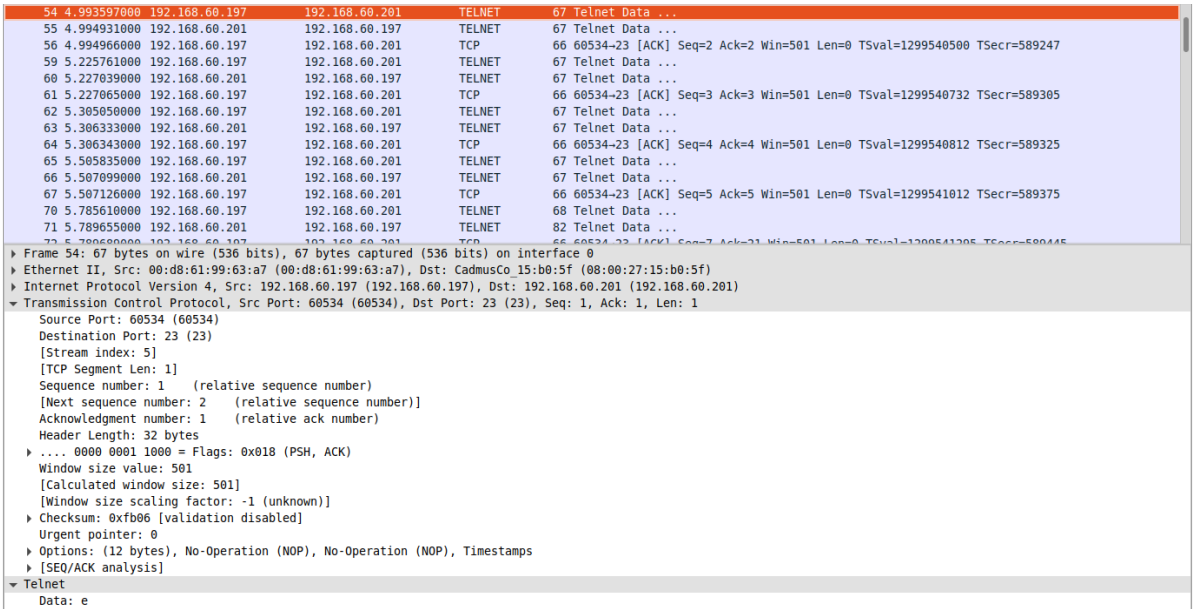
Acknowledgment number: 118

Tots recalcant que seran números relatius a l'aplicació de Wireshark

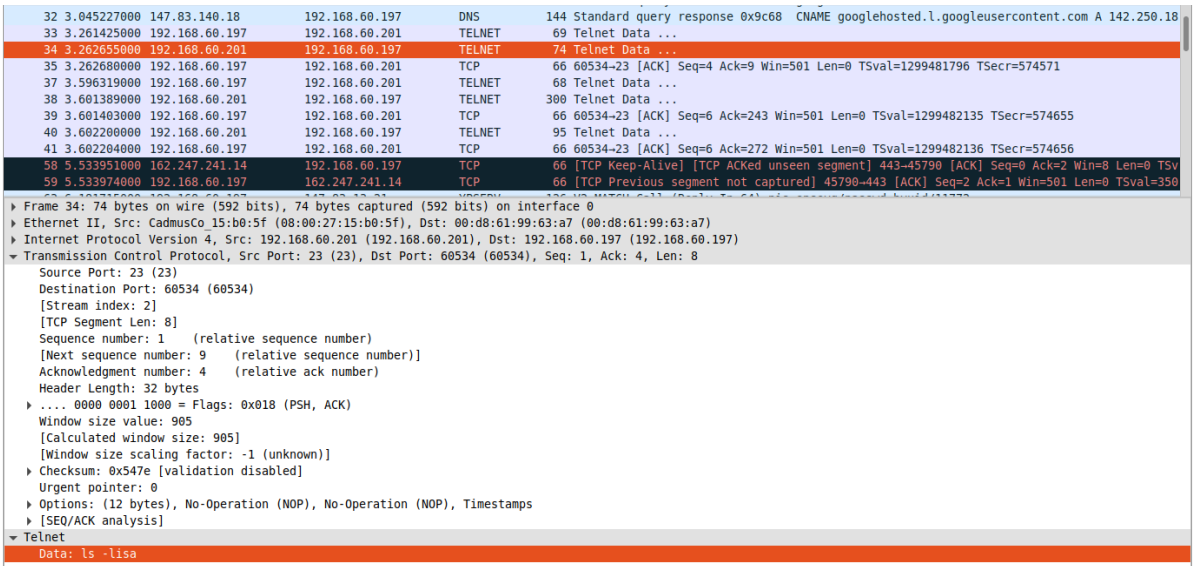
5.

En una connexió telnet, les comandes per consola s'envien directament des del client al servidor a través de la xarxa, caràcter per caràcter o línia per línia, i s'executen al servidor com si s'introduïssin directament a la consola del servidor. Aleshores, el servidor envia la sortida corresponent al client, creant una sessió interactiva. Sense xifrar.

Podem veure com aquí la dada enviada és un caràcter, en aquest cas, un caràcter de la contrasenya "entel":

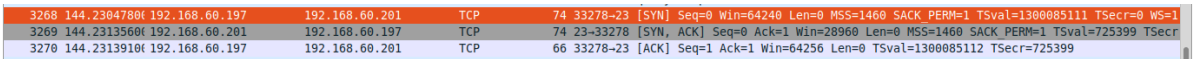


En aquest altre cas podem observar com el que s'envia és tota la comanda sencera de ls-lisa:



6. (0p) estats: listen, established,...

Quan s'estableix connexió es passa per 3 estats:



Paquet SYN: Comença client vol establir connexió a un servidor (VM) enviant un paquet TCP amb la flag SYN marcada.

```
3268 144.230478000 192.168.60.197 192.168.60.201 TCP 74 33278→23 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=1300085111 TSecr=0 WS=128
▸ Frame 3268: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface 0
▸ Ethernet II, Src: 00:d8:61:99:63:a7 (00:d8:61:99:63:a7), Dst: CadmusCo_15:b0:5f (08:00:27:15:b0:5f)
▸ Internet Protocol Version 4, Src: 192.168.60.197 (192.168.60.197), Dst: 192.168.60.201 (192.168.60.201)
▾ Transmission Control Protocol, Src Port: 33278 (33278), Dst Port: 23 (23), Seq: 0, Len: 0
    Source Port: 33278 (33278)
    Destination Port: 23 (23)
    [Stream index: 19]
    [TCP Segment Len: 0]
    Sequence number: 0    (relative sequence number)
    Acknowledgment number: 0
    Header Length: 40 bytes
  ▾ .... 0000 0000 0010 = Flags: 0x002 (SYN)
      000. .... .... = Reserved: Not set
      ...0 .... .... = Nonce: Not set
      .... 0... .... = Congestion Window Reduced (CWR): Not set
      .... .0.. .... = ECN-Echo: Not set
      .... ..0. .... = Urgent: Not set
      .... ...0 .... = Acknowledgment: Not set
      .... .... 0... = Push: Not set
      .... .... .0.. = Reset: Not set
    ▸ .... .... ..1. = Syn: Set
      .... .... ...0 = Fin: Not set
    Window size value: 64240
    [Calculated window size: 64240]
  ▸ Checksum: 0xfb0d [validation disabled]
    Urgent pointer: 0
  ▾ Options: (20 bytes), Maximum segment size, SACK permitted, Timestamps, No-Operation (NOP), Window scale
    ▸ Maximum segment size: 1460 bytes
    ▸ TCP SACK Permitted Option: True
    ▸ Timestamps: TSval 1300085111, TSecr 0
    ▸ No-Operation (NOP)
    ▸ Window scale: 7 (multiply by 128)
```

Paquet SYN-ACK: La resposta servidor del paquet SYN d'un possible client , reconeixent la petició SYN i acceptant establir una connexió amb les flags SYN i ACK marcades.

```
3269 144.231356000 192.168.60.201 192.168.60.197 TCP 74 23→33278 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460 SACK_PERM=1 TSval=725399 TSecr=1300085111 WS=32
▸ Frame 3269: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface 0
▸ Ethernet II, Src: CadmusCo_15:b0:5f (08:00:27:15:b0:5f), Dst: 00:d8:61:99:63:a7 (00:d8:61:99:63:a7)
▸ Internet Protocol Version 4, Src: 192.168.60.201 (192.168.60.201), Dst: 192.168.60.197 (192.168.60.197)
▾ Transmission Control Protocol, Src Port: 23 (23), Dst Port: 33278 (33278), Seq: 0, Ack: 1, Len: 0
    Source Port: 23 (23)
    Destination Port: 33278 (33278)
    [Stream index: 19]
    [TCP Segment Len: 0]
    Sequence number: 0    (relative sequence number)
    Acknowledgment number: 1    (relative ack number)
    Header Length: 40 bytes
  ▾ .... 0000 0001 0010 = Flags: 0x012 (SYN, ACK)
      000. .... .... = Reserved: Not set
      ...0 .... .... = Nonce: Not set
      .... 0... .... = Congestion Window Reduced (CWR): Not set
      .... .0.. .... = ECN-Echo: Not set
      .... ..0. .... = Urgent: Not set
      .... ...1 .... = Acknowledgment: Set
      .... .... 0... = Push: Not set
      .... .... .0.. = Reset: Not set
    ▸ .... .... ..1. = Syn: Set
      .... .... ...0 = Fin: Not set
    Window size value: 28960
    [Calculated window size: 28960]
  ▸ Checksum: 0x56cc [validation disabled]
    Urgent pointer: 0
  ▾ Options: (20 bytes), Maximum segment size, SACK permitted, Timestamps, No-Operation (NOP), Window scale
    ▸ Maximum segment size: 1460 bytes
    ▸ TCP SACK Permitted Option: True
    ▸ Timestamps: TSval 725399, TSecr 1300085111
    ▸ No-Operation (NOP)
    ▸ Window scale: 5 (multiply by 32)
  ▸ [SEQ/ACK analysis]
```

Paquet ACK: l'últim pas per establir connexió, resposta del client al SYN-ACK, acceptant i reconeixent la connexió amb el servidor amb la flag ACK marcada.

```
3270 144.231391000 192.168.60.197 192.168.60.201 TCP 66 33278→23 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=1300085112 TSecr=725399
▸ Frame 3270: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface 0
▸ Ethernet II, Src: 00:d8:61:99:63:a7 (00:d8:61:99:63:a7), Dst: CadmusCo_15:b0:5f (08:00:27:15:b0:5f)
▸ Internet Protocol Version 4, Src: 192.168.60.197 (192.168.60.197), Dst: 192.168.60.201 (192.168.60.201)
▾ Transmission Control Protocol, Src Port: 33278 (33278), Dst Port: 23 (23), Seq: 1, Ack: 1, Len: 0
    Source Port: 33278 (33278)
    Destination Port: 23 (23)
    [Stream index: 19]
    [TCP Segment Len: 0]
    Sequence number: 1    (relative sequence number)
    Acknowledgment number: 1    (relative ack number)
    Header Length: 32 bytes
  ▾ .... 0000 0001 0000 = Flags: 0x010 (ACK)
      000. .... .... = Reserved: Not set
      ...0 .... .... = Nonce: Not set
      .... 0... .... = Congestion Window Reduced (CWR): Not set
      .... .0.. .... = ECN-Echo: Not set
      .... ..0. .... = Urgent: Not set
      .... ...1 .... = Acknowledgment: Set
      .... .... 0... = Push: Not set
      .... .... .0.. = Reset: Not set
      .... .... ..0. = Syn: Not set
      .... .... ...0 = Fin: Not set
    Window size value: 502
    [Calculated window size: 64256]
    [Window size scaling factor: 128]
  ▸ Checksum: 0xfb05 [validation disabled]
    Urgent pointer: 0
  ▾ Options: (12 bytes), No-Operation (NOP), No-Operation (NOP), Timestamps
    ▸ No-Operation (NOP)
    ▸ No-Operation (NOP)
    ▸ Timestamps: TSval 1300085112, TSecr 725399
  ▸ [SEQ/ACK analysis]
```

I per tancar la connexió es passa per 4 estats:

```
17824 701.2584010( 192.168.60.201   192.168.60.197   TELNET   82 Telnet Data ...
17825 701.2584310( 192.168.60.197   192.168.60.201   TCP      66 33278→23 [ACK] Seq=136 Ack=623 Win=64128 Len=0 TSval=1300642139 TSecr=864655
17826 701.3145690( 192.168.60.201   192.168.60.197   TCP      66 23→33278 [FIN, ACK] Seq=623 Ack=136 Win=28960 Len=0 TSval=864669 TSecr=1300642139
17827 701.3147230( 192.168.60.197   192.168.60.201   TCP      66 33278→23 [FIN, ACK] Seq=136 Ack=624 Win=64128 Len=0 TSval=1300642195 TSecr=864669
17828 701.3149480( 192.168.60.201   192.168.60.197   TCP      66 23→33278 [ACK] Seq=624 Ack=137 Win=28960 Len=0 TSval=864670 TSecr=1300642195
```

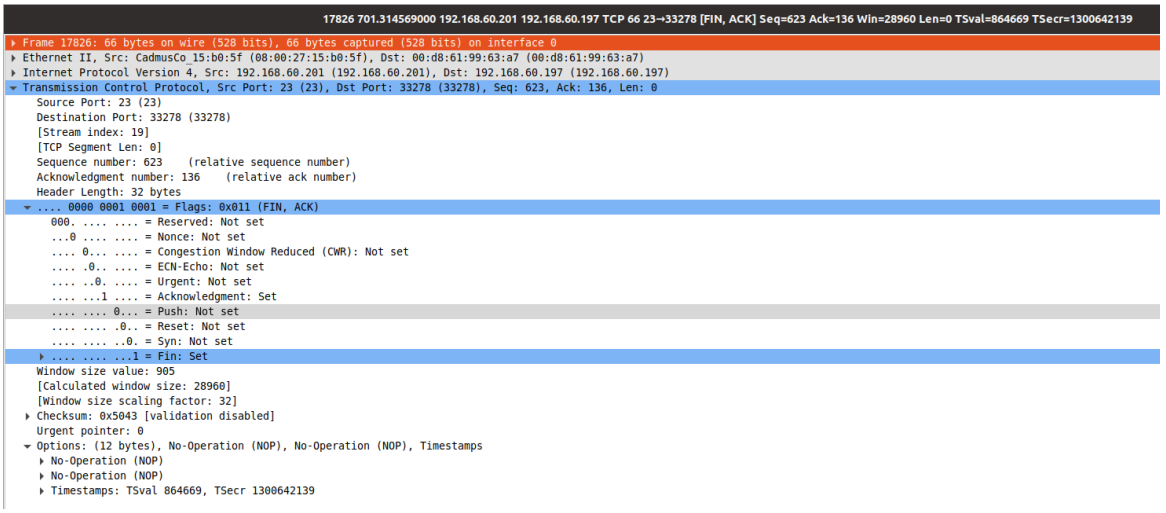
Telnet de desconnexió: el client envia una instrucció de desconnexió al servidor (VM)

```
17824 701.258401000 192.168.60.201 192.168.60.197 TELNET 82 Telnet Data ...
▸ Frame 17824: 82 bytes on wire (656 bits), 82 bytes captured (656 bits) on interface 0
▸ Ethernet II, Src: CadmusCo_15:b0:5f (08:00:27:15:b0:5f), Dst: 00:d8:61:99:63:a7 (00:d8:61:99:63:a7)
▸ Internet Protocol Version 4, Src: 192.168.60.201 (192.168.60.201), Dst: 192.168.60.197 (192.168.60.197)
▾ Transmission Control Protocol, Src Port: 23 (23), Dst Port: 33278 (33278), Seq: 607, Ack: 136, Len: 16
    Source Port: 23 (23)
    Destination Port: 33278 (33278)
    [Stream index: 19]
    [TCP Segment Len: 16]
    Sequence number: 607    (relative sequence number)
    [Next sequence number: 623    (relative sequence number)]
    Acknowledgment number: 136    (relative ack number)
    Header Length: 32 bytes
  ▾ .... 0000 0001 1000 = Flags: 0x018 (PSH, ACK)
      000. .... .... = Reserved: Not set
      ...0 .... .... = Nonce: Not set
      .... 0... .... = Congestion Window Reduced (CWR): Not set
      .... .0.. .... = ECN-Echo: Not set
      .... ..0. .... = Urgent: Not set
      .... ...1 .... = Acknowledgment: Set
      .... .... 1... = Push: Set
      .... .... .0.. = Reset: Not set
      .... .... ..0. = Syn: Not set
      .... .... ...0 = Fin: Not set
    Window size value: 905
    [Calculated window size: 28960]
    [Window size scaling factor: 32]
  ▸ Checksum: 0x4480 [validation disabled]
    Urgent pointer: 0
  ▾ Options: (12 bytes), No-Operation (NOP), No-Operation (NOP), Timestamps
    ▸ No-Operation (NOP)
    ▸ No-Operation (NOP)
    ▸ Timestamps: TSval 864655, TSecr 1300642135
  ▸ [SEQ/ACK analysis]
▾ Telnet
    Data: \r\n
    Data: desconnexi\357\277\275\357\277\275\r\n
```

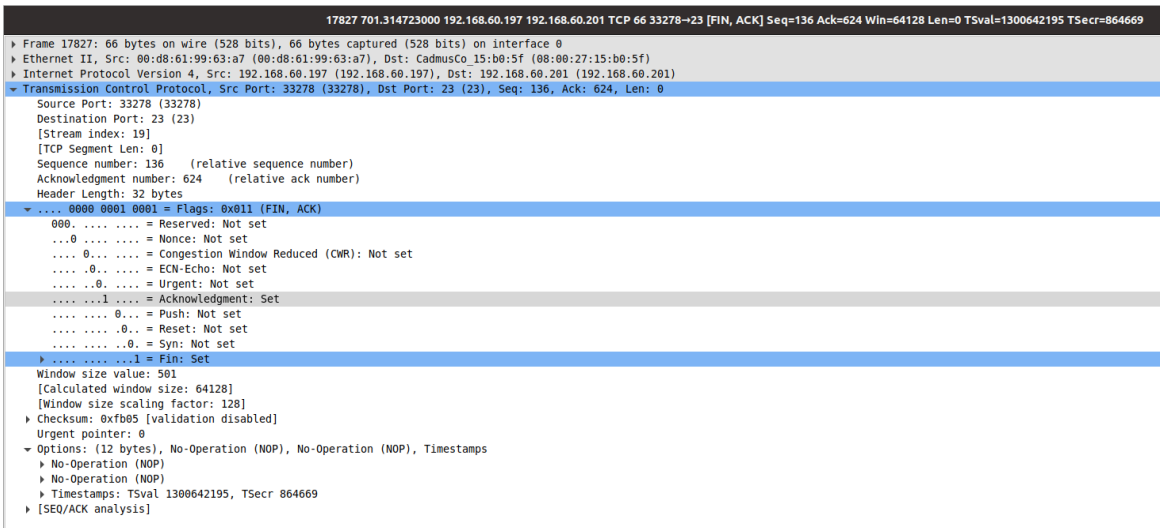
ACK del telnet: Al servidor reconeix el missatge de desconnexió del client i envia un missatge de reconeixement de desconnexió amb la flag ACK marcada.

```
17825 701.258431000 192.168.60.197 192.168.60.201 TCP 66 33278→23 [ACK] Seq=136 Ack=623 Win=64128 Len=0 TSval=1300642139 TSecr=864655
▸ Frame 17825: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface 0
▸ Ethernet II, Src: 00:d8:61:99:63:a7 (00:d8:61:99:63:a7), Dst: CadmusCo_15:b0:5f (08:00:27:15:b0:5f)
▸ Internet Protocol Version 4, Src: 192.168.60.197 (192.168.60.197), Dst: 192.168.60.201 (192.168.60.201)
▾ Transmission Control Protocol, Src Port: 33278 (33278), Dst Port: 23 (23), Seq: 136, Ack: 623, Len: 0
    Source Port: 33278 (33278)
    Destination Port: 23 (23)
    [Stream index: 19]
    [TCP Segment Len: 0]
    Sequence number: 136    (relative sequence number)
    Acknowledgment number: 623    (relative ack number)
    Header Length: 32 bytes
  ▾ .... 0000 0001 0000 = Flags: 0x010 (ACK)
      000. .... .... = Reserved: Not set
      ...0 .... .... = Nonce: Not set
      .... 0... .... = Congestion Window Reduced (CWR): Not set
      .... .0.. .... = ECN-Echo: Not set
      .... ..0. .... = Urgent: Not set
      .... ...1 .... = Acknowledgment: Set
      .... .... 0... = Push: Not set
      .... .... .0.. = Reset: Not set
      .... .... ..0. = Syn: Not set
      .... .... ...0 = Fin: Not set
    Window size value: 501
    [Calculated window size: 64128]
    [Window size scaling factor: 128]
  ▸ Checksum: 0xfb05 [validation disabled]
    Urgent pointer: 0
  ▾ Options: (12 bytes), No-Operation (NOP), No-Operation (NOP), Timestamps
    ▸ No-Operation (NOP)
    ▸ No-Operation (NOP)
    ▸ Timestamps: TSval 1300642139, TSecr 864655
  ▸ [SEQ/ACK analysis]
```

FIN ACK 1: el client reconeix el ACK del servidor, retornant-li un paquet de reconeixement de desconnexió amb el flag FIN i ACK marcada



FIN ACK 2: el servidor al rebre el ACK de finalització del client, li retorna un últim missatge de reconeixement, acabant la connexió amb la flag ACK i FIN marcades.



Preguntes FTP

7. 2 connexions, la de dades es crea cada cop. I tmb hi ha la de control (port21)

Podem mirar les connexions actives d'un servidor d'una màquina virtual executant: ss -an
Ens retornarà una llista de totes les connexions amb els seus ports actius i oberts.
Connexions tcp sense FTP:

```
tcp    LISTEN     0      10     192.168.60.201:53                    *:*
tcp    LISTEN     0      10         127.0.0.1:53                    *:*
tcp    LISTEN     0      128              *:22                    *:*
tcp    LISTEN     0      5          127.0.0.1:631                   *:*
tcp    LISTEN     0      128              *:23                    *:*
tcp    LISTEN     0      100              *:25                    *:*
tcp    LISTEN     0      128        127.0.0.1:953                   *:*
tcp    LISTEN     0      128             :::80                   :::*
tcp    LISTEN     0      10              :::53                   :::*
tcp    LISTEN     0      32              :::21                   :::*
tcp    LISTEN     0      128             :::22                   :::*
tcp    LISTEN     0      5              ::1:631                  :::*
tcp    LISTEN     0      100             :::25                   :::*
tcp    LISTEN     0      128            ::1:953                  :::*
root@debian95-INTE-server:~#
```

Quan fem la connexió ftp podem veure que hi ha 1 connexió establerta, des de 192.168.69.3 que és qui ha fet el ftp en el port 21:

```
tcp    LISTEN     0      10     192.168.60.201:53                    *:*
tcp    LISTEN     0      10         127.0.0.1:53                    *:*
tcp    LISTEN     0      128              *:22                    *:*
tcp    LISTEN     0      5          127.0.0.1:631                   *:*
tcp    LISTEN     0      128              *:23                    *:*
tcp    LISTEN     0      100              *:25                    *:*
tcp    LISTEN     0      128        127.0.0.1:953                   *:*
tcp    LISTEN     0      128             :::80                   :::*
tcp    LISTEN     0      10              :::53                   :::*
tcp    LISTEN     0      32              :::21                   :::*
tcp    LISTEN     0      128             :::22                   :::*
tcp    LISTEN     0      5              ::1:631                  :::*
tcp    LISTEN     0      100             :::25                   :::*
tcp    LISTEN     0      128            ::1:953                  :::*
tcp    ESTAB      0      0       ::ffff:192.168.60.201:21        ::ffff:192.168.60.3:416
70
```

8.

màquina1: 192.168.60.3 és la màquina que fa FTP (client)
màquina2: 192.168.60.201 és l'adreça de la màquina virtual (servidor)

```
  8 1.827230000 192.168.60.3      192.168.60.201    TCP        74 33730→21 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=2203975240 TSecr=0 WS=128
  9 1.828023000 192.168.60.201    192.168.60.3      TCP        74 21→33730 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460 SACK_PERM=1 TSval=238808 TSecr=2203975240 WS=32
 10 1.828065000 192.168.60.3      192.168.60.201    TCP        66 33730→21 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=2203975241 TSecr=238808
 11 1.831654000 192.168.60.201    192.168.60.3      FTP        86 Response: 220 (vsFTPd 3.0.3)
 12 1.831689000 192.168.60.3      192.168.60.201    TCP        66 33730→21 [ACK] Seq=1 Ack=21 Win=64256 Len=0 TSval=2203975245 TSecr=238809
 19 6.608021000 192.168.60.3      192.168.60.201    FTP        78 Request: USER entel
 20 6.608851000 192.168.60.201    192.168.60.3      TCP        66 21→33730 [ACK] Seq=21 Ack=13 Win=28960 Len=0 TSval=240004 TSecr=2203980021
 21 6.608852000 192.168.60.201    192.168.60.3      FTP       100 Response: 331 Please specify the password.
 22 6.608876000 192.168.60.3      192.168.60.201    TCP        66 33730→21 [ACK] Seq=13 Ack=55 Win=64256 Len=0 TSval=2203980022 TSecr=240004
 30 10.08751000( 192.168.60.3      192.168.60.201    FTP        78 Request: PASS letne
 31 10.10875200( 192.168.60.201    192.168.60.3      FTP        89 Response: 230 Login successful.
 32 10.10878000( 192.168.60.3      192.168.60.201    TCP        66 33730→21 [ACK] Seq=25 Ack=78 Win=64256 Len=0 TSval=2203983522 TSecr=240879
 33 10.10888000( 192.168.60.3      192.168.60.201    FTP        72 Request: SYST
 34 10.10952700( 192.168.60.201    192.168.60.3      FTP        85 Response: 215 UNIX Type: L8
 35 10.15135500( 192.168.60.3      192.168.60.201    TCP        66 33730→21 [ACK] Seq=31 Ack=97 Win=64256 Len=0 TSval=2203983565 TSecr=240879
```

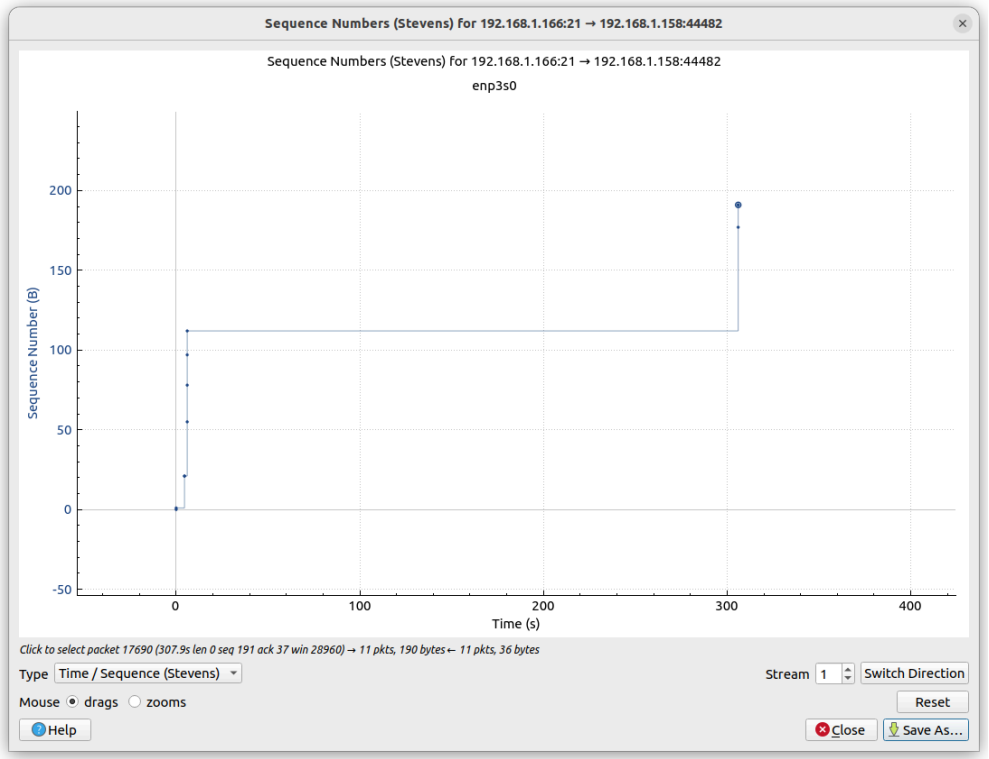
Podem veure com a l'hora de demanar la connexió ho fa la màquina1 amb el SYN (protocol: TCP), llavors la màquina2 li diu que "vale" ACK (protocol: TCP) i li envia ella també un SYN (protocol: TCP) i la màquina1 li respon amb ACK (protocol: TCP).
Ara la màquina2 envia una resposta (protocol: FTP) de la connexió FTP al port 21 de l'adreça 192.168.60.201, la màquina1 li respon un ACK (protocol: TCP).

La màquina1 li envia un request (protocol: FTP) amb el nom d'usuari i la màquina2 li respon amb un ACK (protocol: TCP).
La màquina2 li demana que especifiqui el password de l'usuari (protocol: FTP), la màquina1 li envia un ACK (protocol: TCP) i seguidament li envia el password, que podem veure "letne".
La màquina2 li envia un response (protocol: FTP) dient que s'ha pogut fer el login i la màquina1 li envia un ACK (protocol: TCP).
La màquina1 envia un últim request (protocol: FTP) del sistema, la màquina2 li respon amb l'accés a la màquina "ella mateixa" (protocol: FTP).
I per últim la màquina1 li respon amb un ACK (protocol: TCP).

Totes 2 màquines s'encarreguen d'establir la connexió de dades, però la màquina que inicia la connexió és la màquina1 i la màquina2 és la que s'encarrega d'acceptar-la.

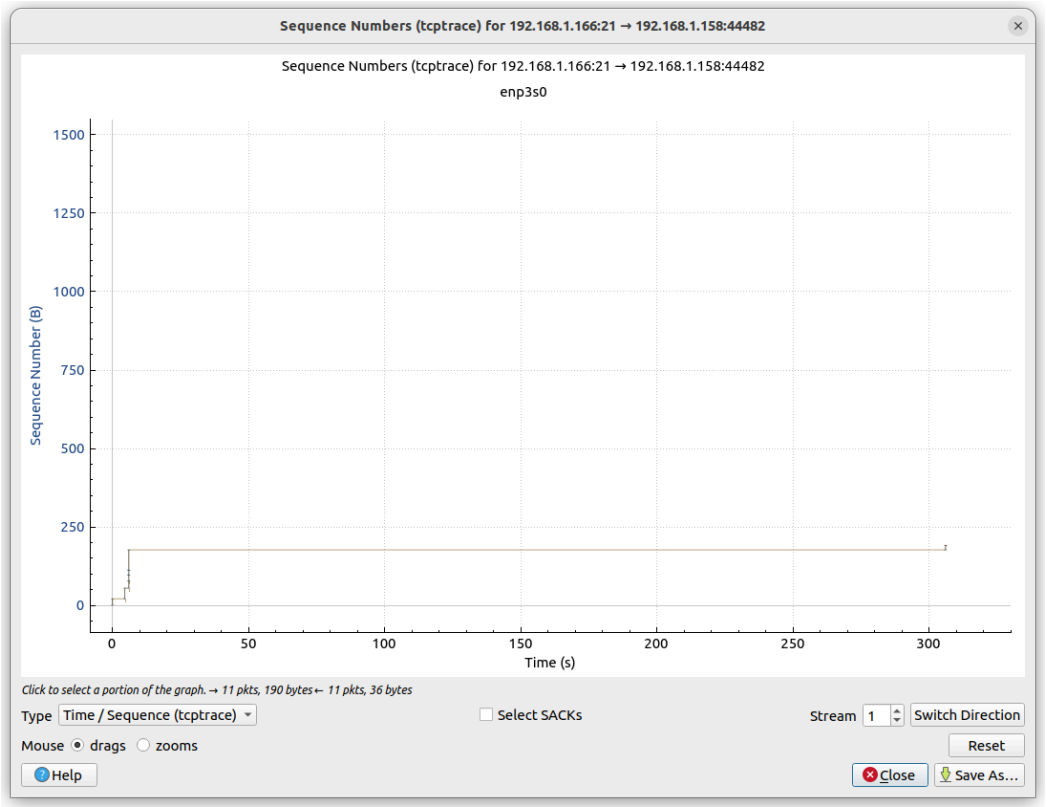
9.
Per mirar les estadístiques, hem utilitzat el paquet de SYN-ACK a l'establiment de connexió de FTP.
Hi ha 5 eines de representació gràfica:

- Time-Sequence Graph (Stevens)



Aquest gràfic mostra els números de seqüència al llarg del temps per a una connexió TCP.
És útil per identificar la pèrdua de paquets i les retransmissions.
Els camps TCP clau utilitzats són els números de seqüència i les marques de temps.

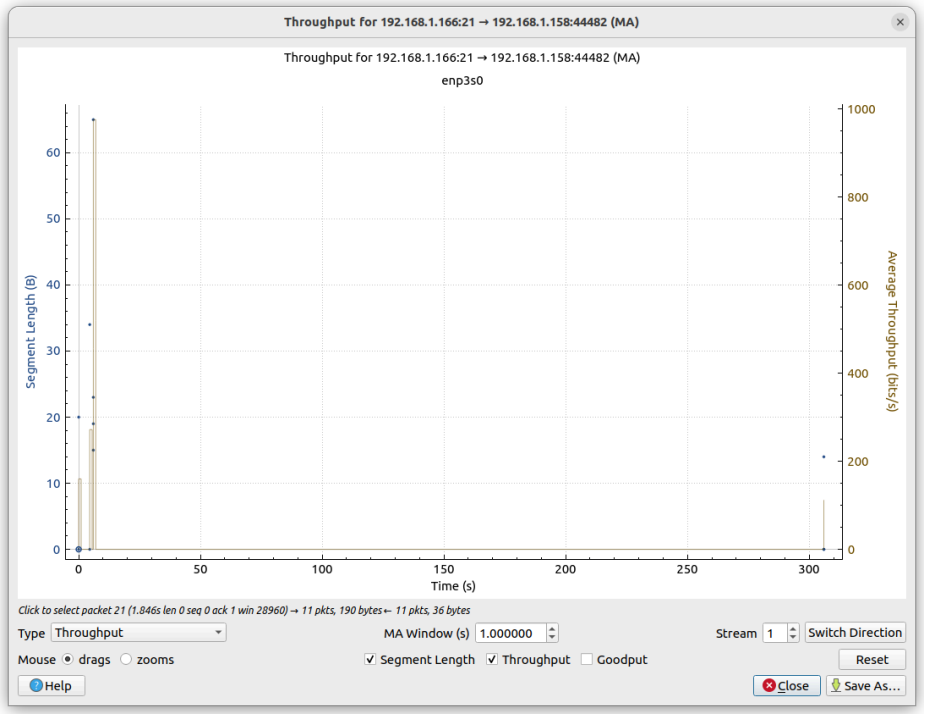
- Time-Sequance Graph (tctrace)



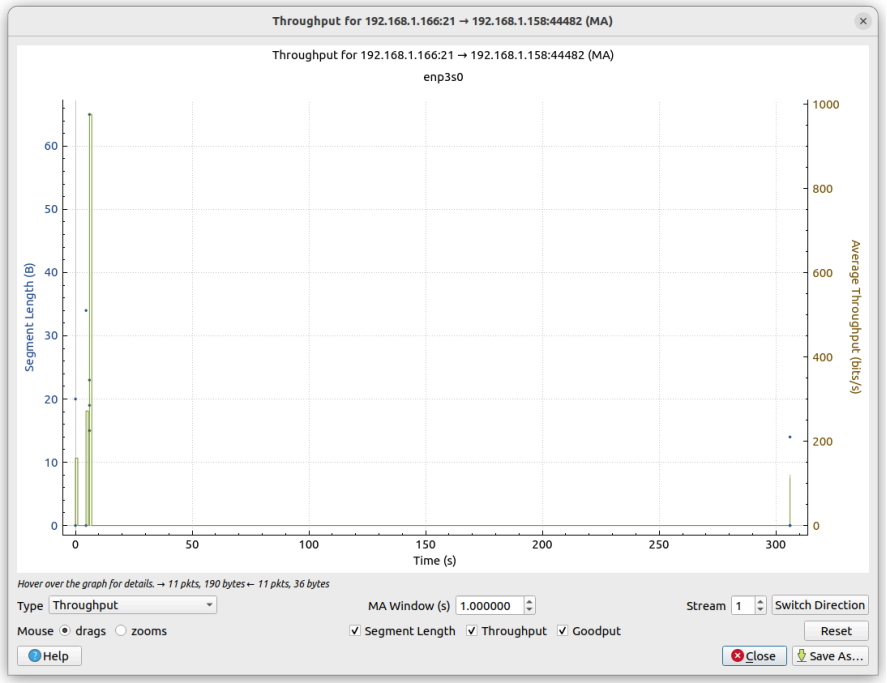
De manera similar al gràfic de Stevens, aquesta versió representa els números de seqüència TCP al llarg del temps,donant una visió més clara del flux de dades centrant-se amb les finestres de dades.
També utilitza números de seqüència i marques de temps de la capçalera TCP.

- Thoughput Graph

Throughput + segment length

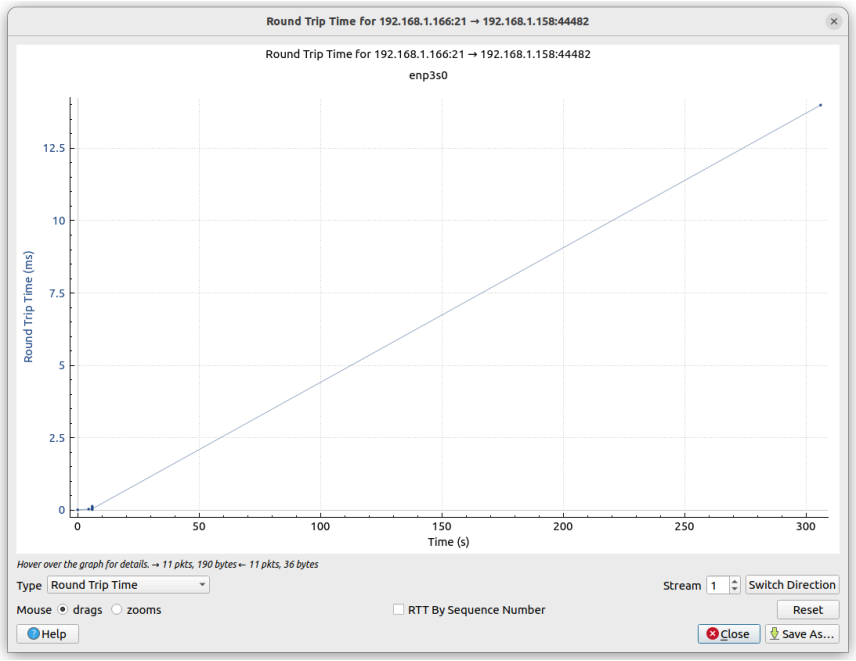


Throughput + segment length + Goodput

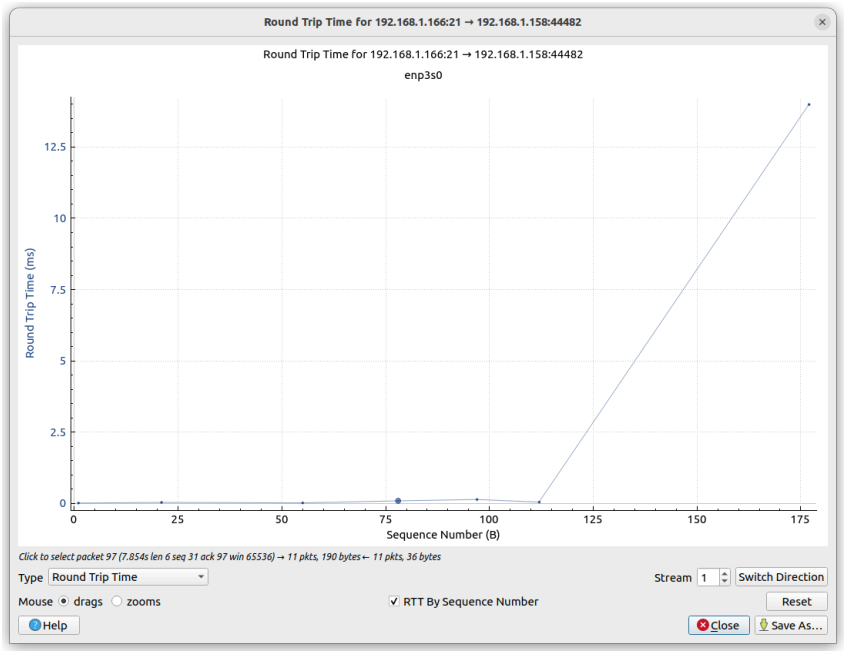


Aquest gràfic mostra la velocitat de transferència de dades al llarg del temps. És útil per analitzar l'eficiència i l'ús de la capacitat d'una connexió. El gràfic es basa en els números de seqüència i el moment en què es capturen els paquets, proporcionant informació sobre el rendiment de la connexió TCP.
La diferència entre throughput i goodput és que el throughput és la taxa total de transferència de dades incloent la sobrecàrrega i les retransmissions del protocol, mentre que Goodput, és la taxa de lliurament satisfactori de les dades reals de l'aplicació.

- Round Trip Time Graph

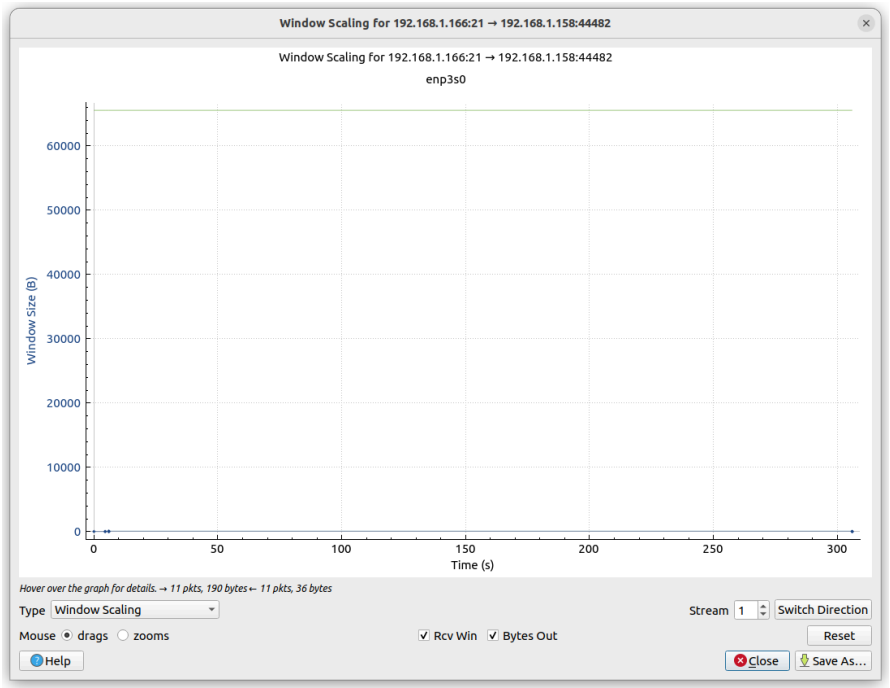


Aquest gràfic mostra l'RTT dins d'una connexió TCP. Calcula RTT mesurant la diferència de temps entre un paquet de dades i el seu corresponent reconeixement. Aquest gràfic té l'opció dels números de reconeixement juntament amb les seves corresponents marques de temps per calcular RTT.



També podem veure el RTT segons el número de seqüència (Sequence Number)

- Window Scaling Graph



La línia d'adalt verda és el Rcv Win

Aquest gràfic ensenya la mida de la finestra TCP al llarg del temps. Utilitza el camp de mida de la finestra a la capçalera TCP i fa un seguiment de com canvia durant la durada de la connexió.

10. RTT. Si està més a prop RTT més petit i si està més lluny RTT més elevat.
Les connexions de laboratori utilitzen les xarxes locals (LAN), mentre que les connexions a Internet travessen xarxes més grans i complexes.

- **Adreces IP**: En el laboratori, les adreces IP poden ser privades i internes, mentre que la connexió amb un equip a Internet utilitzarà adreces IP públiques.
- **Latència i velocitat**: Les connexions dins del laboratori tendeixen a tenir menor latència i més velocitat degut a la proximitat física dels dispositius. En canvi, les connexions a Internet poden experimentar major latència i variabilitat en la velocitat.
- **Rutes de paquets**: La ruta que segueixen els paquets en una xarxa LAN és més directa i menys complexa que en les connexions a Internet, on els paquets poden passar per múltiples routers i xarxes.