Benchmark W4D4

Traccia:

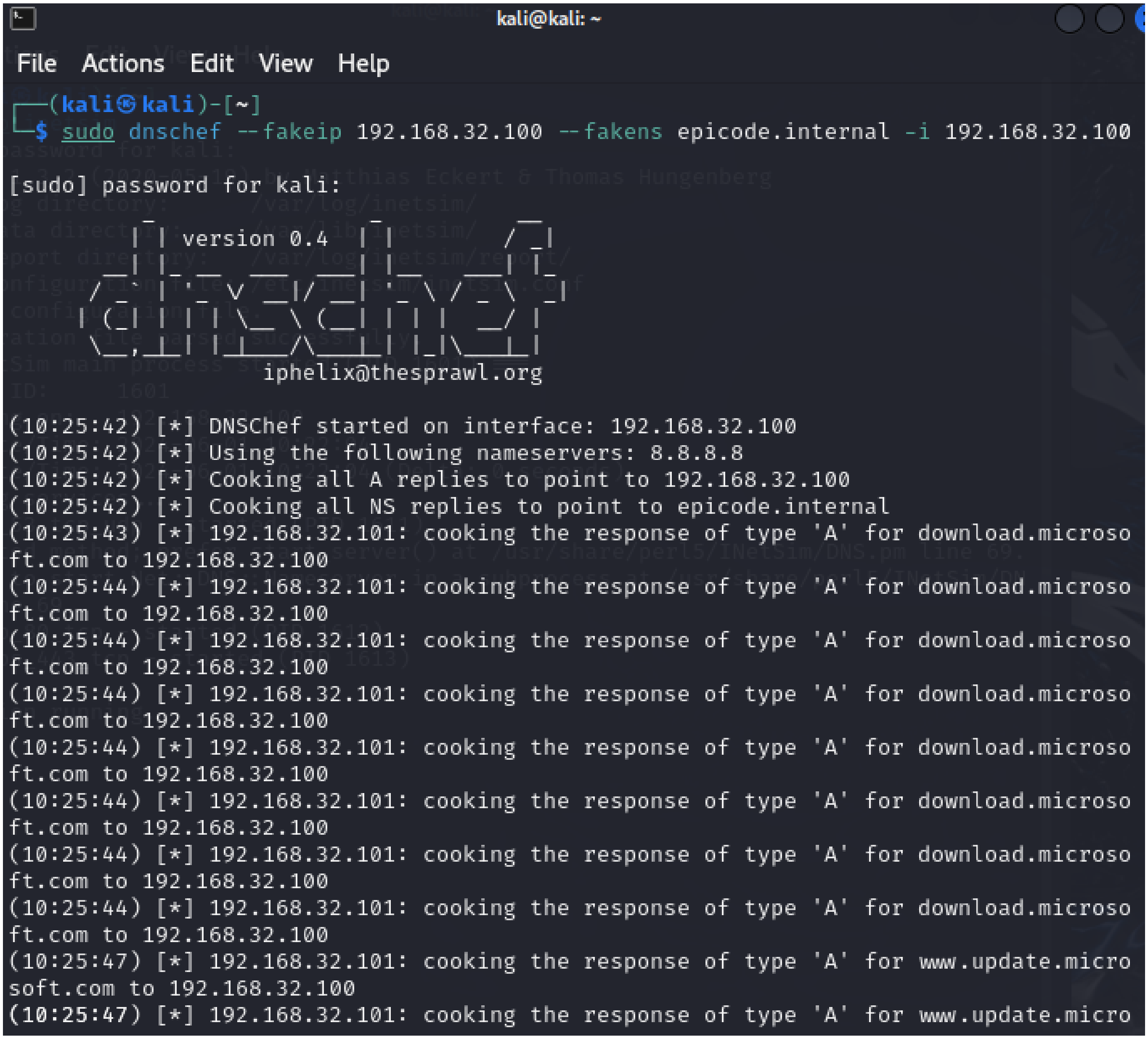
Simulare, in ambiente di laboratorio virtuale, un'architettura client server in cui un client con indirizzo 192.168.32.101 (Windows 7) richiede tramite web browser una risorsa all'hostname **epicode.internal** che risponde all'indirizzo 192.168.32.100 (Kali).

Si intercetti poi la comunicazione con Wireshark, evidenziando i MAC address di sorgente e destinazione ed il contenuto della richiesta HTTPS.

Per prima cosa, dopo aver avviato le due macchine (Kali e Windows 7) le imposto per renderle una server (Kali) e l'altra client (Windows 7).
In Kali avvio inetsim:

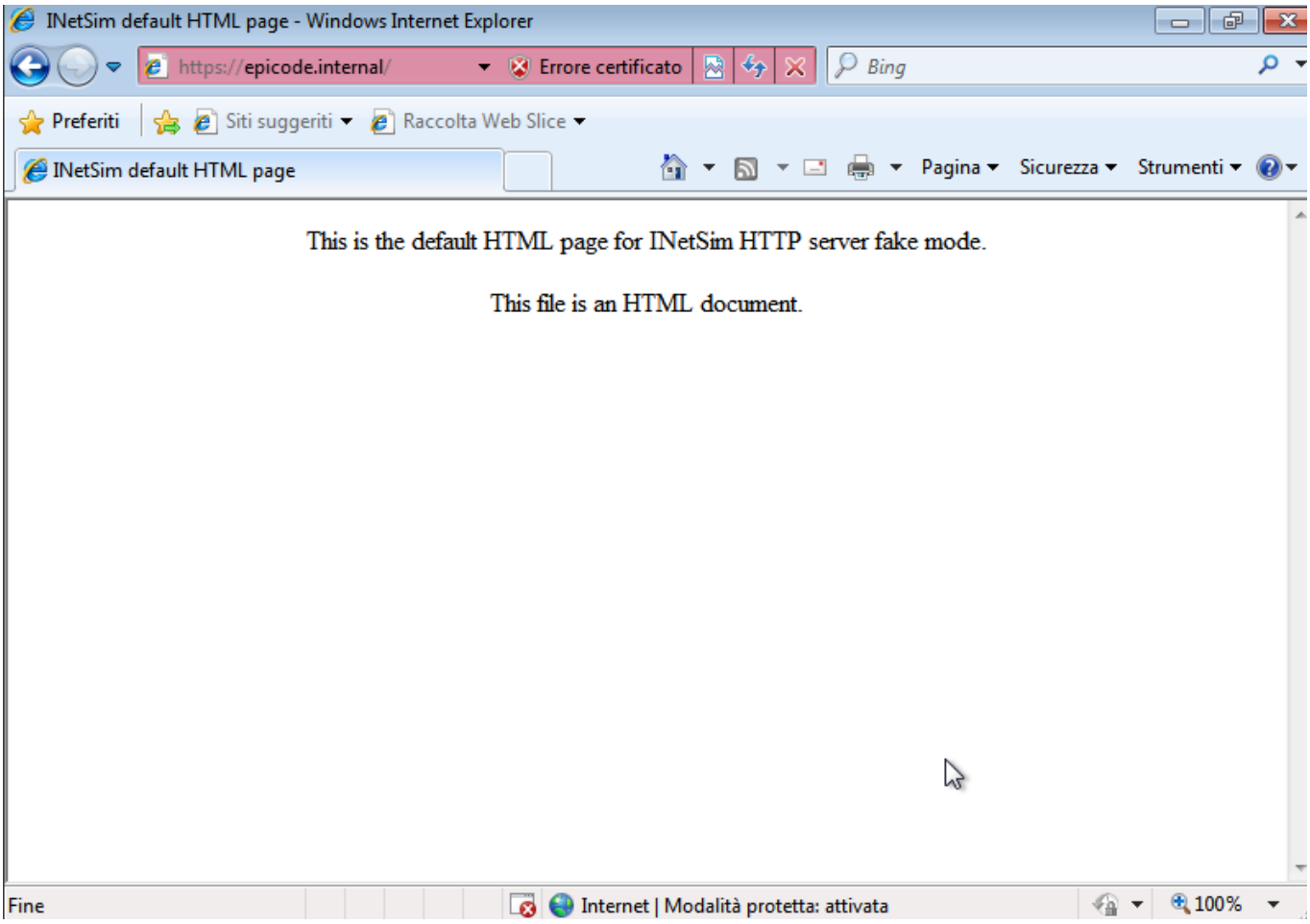
```
kali@kali: ~
File  Actions  Edit  View  Help
┌──(kali㉿kali)-[~]
└─$ sudo inetsim
[sudo] password for kali:
INetSim 1.3.2 (2020-05-19) by Matthias Eckert & Thomas Hungenberg
Using log directory:      /var/log/inetsim/
Using data directory:     /var/lib/inetsim/
Using report directory:   /var/log/inetsim/report/
Using configuration file: /etc/inetsim/inetsim.conf
Parsing configuration file.
Configuration file parsed successfully.
=== INetSim main process started (PID 1601) ===
Session ID:     1601
Listening on:   192.168.32.100
Real Date/Time: 2024-06-01 10:22:04
Fake Date/Time: 2024-06-01 10:22:04 (Delta: 0 seconds)
 Forking services...
  * dns_53_tcp_udp - started (PID 1611)
deprecated method; prefer start_server() at /usr/share/perl5/INetSim/DNS.pm line 69.
Attempt to start Net::DNS::Nameserver in a subprocess at /usr/share/perl5/INetSim/DN
S.pm line 69.
  * http_80_tcp - started (PID 1612)
  * https_443_tcp - started (PID 1613)
 done.
Simulation running.
```

Avvio poi dnschef:



```
┌──(kali㉿kali)-[~]
└─$ sudo dnschef --fakeip 192.168.32.100 --fakens epicode.internal -i 192.168.32.100

[sudo] password for kali:
          _                _          __
         | | version 0.4  | |        / _|
       __| |_ __  ___  ___| |__   ___| |_
      / _` | '_ \/ __|/ __| '_ \ / _ \  _|
     | (_| | | | \__ \ (__| | | |  __/ |
      \__,_|_| |_|___/\___|_| |_|\___|_|
                   iphelix@thesprawl.org

(10:25:42) [*] DNSChef started on interface: 192.168.32.100
(10:25:42) [*] Using the following nameservers: 8.8.8.8
(10:25:42) [*] Cooking all A replies to point to 192.168.32.100
(10:25:42) [*] Cooking all NS replies to point to epicode.internal
(10:25:43) [*] 192.168.32.101: cooking the response of type 'A' for download.microso
ft.com to 192.168.32.100
(10:25:44) [*] 192.168.32.101: cooking the response of type 'A' for download.microso
ft.com to 192.168.32.100
(10:25:44) [*] 192.168.32.101: cooking the response of type 'A' for download.microso
ft.com to 192.168.32.100
(10:25:44) [*] 192.168.32.101: cooking the response of type 'A' for download.microso
ft.com to 192.168.32.100
(10:25:44) [*] 192.168.32.101: cooking the response of type 'A' for download.microso
ft.com to 192.168.32.100
(10:25:44) [*] 192.168.32.101: cooking the response of type 'A' for download.microso
ft.com to 192.168.32.100
(10:25:44) [*] 192.168.32.101: cooking the response of type 'A' for download.microso
ft.com to 192.168.32.100
(10:25:44) [*] 192.168.32.101: cooking the response of type 'A' for download.microso
ft.com to 192.168.32.100
(10:25:47) [*] 192.168.32.101: cooking the response of type 'A' for www.update.micro
soft.com to 192.168.32.100
(10:25:47) [*] 192.168.32.101: cooking the response of type 'A' for www.update.micro
```

Su Windows 7 faccio la richiesta HTTPS:



Mi controllo gli indirizzi IP e MAC delle due macchine con i loro rispettivi comandi:

C:\Windows\system32\cmd.exe

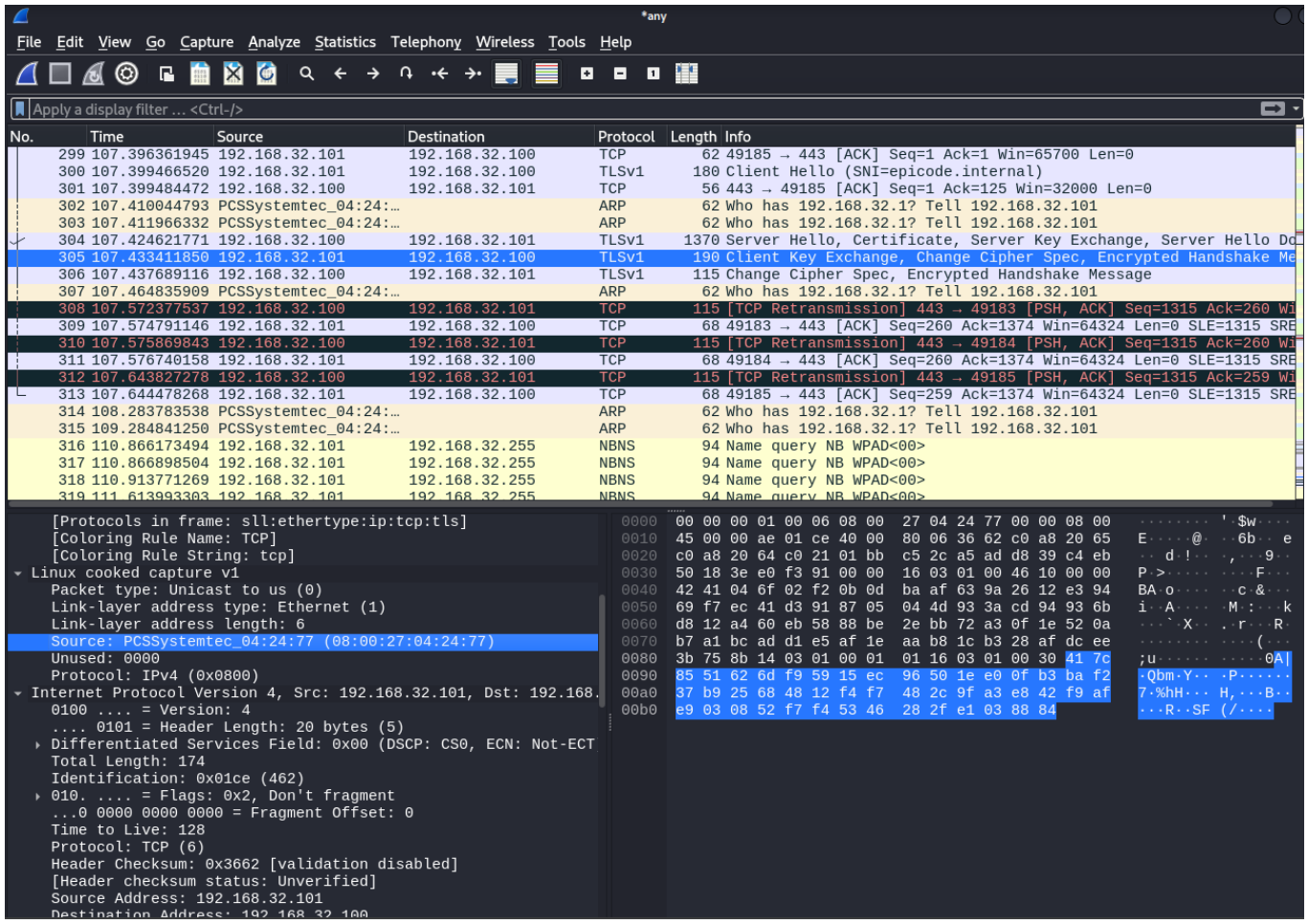
```
Microsoft Windows [Versione 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. Tutti i diritti riservati.

C:\Users\Daniele>ipconfig /all

Configurazione IP di Windows

   Nome host . . . . . . . . . . . . . : Daniele-PC
   Suffisso DNS primario . . . . . . . :
   Tipo nodo . . . . . . . . . . . . . : Ibrido
   Routing IP abilitato. . . . . . . . : No
   Proxy WINS abilitato . . . . . . . . : No

Scheda Ethernet Connessione alla rete locale (LAN):

   Suffisso DNS specifico per connessione:
   Descrizione . . . . . . . . . . . . : Scheda desktop Intel(R) PRO/1000 MT
   Indirizzo fisico. . . . . . . . . . : 08-00-27-04-24-77
   DHCP abilitato. . . . . . . . . . . : No
   Configurazione automatica abilitata   : Sì
   Indirizzo IPv6 locale rispetto al collegamento . : fe80::85be:a319:c60d:e892%
11(Preferenziale)
   Indirizzo IPv4. . . . . . . . . . . : 192.168.32.101(Preferenziale)
   Subnet mask . . . . . . . . . . . . : 255.255.255.0
   Gateway predefinito . . . . . . . . : 192.168.32.1
   IAID DHCPv6 . . . . . . . . . . . . : 235405351
   DUID Client DHCPv6. . . . . . . . . : 00-01-00-01-2D-D3-F0-70-08-00-27-04-24-77

   Server DNS . . . . . . . . . . . . .  : 192.168.32.100
                                           8.8.4.4
   NetBIOS su TCP/IP . . . . . . . . . . : Attivato

Scheda Tunnel isatap.{1F6A7472-EE00-478D-8FAC-6F32F71919B6}:

   Stato supporto. . . . . . . . . . . : Supporto disconnesso
   Suffisso DNS specifico per connessione:
   Descrizione . . . . . . . . . . . . : Microsoft ISATAP Adapter
   Indirizzo fisico. . . . . . . . . . : 00-00-00-00-00-00-00-E0
   DHCP abilitato. . . . . . . . . . . : No
   Configurazione automatica abilitata   : Sì

C:\Users\Daniele>_
```

```
kali@kali: ~
File  Actions  Edit  View  Help

┌──(kali㉿kali)-[~]
└─$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:1e:36:4a brd ff:ff:ff:ff:ff:ff
    inet 192.168.32.100/24 brd 192.168.32.255 scope global noprefixroute eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::e5ee:b875:45f7:50a0/64 scope link noprefixroute
       valid_lft forever preferred_lft forever

┌──(kali㉿kali)-[~]
└─$
```

Avvio Wireshark e registro i pacchetti in entrata e in uscita da Kali:

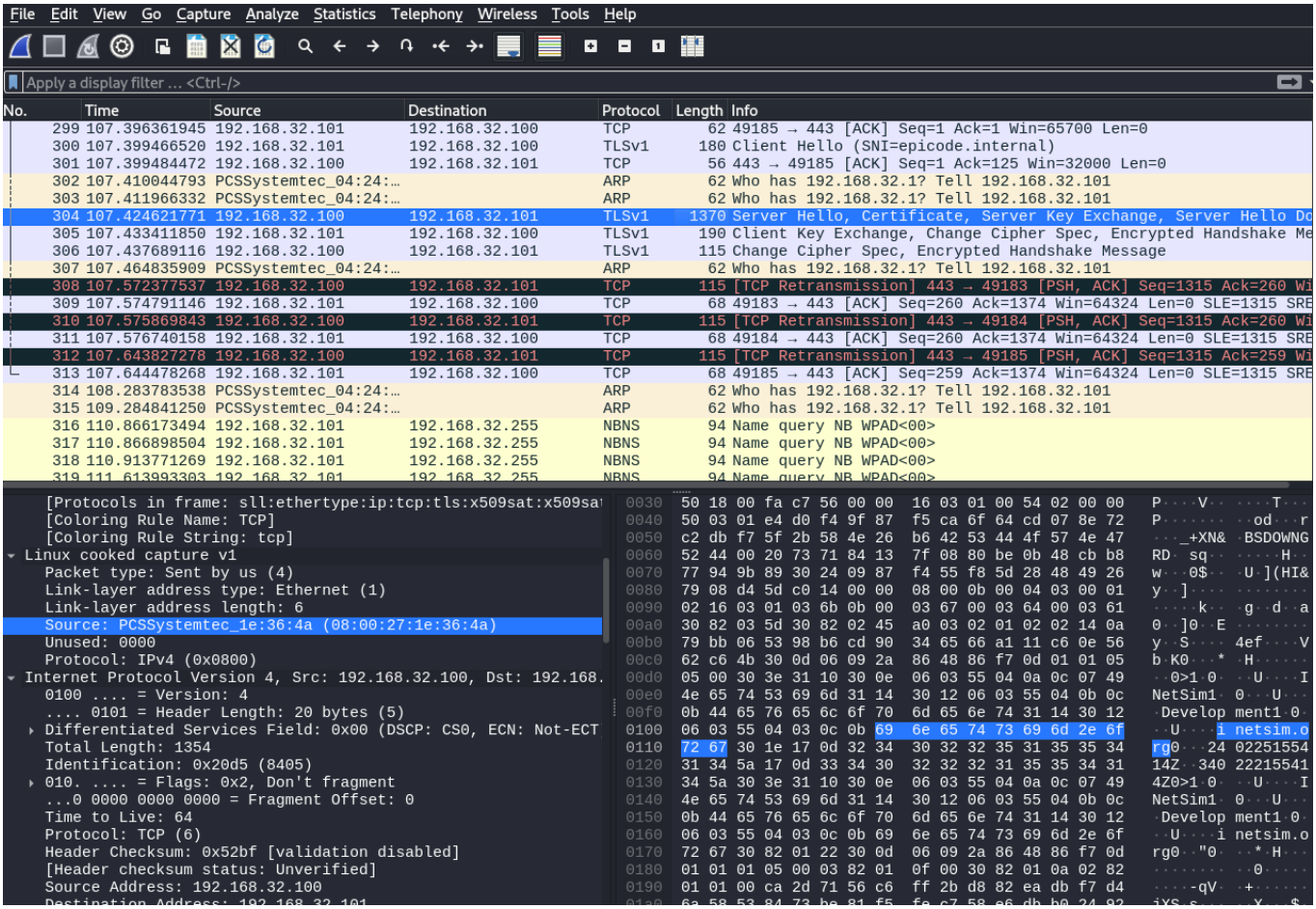
10	0.008726846	192.168.32.101	192.168.32.100	TCP	62	49308 → 443 [ACK] Seq=1 Ack=1 Win=65700 Len=0
11	0.015818398	192.168.32.101	192.168.32.100	TLSv1	213	Client Hello (SNI=urs.microsoft.com)
12	0.015818773	192.168.32.101	192.168.32.100	TLSv1	213	Client Hello (SNI=urs.microsoft.com)
13	0.015839794	192.168.32.100	192.168.32.101	TCP	56	443 → 49307 [ACK] Seq=1 Ack=158 Win=32000 Len=0
14	0.015916289	192.168.32.100	192.168.32.101	TCP	56	443 → 49308 [ACK] Seq=1 Ack=158 Win=32000 Len=0
15	0.053361731	192.168.32.101	192.168.32.100	TCP	68	49309 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=4 SACK_PERM
16	0.053385353	192.168.32.100	192.168.32.101	TCP	68	443 → 49309 [SYN, ACK] Seq=0 Ack=1 Win=32120 Len=0 MSS=1460 SACK_PERM WS=128
17	0.054975827	192.168.32.101	192.168.32.100	TCP	62	49309 → 443 [ACK] Seq=1 Ack=1 Win=65700 Len=0
18	0.055930888	192.168.32.101	192.168.32.100	TLSv1	212	Client Hello (SNI=epicode.internal)
19	0.055944131	192.168.32.100	192.168.32.101	TCP	56	443 → 49309 [ACK] Seq=1 Ack=157 Win=32000 Len=0
20	0.060298643	192.168.32.100	192.168.32.101	TLSv1	1370	Server Hello, Certificate, Server Key Exchange, Server Hello Done
21	0.061076124	192.168.32.100	192.168.32.101	TLSv1	1370	Server Hello, Certificate, Server Key Exchange, Server Hello Done
22	0.067744939	192.168.32.101	192.168.32.100	TLSv1	190	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
23	0.067824911	192.168.32.100	192.168.32.101	TCP	56	443 → 49308 [ACK] Seq=1315 Ack=292 Win=31872 Len=0

```
▸ Frame 11: 213 bytes on wire (1704 bits), 213 bytes captured (1704 bits) on interface any, id
▾ Linux cooked capture v1
    Packet type: Unicast to us (0)
    Link-layer address type: Ethernet (1)
    Link-layer address length: 6
    Source: PCSSystemtec_04:24:77 (08:00:27:04:24:77)
    Unused: 0000
    Protocol: IPv4 (0x0800)
▸ Internet Protocol Version 4, Src: 192.168.32.101, Dst: 192.168.32.100
▸ Transmission Control Protocol, Src Port: 49307, Dst Port: 443, Seq: 1, Ack: 1, Len: 157
▸ Transport Layer Security

0000  00 00 00 01 00 06 08 00  27 04 24 77 00 00 08 00   ........ '$w....
0010  45 00 00 c5 06 47 40 00  80 06 31 d2 c0 a8 20 65   E...G@.. ..1...e
0020  c0 a8 20 64 c0 9b 01 bb  b1 27 6d 38 0c e0 46 97   ...d.... .'m8..F.
0030  50 18 40 29 00 69 00 00  16 03 01 00 98 01 00 00   P.@).i.. ........
0040  94 03 01 66 5b 48 e2 bb  c6 88 38 76 0f 5a 4f 87   ...f[H.. ..8v.ZO.
0050  a3 31 79 a6 ab 31 a3 1b  c0 6b 61 1e e6 ad f8 85   .1y..1.. .ka.....
0060  d0 fa 49 20 91 03 08 af  df ae 1c ab 84 14 8a 88   ..I ..... ........
0070  a3 52 45 db 32 0f 55 b7  99 a6 55 3c bc aa 89 bd   .RE.2.U. ..U<....
0080  32 a3 3a e9 00 18 00 2f  00 35 00 05 00 0a c0 13   2.:..../ .5......
0090  c0 14 c0 09 c0 0a 00 32  00 38 00 13 00 04 01 00   .......2 .8......
00a0  00 33 00 00 00 16 00 14  00 00 11 75 72 73 2e 6d   .3...... ...urs.m
00b0  69 63 72 6f 73 6f 66 74  2e 63 6f 6d 00 05 00 05   icrosoft .com....
```



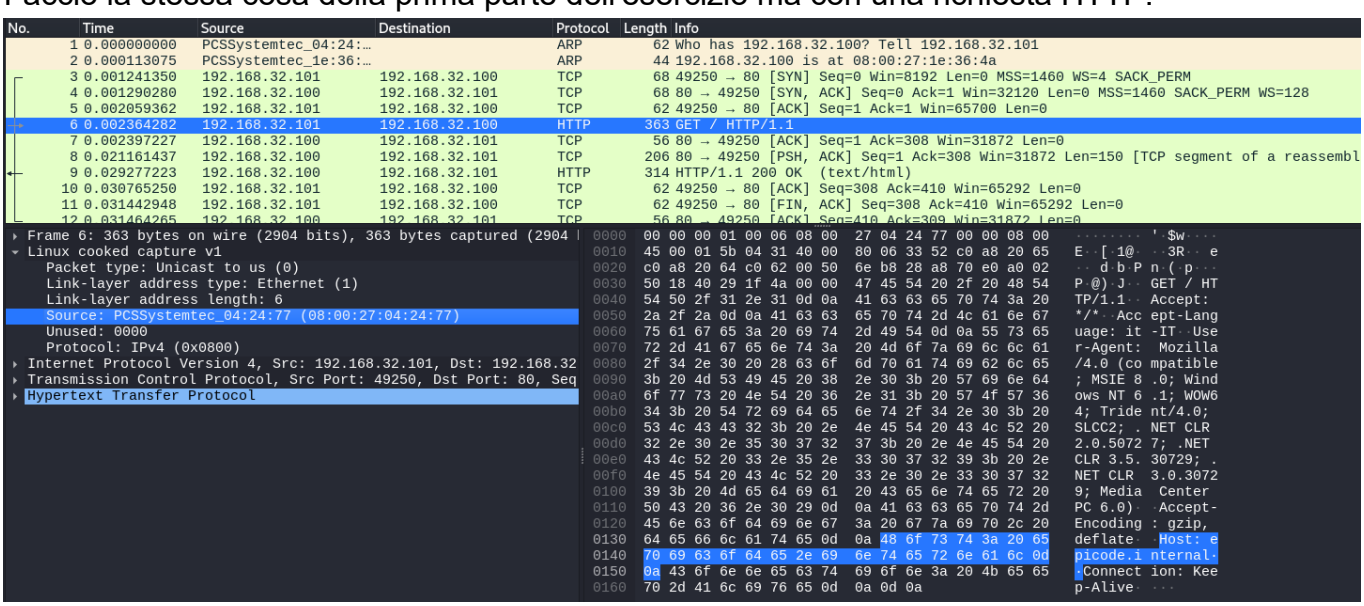
Noto che la connessione è privata vedendo il protocollo TLSv1 (Transport Layer Security) e che

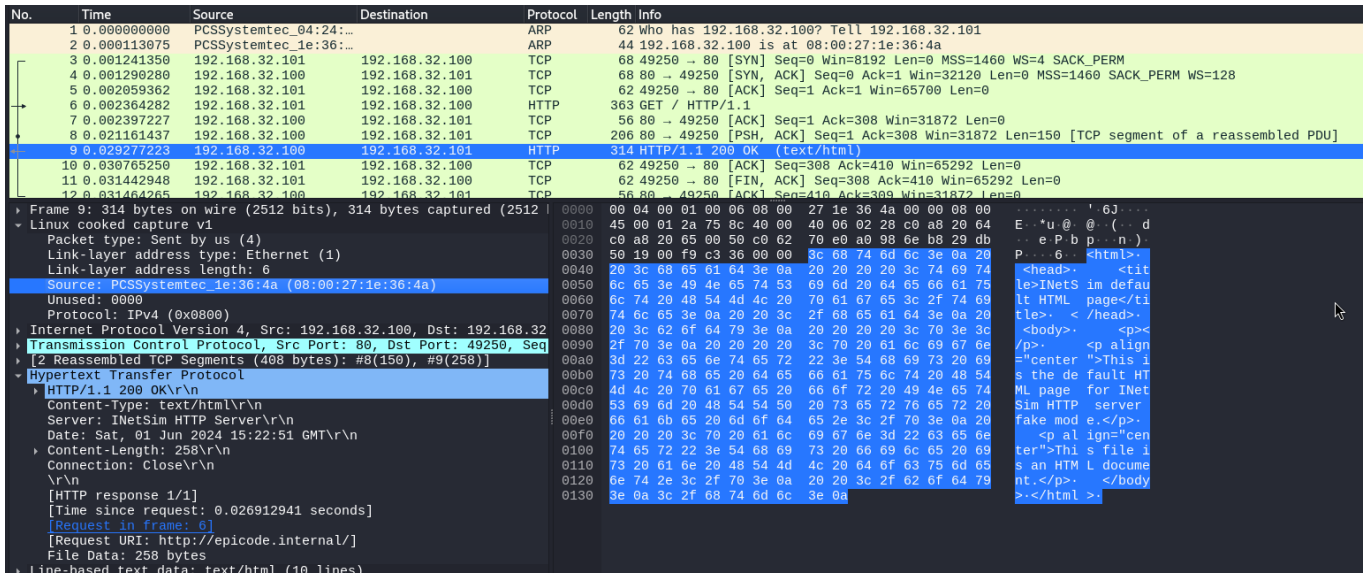
gli indirizzi IP e MAC sono corretti:



Ripetere l'esercizio, sostituendo il server HTTPS, con un server HTTP. Si intercetti nuovamente il traffico, evidenziando le eventuali differenze tra il traffico appena catturato in HTTP ed il traffico precedente in HTTPS. Spiegare, motivandole, le principali differenze se presenti.

Faccio la stessa cosa della prima parte dell'esercizio ma con una richiesta HTTP:





La differenza principale tra le richieste HTTP e HTTPS catturate con Wireshark è che con HTTP puoi vedere tutte le informazioni trasmesse in chiaro, mentre con HTTPS tutto il contenuto dopo l'handshake è cifrato e non visibile direttamente.