

Report day 5

Obbiettivo: Intercettare la comunicazione tra Windows 7 e il sito HTTP e HTTPS

Software per Virtual Lab: Oracle Virtual Box

Macchine Virtuali:

Kali Linux (IP: 192.168.32.100)

Windows 7 (IP: 192.168.32.101)

Tools:

Wireshark: software usato per sniffing

Inetsim: software per simulare dei servizi internet in un ambiente virtuale.


Prima di tutto cambieremo l'IP del Kali Linux, "192.168.50.100" > "192.168.32.100"

aprendo il terminal nel Kali e eseguendo il comando: "sudo nano /etc/network/interfaces" visto che abbiamo usato il comando "sudo" per poter modificare il file "interfaces" ci verrà richiesta la password. Inserita la password potremmo modificare il terzo ottetto del address dal 50 al 32, la stessa manovra la faremo per il gateway, il terzo ottetto dal 50 al 32, per chiudere si preme sulla tastiera "Control" + "O", Invio, "Control" + "X"

Ctrl + O andrà a sovrascrivere le modifiche eseguite sul file.

Invio confermerà il nome del file, che non andremo a modificare.

Ctrl + X per chiudere il file e tornare sul Directory del Terminal precedente

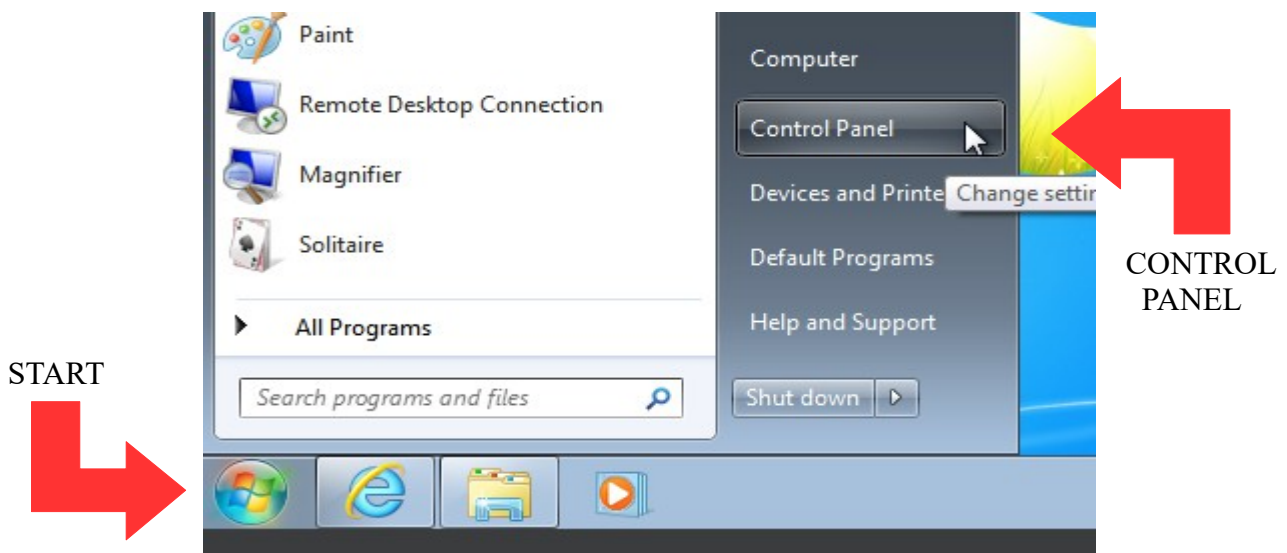


```
auto eth0
iface eth0 inet static
address 192.168.50.100/24
gateway 192.168.50.1
```

```
auto eth0
iface eth0 inet static
address 192.168.32.100/24
gateway 192.168.32.1
```

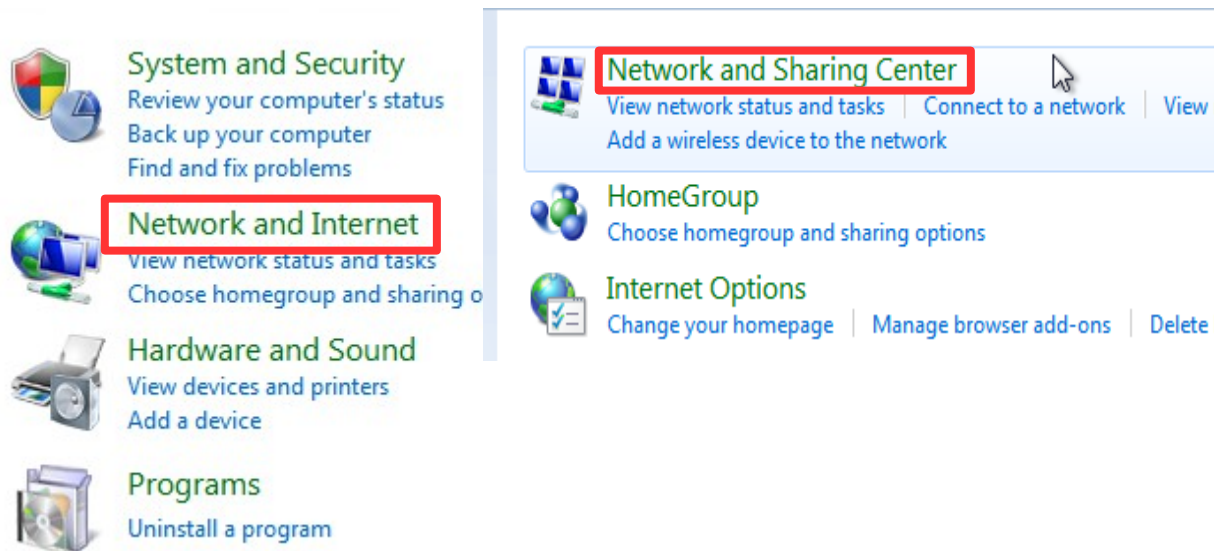
Il prossimo passo è cambiare l'IP del Win7 dal "192.168.50.102" in "192.168.32.101"

Win7 essendo più user-friendly, faremo il click sul Start e clic sul Control Panel:

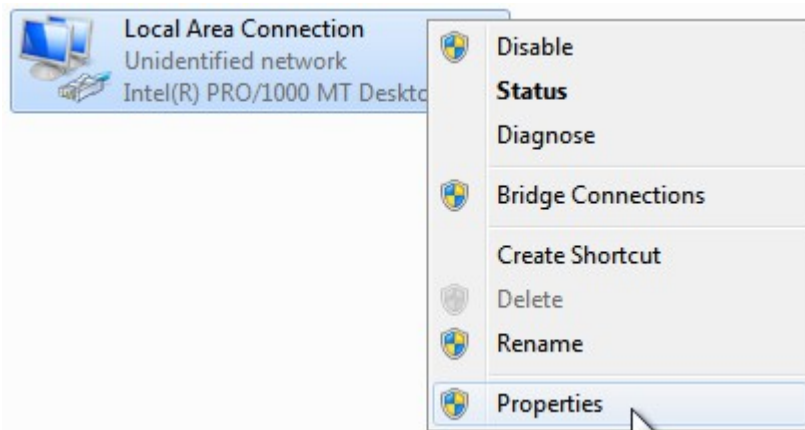


poi scegliamo Network and Internet

e dopo clicchiamo sul Network and Sharing Center

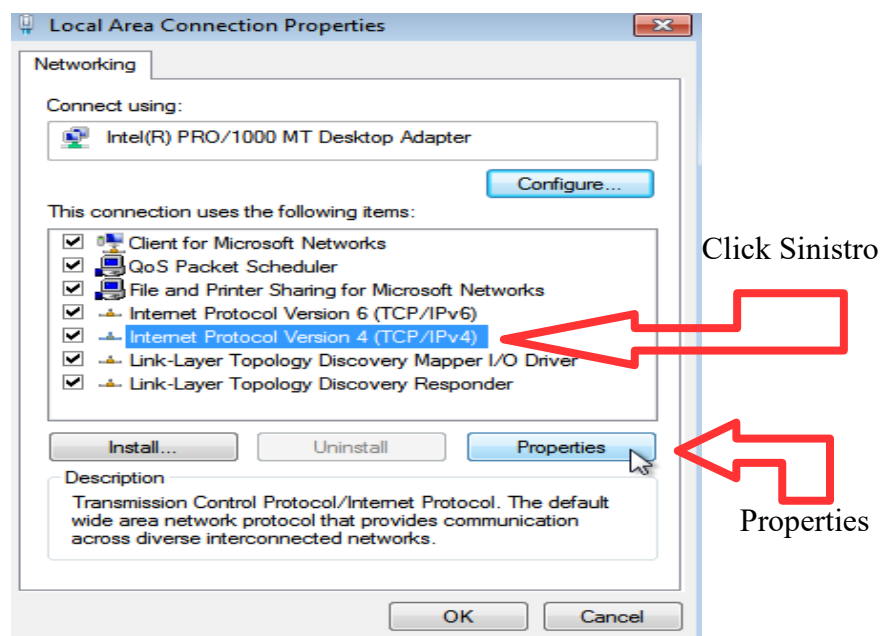


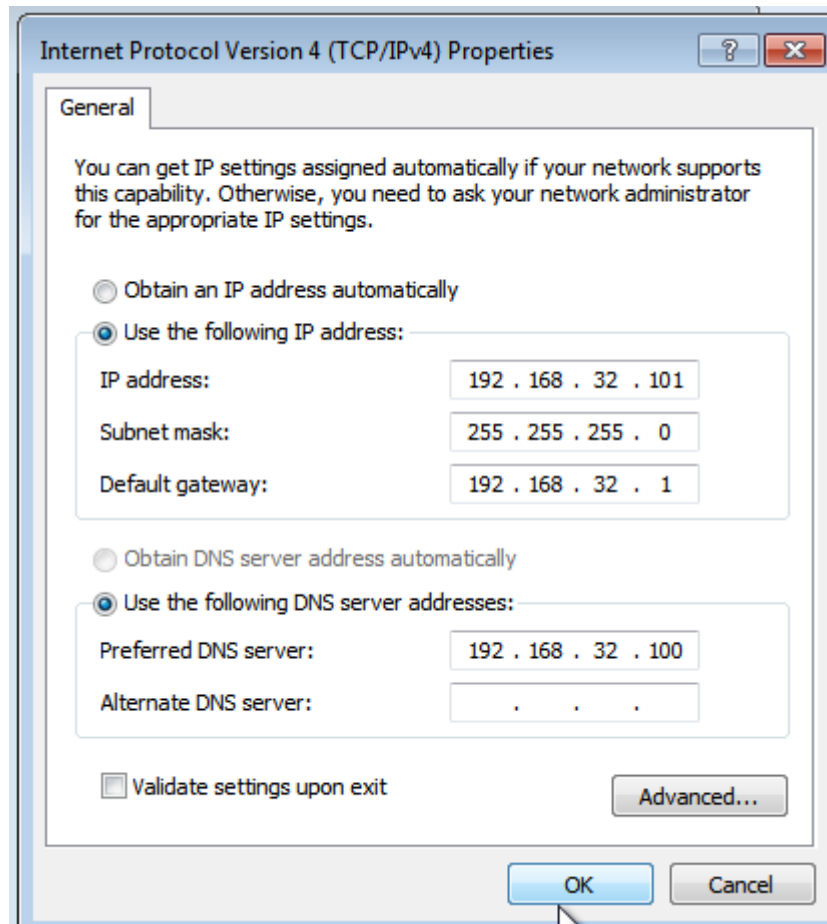
una volta nel Network and Sharing Center, con tasto destro andremo sul Local Area Connection e scegliamo Properties:



ci aprirà una nuova finestra dove selezioneremo Internet Protocol Version 4 (TCP/IPv4) e poi clicchiamo sul Properties:

e modifichiamo l'IP, gateway, DNS:





Fatto questo torniamo sul Kali Terminal dove scriveremo: "sudo nano /etc/inetsim/inetsim.conf" qui possiamo configurare il nostra server virtuale Inetsim. Andremo Semplicemente a togliere le hashtag "#" per modificare le impostazioni default in quelle desiderate. Nel nostro caso andremo a impostare l'IP del Host per associarlo al sito, quindi Kali (service_bind_address 192.168.32.100)

```
#####
# service_bind_address
#
# IP address to bind services to
#
# Syntax: service_bind_address <IP address>
#
# Default: 127.0.0.1
#
service_bind_address 192.168.32.100
```

poi andremo ad aggiungere il nostro domain name e l'IP nella mappatura:

```
#####
# dns_static
#
# Static mappings for DNS
#
# Syntax: dns_static <fqdn hostname> <IP address>
#
# Default: none
#
dns_static epicode.internal 192.168.32.100
#dns_static ns1.foo.com 10.70.50.30
#dns_static ftp.bar.net 10.10.20.30
```

Fatto questo possiamo fare una prova sul Command Prompt del Win7 dando il

```
C:\Users\UulnWindows7>ping epicode.internal

Pinging epicode.internal [192.168.32.100] with 32 bytes of data:
Reply from 192.168.32.100: bytes=32 time<1ms TTL=64
Reply from 192.168.32.100: bytes=32 time<1ms TTL=64
Reply from 192.168.32.100: bytes=32 time<1ms TTL=64
Reply from 192.168.32.100: bytes=32 time<1ms TTL=64

Ping statistics for 192.168.32.100:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

comando -"ping epicode.internal"

se apriamo il Wireshark sul Kali, e iniziamo ad ascoltare la connessione:

The image shows two overlapping windows. The background window is Wireshark, displaying a packet capture on interface eth0. It shows several ICMP Echo (ping) requests and replies between 192.168.32.100 and 192.168.32.101. The foreground window is the Windows 7 Command Prompt, showing the execution of the 'ping epicode.internal' command, which results in four successful replies with 0ms round trip times. Below the Command Prompt, the 'getmac' command is executed, showing the physical address 08-00-27-FC-7B-53 and the transport name \Device\NPF{...}.

Con il Wireshark possiamo identificare anche l'indirizzo mac del client (win7):

qui sotto abbiamo selezionato la richiesta ICMP:

Ethernet II, Src: PcsCompu_fc:7b:53 (08:00:27:fc:7b:53), Dst: PcsCompu_c0:5a:c9 (08:00:27:c0:5a:c9)

qui sotto abbiamo il MAC del Win7(sotto il Physical Address)

Physical Address	Transport Name
08-00-27-FC-7B-53	\Device\NPF{7DE58B8A-6E23-4EAD-9D07-FE63EB1F78FA}

C:\Users\UulnWindows7>

qui sotto abbiamo il MAC del Kali: eseguito ifconfig nel kali terminal per sapere l'indirizzo MAC

ether 08:00:27:c0:5a:c9 txqueuelen 1000 (Ethernet)

Andiamo adesso ad avviare il nostro server virtuale sul Kali, scrivendo sul terminal -"sudo inetsim", scriviamo la password se necessario, aspettiamo finché non compare la scritta "Simulation running", questo Terminal non lo chiudiamo e non lo sospendiamo finché non decidiamo che il server non ci serve più, per terminarlo basterà fare Ctrl+C.

Da qui andremo sul Win7, apriremo Internet Explorer, e nella barra URL scriviamo "epicode.internal"

The image shows a Wireshark packet capture of an HTTP GET request. The packet list on the left shows packet 218 as a GET request to http://epicode.internal/. The packet details pane on the right shows the Hypertext Transfer Protocol section. The packet bytes pane at the bottom shows the raw data of the request. To the right, a screenshot of Internet Explorer shows the default HTML page for InetSim, with the address bar displaying http://epicode.internal/.

No.	Time	Source	Destination	Protocol	Length	Info
122	1908.7077587...	192.168.32.100	192.168.32.101	ICMP	74	Echo (ping) reply id=0x0001, seq=54/13824, ttl=64 (request in 121)
123	1909.7080611...	192.168.32.101	192.168.32.100	ICMP	74	Echo (ping) request id=0x0001, seq=55/14080, ttl=128 (reply in 124)
124	1909.7080793...	192.168.32.100	192.168.32.101	ICMP	74	Echo (ping) reply id=0x0001, seq=55/14080, ttl=64 (request in 123)
125	1910.7073737...	192.168.32.101	192.168.32.100	ICMP	74	Echo (ping) request id=0x0001, seq=56/14336, ttl=128 (reply in 126)
126	1910.7073919...	192.168.32.100	192.168.32.101	ICMP	74	Echo (ping) reply id=0x0001, seq=56/14336, ttl=64 (request in 125)
144	1780.3377765...	192.168.32.101	192.168.32.100	DNS	82	Standard query 0x8a2b A ieonline.microsoft.com
145	1780.3428885...	192.168.32.100	192.168.32.101	DNS	98	Standard query response 0x8a2b A ieonline.microsoft.com
146	1782.3541236...	192.168.32.101	192.168.32.100	DNS	76	Standard query 0x0e7b A go.microsoft.com
147	1782.3591777...	192.168.32.100	192.168.32.101	DNS	92	Standard query response 0x0e7b A go.microsoft.com
162	2318.7743535...	192.168.32.101	192.168.32.100	DNS	72	Standard query 0xc784 A api.bing.com
164	2318.7844451...	192.168.32.100	192.168.32.101	DNS	88	Standard query response 0xc784 A api.bing.com
167	2321.9149407...	192.168.32.101	192.168.32.100	DNS	72	Standard query 0x7e3f A www.bing.com
169	2321.9200099...	192.168.32.100	192.168.32.101	DNS	88	Standard query response 0x7e3f A www.bing.com
215	2371.5198327...	192.168.32.101	192.168.32.100	TCP	66	66 80 → 49622 [SYN] Seq=0 Win=8192 Len=0
216	2371.5198546...	192.168.32.100	192.168.32.101	TCP	66	80 80 → 49622 [SYN, ACK] Seq=0 Ack=1 Win=0
217	2371.5200156...	192.168.32.101	192.168.32.100	TCP	60	60 49622 → 80 [ACK] Seq=1 Ack=1 Win=65535
218	2371.5201985...	192.168.32.101	192.168.32.100	HTTP	395	GET / HTTP/1.1
219	2371.5202056...	192.168.32.100	192.168.32.101	TCP	54	80 80 → 49622 [ACK] Seq=1 Ack=252 Win=0
220	2371.5321369...	192.168.32.100	192.168.32.101	TCP	204	80 80 → 49622 [PSH, ACK] Seq=1 Ack=252 Win=0
221	2371.5324470...	192.168.32.101	192.168.32.100	TCP	60	49622 → 80 [ACK] Seq=252 Ack=151 Win=0
222	2371.5324578...	192.168.32.100	192.168.32.101	HTTP	312	HTTP/1.1 200 OK (text/html)
223	2371.5327253...	192.168.32.101	192.168.32.100	TCP	60	49622 → 80 [ACK] Seq=252 Ack=409 Win=0
224	2371.5338118...	192.168.32.101	192.168.32.100	TCP	60	49622 → 80 [FIN, ACK] Seq=252 Ack=409 Win=0
225	2371.5339381...	192.168.32.100	192.168.32.101	TCP	54	80 80 → 49622 [FIN, ACK] Seq=409 Ack=252 Win=0
226	2371.5341493...	192.168.32.101	192.168.32.100	TCP	60	49622 → 80 [ACK] Seq=253 Ack=410 Win=0

Frame 218: 305 bytes on wire (2440 bits), 305 bytes captured (2440 bits) on interface eth0, id 0
Ethernet II, Src: PcsCompu_fc:7b:53 (08:00:27:fc:7b:53), Dst: PcsCompu_c0:5a:c9 (08:00:27:c0:5a:c9)
Internet Protocol Version 4, Src: 192.168.32.101, Dst: 192.168.32.100
Transmission Control Protocol, Src Port: 49622, Dst Port: 80, Seq: 1, Ack: 1, Len: 251
Hypertext Transfer Protocol

0000 08 00 27 c0 5a c9 08 00 27 fc 7b 53 08 00 45 00 ...Z...{S-E
0010 01 23 0d 82 40 00 80 06 2a 39 c0 a8 20 65 c0 a8 ...@...9...e.
0020 20 64 c1 d6 00 50 ba 33 df 55 2e 25 af f0 50 18 ...P.3.U.%-P
0030 01 00 de 6c 00 00 47 45 54 20 2f 20 48 54 54 50 ...LGE T / HTTP
0040 2f 31 2e 31 0d 0a 41 63 63 65 70 74 3a 20 74 65 /1.1Ac cept: te
0050 78 74 2f 68 74 6d 6c 2c 20 61 70 70 6c 69 63 61 xt/html, applica
0060 74 69 6f 6e 2f 78 68 74 6d 6c 2b 78 6d 6c 2c 20 tion/xht ml+xml,
0070 2a 2f 2a 0d 0a 41 63 63 65 70 74 2d 4c 61 6e 67 /*.Acc ept-Lang
0080 75 61 67 65 3a 20 69 74 2d 49 54 0d 0a 55 73 65 uage: it -IT-Use

anche qui possiamo vedere l'indirizzo MAC delle 2 macchine virtuali e anche il contenuto della richiesta, come possiamo vedere, esaminando il pacchetto del protocollo HTTP riusciamo a scoprire il contenuto(a decryptarlo)

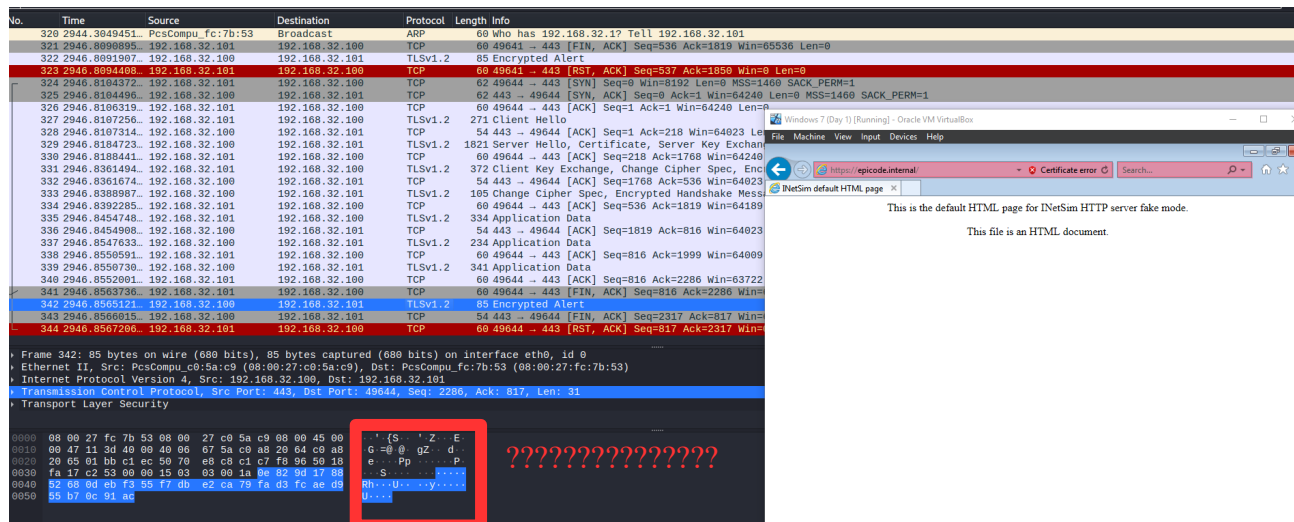
The image shows a Wireshark packet capture of an HTTP response. The packet list on the left shows packet 222 as an HTTP response. The packet details pane on the right shows the Hypertext Transfer Protocol section. The packet bytes pane at the bottom shows the raw data of the response. To the right, a screenshot of Internet Explorer shows the default HTML page for InetSim, with the address bar displaying http://epicode.internal/.

No.	Time	Source	Destination	Protocol	Length	Info
222	2371.5324578...	192.168.32.100	192.168.32.101	HTTP	312	HTTP/1.1 200 OK (text/html)

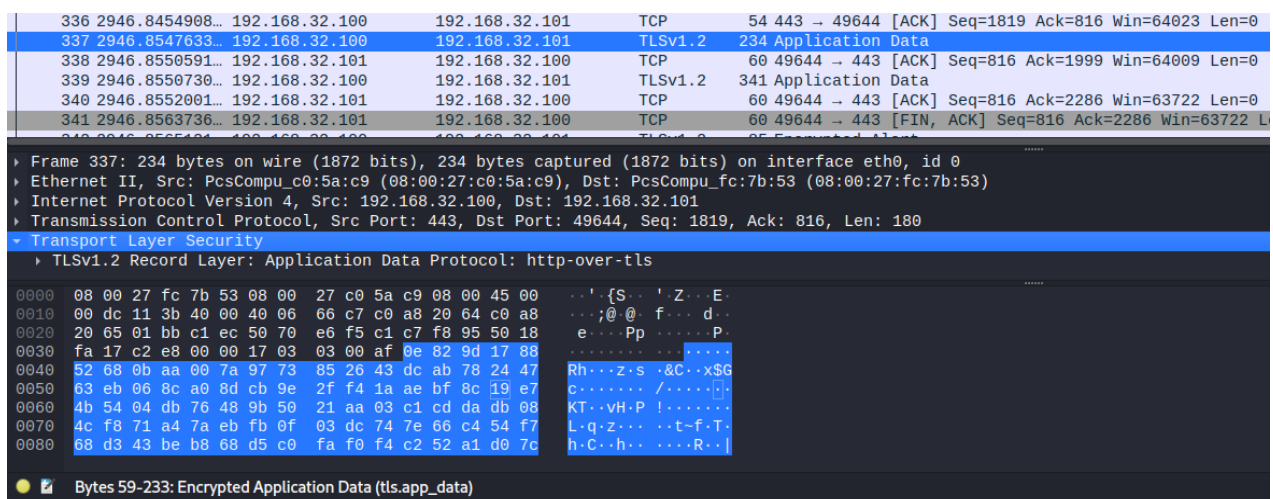
Frame 222: 312 bytes on wire (2496 bits), 312 bytes captured (2496 bits) on interface eth0, id 0
Ethernet II, Src: PcsCompu_fc:7b:53 (08:00:27:fc:7b:53), Dst: PcsCompu_c0:5a:c9 (08:00:27:c0:5a:c9)
Internet Protocol Version 4, Src: 192.168.32.100, Dst: 192.168.32.101
Transmission Control Protocol, Src Port: 80, Dst Port: 49622, Seq: 252, Ack: 1, Len: 251
Hypertext Transfer Protocol

0000 08 00 27 fc 7b 53 08 00 27 c0 5a c9 08 00 45 00 ...Z...{S-E
0010 01 2a c6 06 40 00 40 06 b1 ad c0 a8 20 64 c0 a8 ...*...@...d..
0020 20 65 00 50 c1 d6 2e 25 ad 86 ba 33 e0 50 50 18 ...e.P.%.3 PP.
0030 01 f5 c3 36 00 00 3c 68 74 6d 6c 3e 0a 20 20 3c head> <title
0040 68 65 61 64 3e 0a 20 20 20 20 3c 74 69 74 6c 65 >INetSim default
0050 3e 49 4e 65 74 53 69 6d 20 64 65 66 61 75 6c 74 HTML pa ge<titl
0060 20 48 54 4d 4c 20 70 61 67 65 3c 2f 74 69 74 6c e> </h ead> <
0070 65 3e 0a 20 20 3c 2f 68 65 61 64 3e 0a 20 20 3c body> <p></p>
0080 62 6f 64 79 3e 0a 20 20 20 20 3c 70 3e 3c 2f 70 >> <p align=
0090 3e 0a 20 20 20 20 3c 70 20 61 6c 69 67 6e 3d 22 center"> This is
00a0 63 65 6e 74 65 72 22 3e 54 68 69 73 20 69 73 20 the defa ult HTML
00b0 74 68 65 20 64 65 66 61 75 6c 74 20 48 54 4d 4c page fo r InetSi
00c0 20 70 61 67 65 20 66 6f 72 20 49 4e 65 74 53 69 m HTTP s erver fa
00d0 6d 20 48 54 54 50 20 73 65 72 76 65 72 20 66 61 ke mode. </p>
00e0 6b 65 20 6d 6f 64 65 2e 3c 2f 70 3e 0a 20 20 20 <p align="cente
00f0 20 3c 70 20 61 6c 69 67 6e 3d 22 63 65 6e 74 65 r">This file is
0100 72 22 3e 54 68 69 73 20 66 69 6c 65 20 69 73 20

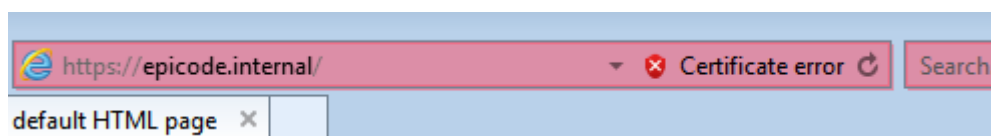
Invece se navighiamo sul [HTTPS://epicode.internal](https://epicode.internal), riusciamo sempre a vedere l'indirizzo MAC, però i contenuti sono cryptati.



Quindi abbiamo visto che il protocollo HTTPS è più sicura perchè usa la TLS(Transport Layer Security) a crittografare le normali richieste e risposte HTTP, rendendole molto più sicure.



SSL/TLS ci da anche la conferma che il website server ci dice veramente che è quello che è, in senso che non sia un impostore, perchè ha un Certificato SSL.



Andare sui siti con SSL Certificati non ci da la garanzia al 100% di sicurezza per quanto i certificati possono essere anche falsi.

28/10/2022

Filip S.