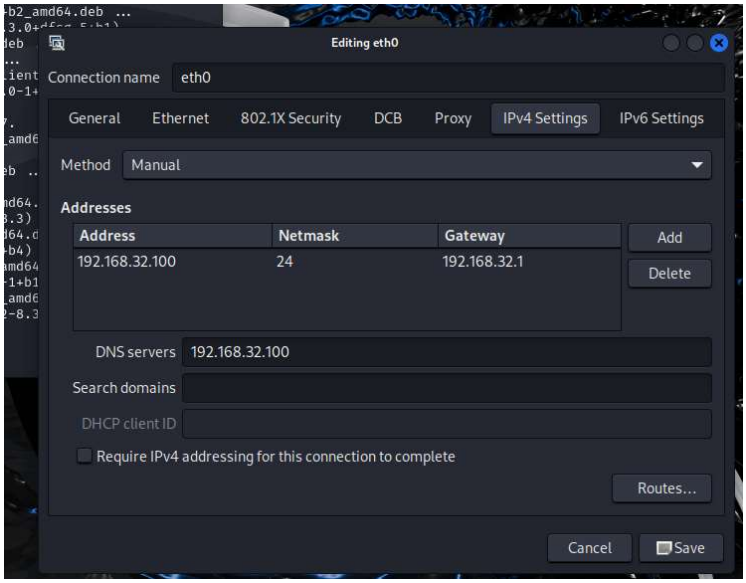
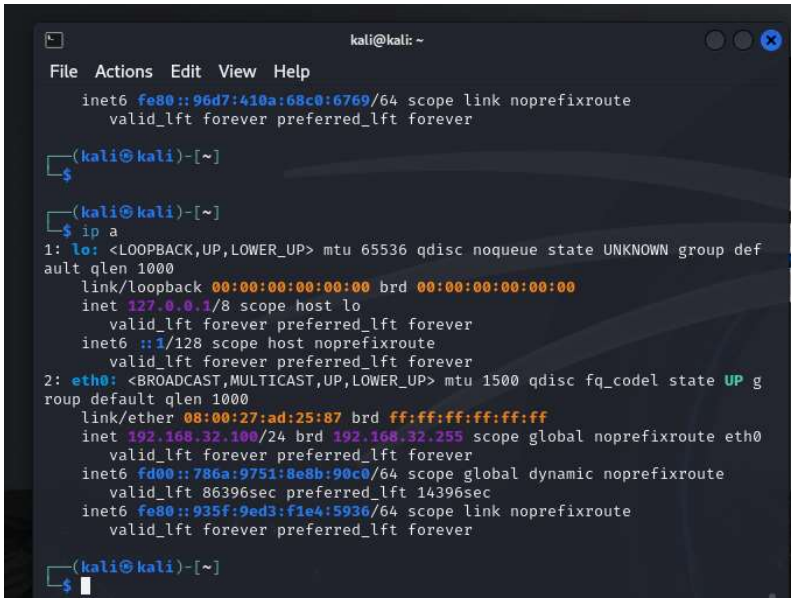
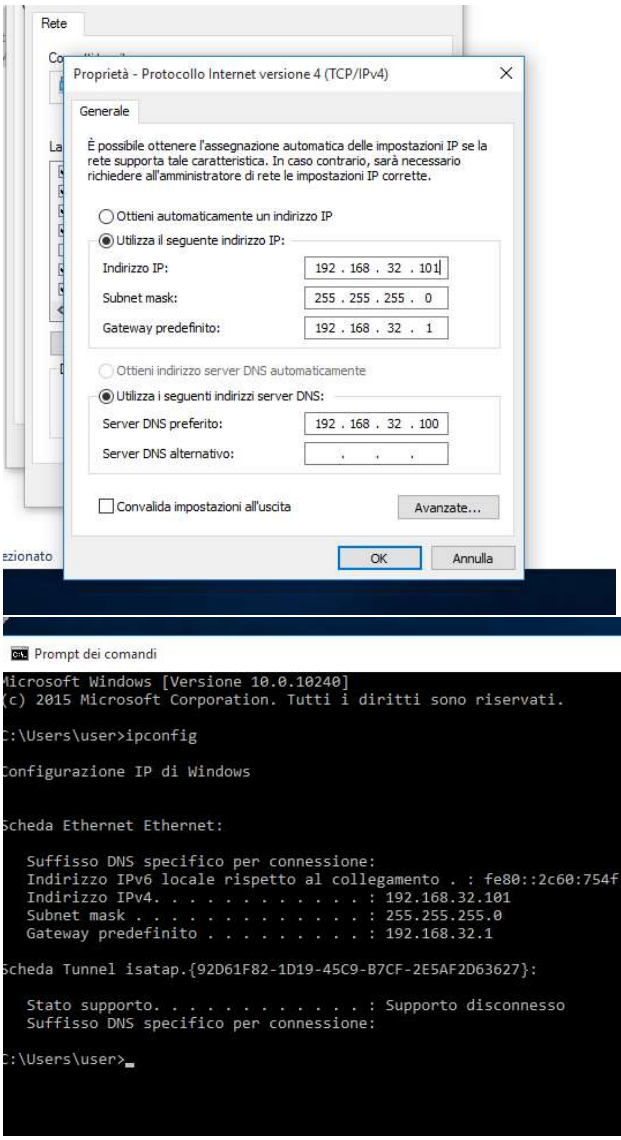
Richiesta progetto finale: mediate Wireshark intercettare il traffico HTTPS e HTTP tra Kali Linux e Windows 10 con epicode.internal

Configurazione ambiente Kali Linux IP 192.168.32.10

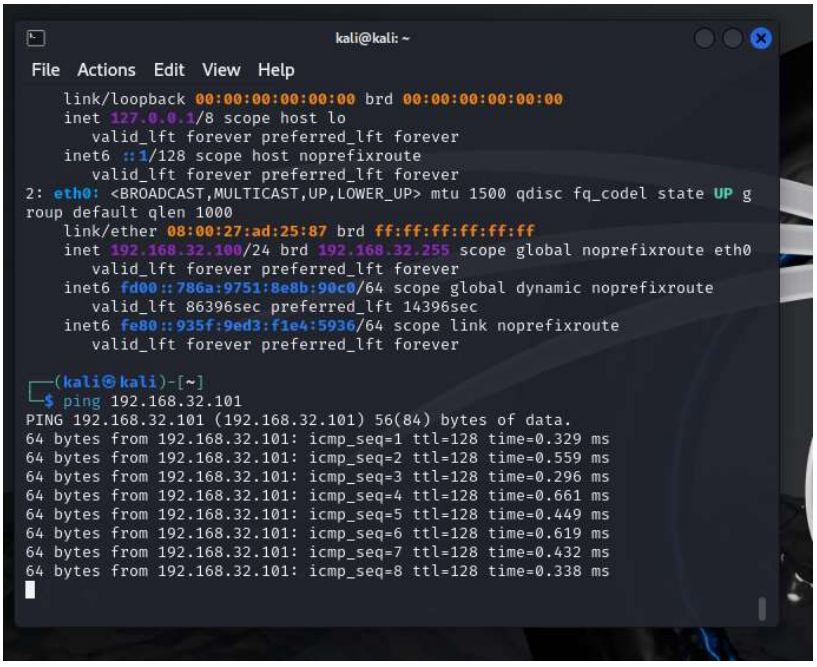


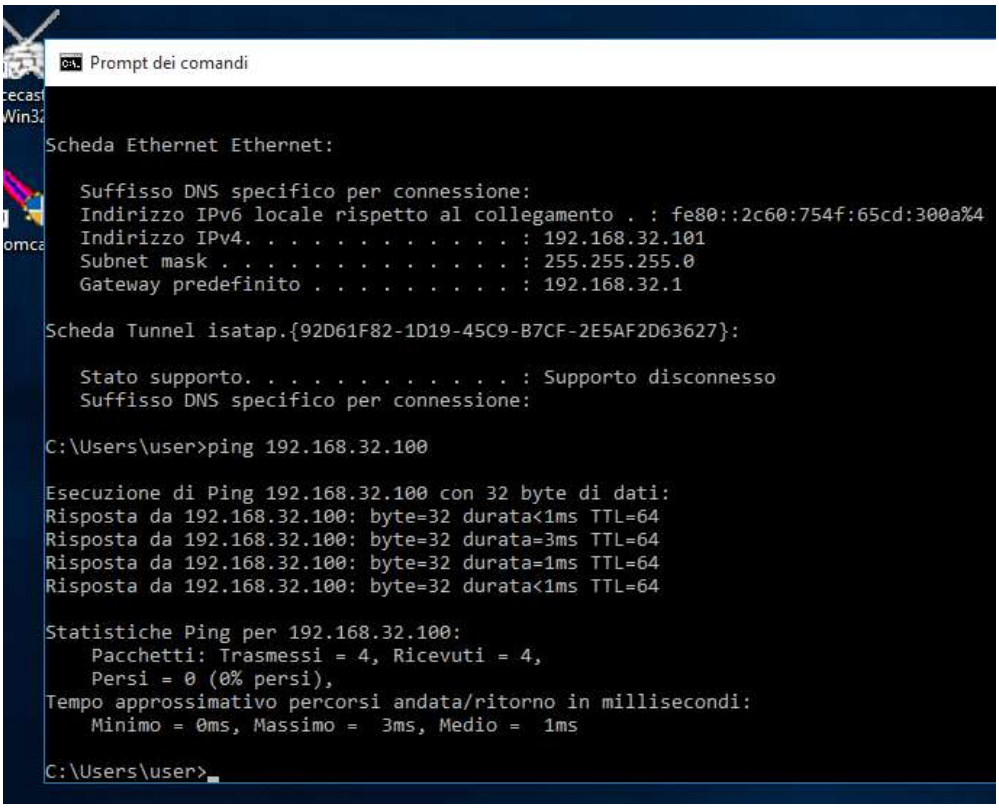


Configurazione ambiente Windows 10 IP 192.168.32.101

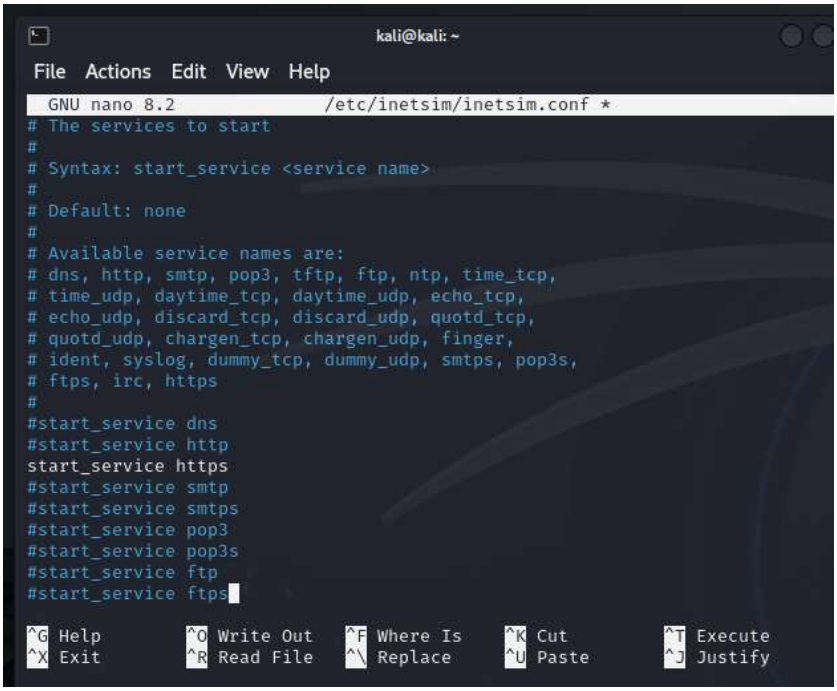


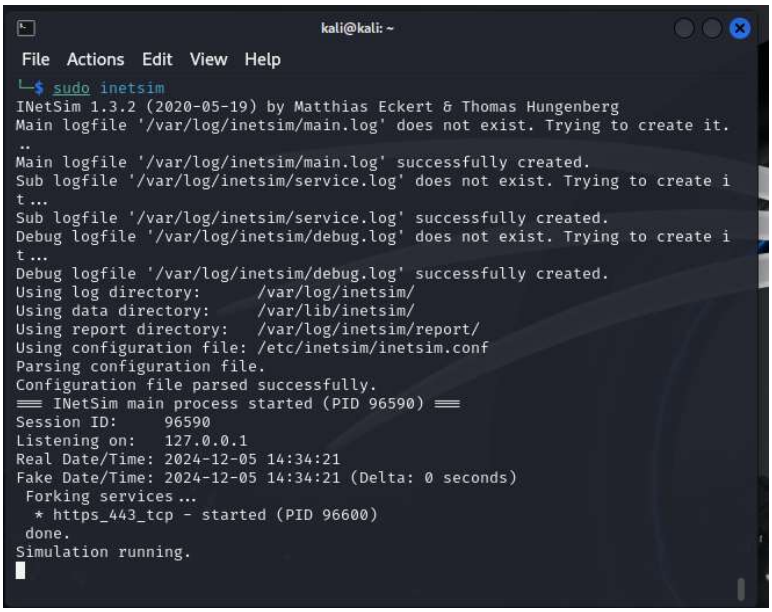
Eseguo un ping di test tra le due macchine inserendole in rete interna



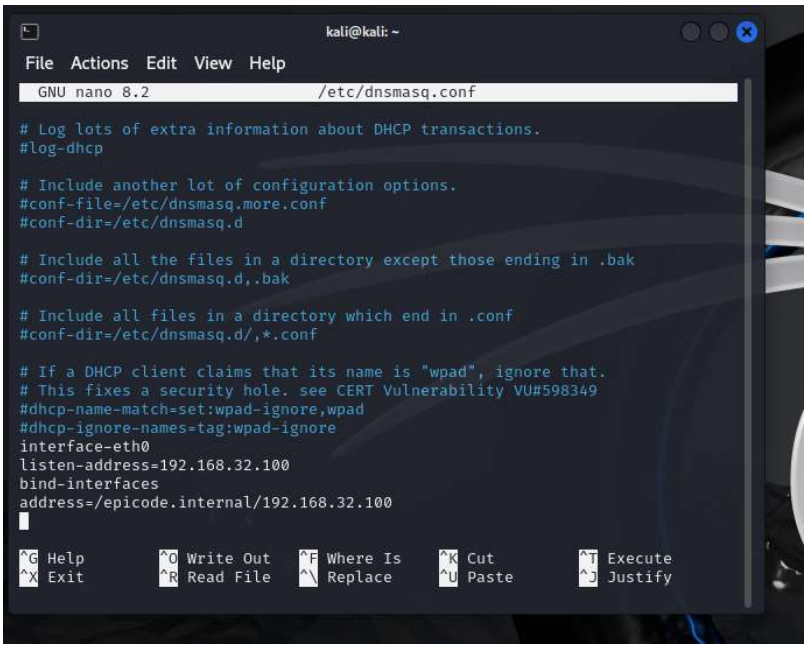


L’applicativo intesim permette di simulare il traffico HTTPS ma anche http, affinchè possa essere in grado di eseguire tale compito è necessario attivare i servizi necessari modificandone la configurazione.
Disattivo inserendo il # davanti a tutti i servizi non necessari.
Infine, attivo il servizio come da screen.

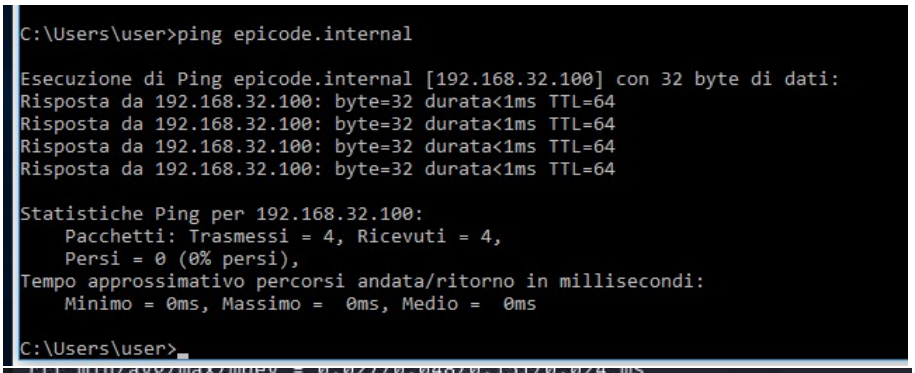


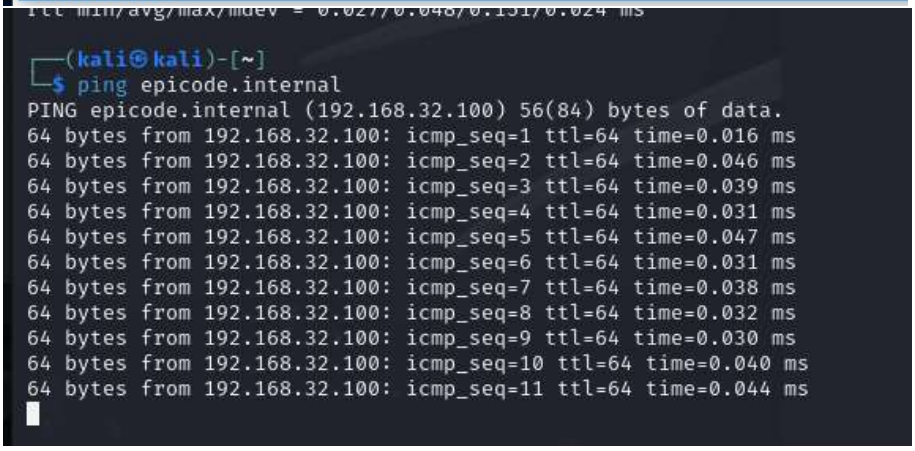


L’attivazione del server DNS viene eseguito tramite il dnsmasq precedentemente installato.

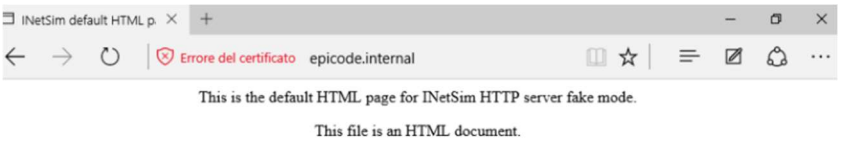


Prova ad eseguire ping su epicode.internal



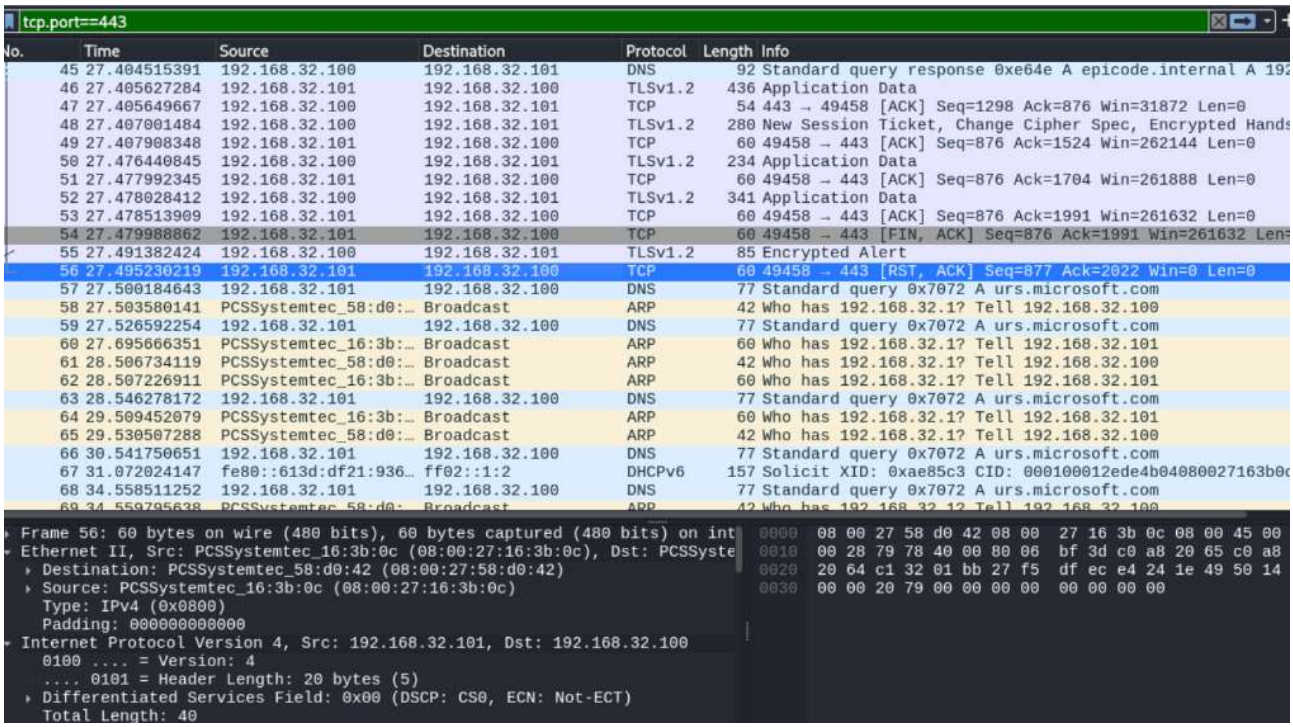


Provando a cercare sul browser predefinito in Windows http://epicode.internal viene visualizzata questa pagina HTML

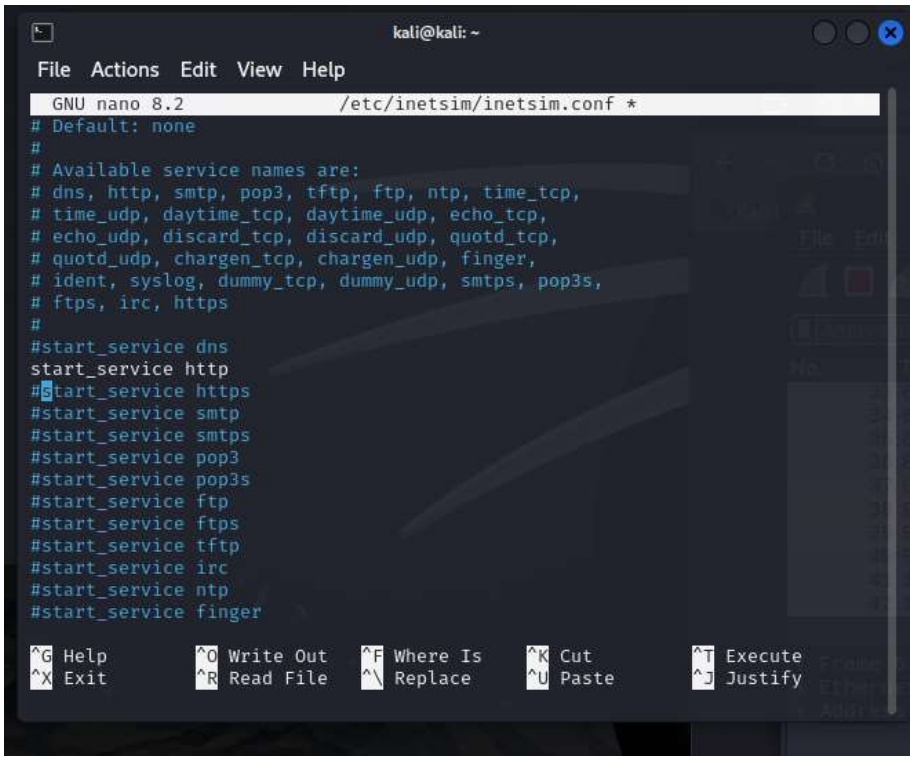


Ora apro su Kali Linux il programma Wireshark per il monitoraggio del traffico in questo caso HTTPS tra le due macchine.

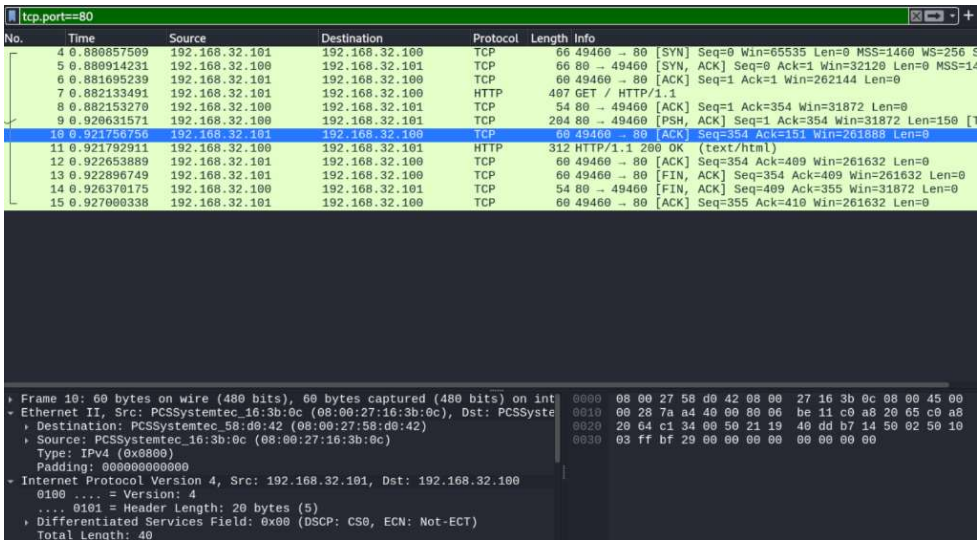
Tra le interfacce disponibili seleziono eth0 ed attivo il filtro sulla porta 443 (HTTPS).



Per eseguire la seconda parte dell'esercizio sarà necessario andare a disattivare da inetsim il servizio HTTPS e attivare HTTP



Il procedimento a seguire è il medesimo a differenza di Wireshark, quindi aprendo l'applicativo seleziono eth0 e questa volta filtro per la porta 80 (HTTP)


tcp.port==80
No. Time Source Destination Protocol Length Info
4 0.880857509 192.168.32.101 192.168.32.100 TCP 66 49460 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256
5 0.880914231 192.168.32.100 192.168.32.101 TCP 66 80 → 49460 [SYN, ACK] Seq=0 Ack=1 Win=32120 Len=0 MSS=14
6 0.881695239 192.168.32.101 192.168.32.100 TCP 60 49460 → 80 [ACK] Seq=1 Ack=1 Win=262144 Len=0
7 0.882133491 192.168.32.101 192.168.32.100 HTTP 407 GET / HTTP/1.1
8 0.882153270 192.168.32.100 192.168.32.101 TCP 54 80 → 49460 [ACK] Seq=1 Ack=354 Win=31872 Len=0
9 0.920631571 192.168.32.100 192.168.32.101 TCP 204 80 → 49460 [PSH, ACK] Seq=1 Ack=354 Win=31872 Len=150 [T
10 0.921756756 192.168.32.101 192.168.32.100 TCP 60 49460 → 80 [ACK] Seq=354 Ack=151 Win=261888 Len=0
11 0.921792911 192.168.32.100 192.168.32.101 HTTP 312 HTTP/1.1 200 OK (text/html)
12 0.922653889 192.168.32.101 192.168.32.100 TCP 60 49460 → 80 [ACK] Seq=354 Ack=409 Win=261632 Len=0
13 0.922896749 192.168.32.101 192.168.32.100 TCP 60 49460 → 80 [FIN, ACK] Seq=354 Ack=409 Win=261632 Len=0
14 0.926370175 192.168.32.100 192.168.32.101 TCP 54 80 → 49460 [FIN, ACK] Seq=409 Ack=355 Win=31872 Len=0
15 0.927000338 192.168.32.101 192.168.32.100 TCP 60 49460 → 80 [ACK] Seq=355 Ack=410 Win=261632 Len=0
Frame 10: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on int
Ethernet II, Src: PCSSystemtec_16:3b:0c (08:00:27:16:3b:0c), Dst: PCSSyste
Destination: PCSSystemtec_58:d0:42 (08:00:27:58:d0:42)
Source: PCSSystemtec_16:3b:0c (08:00:27:16:3b:0c)
Type: IPv4 (0x0800)
Padding: 000000000000
Internet Protocol Version 4, Src: 192.168.32.101, Dst: 192.168.32.100
0100 .... = Version: 4
.... 0101 = Header Length: 20 bytes (5)
Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
Total Length: 40
0000 08 00 27 58 d0 42 08 00 27 16 3b 0c 08 00 45 00
0010 00 28 7a a4 40 00 80 06 be 11 c0 a8 20 65 c0 a8
0020 20 64 c1 34 00 50 21 19 40 dd b7 14 50 02 50 10
0030 03 ff bf 29 00 00 00 00 00 00 00 00