

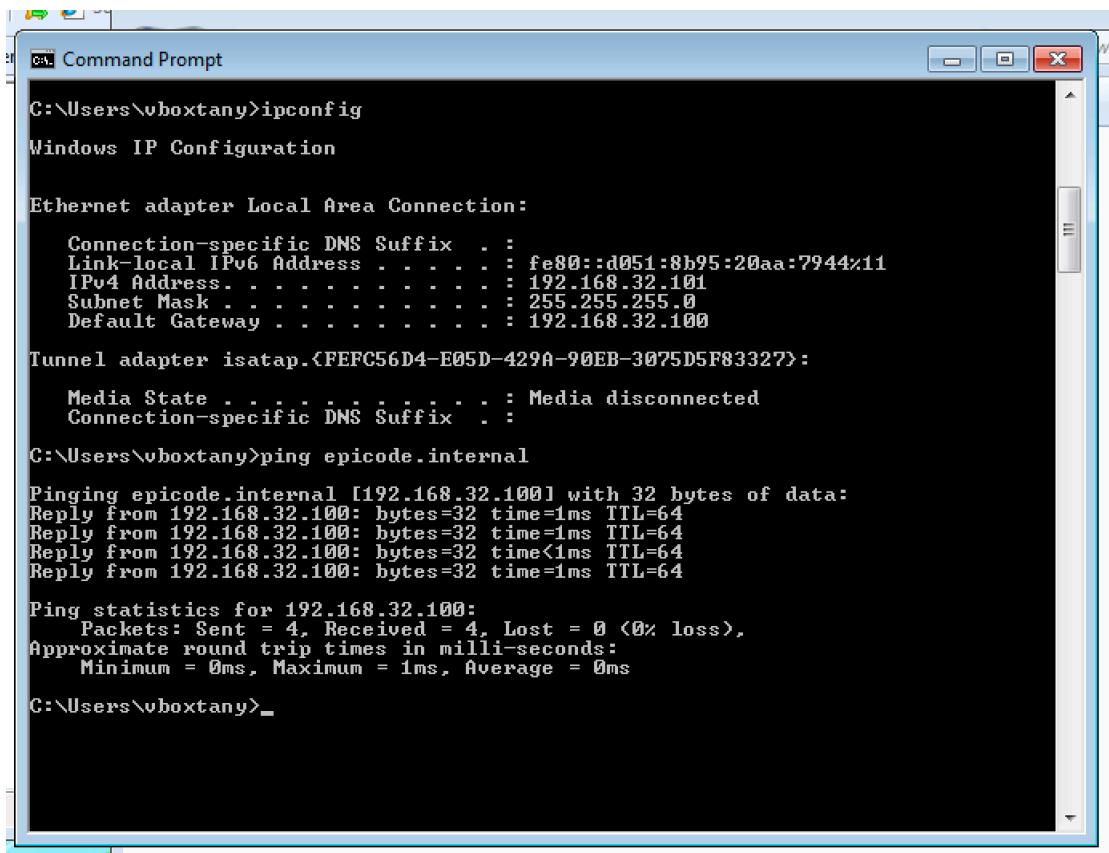
PROGETTO 1° Modulo

1) Kali Linux con IP 192.168.32.10

```
(kali㉿kali)-[~]
└─$ sudo setxkbmap it
[sudo] password for kali:

(kali㉿kali)-[~]
└─$ ip addr
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
            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 08:00:27:53:0c:ba brd ff:ff:ff:ff:ff:ff
        inet 192.168.32.100/24 brd 192.168.32.255 scope global eth0
            valid_lft forever preferred_lft forever
        inet6 fe80::a00:27ff:fe53:cba/64 scope link
            valid_lft forever preferred_lft forever
            7 0.001048749 192.168.32.100 192.168.32.101 TCP 66 49165 → 80
            192.168.32.101 TCP 66 80 → 49165
            192.168.32.100 TCP 60 49165 → 80
            192.168.32.100 HTTP 361 GET / HTTP/1.1
            192.168.32.101 TCP 54 80 → 49165
            192.168.32.101 TCP 204 80 → 49165
            192.168.32.101 HTTP 312 HTTP/1.1
(kali㉿kali)-[~] 74278
└─$
```

2) Windows7 con IP 192.168.32.101



```
C:\ Command Prompt
C:\Users\vboxtany>ipconfig
Windows IP Configuration

Ethernet adapter Local Area Connection:
  Connection-specific DNS Suffix . . . . .
  Link-local IPv6 Address . . . . . : fe80::d051:8b95:20aa:7944%11
  IPv4 Address . . . . . : 192.168.32.101
  Subnet Mask . . . . . : 255.255.255.0
  Default Gateway . . . . . : 192.168.32.100

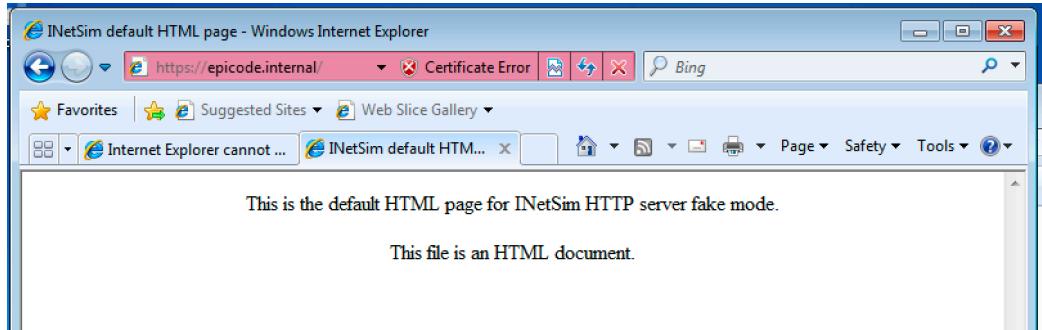
Tunnel adapter isatap.{FEFC56D4-E05D-429A-90EB-3075D5F83327}:
  Media State . . . . . : Media disconnected
  Connection-specific DNS Suffix . . . . .

C:\Users\vboxtany>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

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 = 1ms, Average = 0ms

C:\Users\vboxtany>
```

3) Richiesta tramite web browser “epicode.internal”, intercettazione con Wireshark e MAC Address di sorgente e destinazione evidenziati.
 Contenuto della richiesta HTTPS.



File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

tcp.stream eq 4

No.	Time	Source	Destination	Protocol	Length	Info
84	145.2136322107	192.168.32.101	192.168.32.100	TCP	66	49170 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=4 SACK_PERM
85	145.213718512	192.168.32.100	192.168.32.101	TCP	66	443 → 49170 [SYN, ACK] Seq=1 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM WS=128
86	145.214725582	192.168.32.101	192.168.32.100	TCP	66	49170 → 443 [ACK] Seq=1 Ack=1 Win=65700 Len=0
87	145.218217173	192.168.32.101	192.168.32.100	TLSv1	183	Client Hello
88	145.218241448	192.168.32.100	192.168.32.101	TCP	54	443 → 49170 [ACK] Seq=1 Ack=130 Win=64128 Len=0
89	145.259472574	192.168.32.100	192.168.32.101	TLSv1	1373	Server Hello, Certificate, Server Key Exchange, Server Hello Done
90	145.265432895	192.168.32.101	192.168.32.100	TLSv1	188	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
91	145.265469125	192.168.32.100	192.168.32.101	TCP	54	443 → 49170 [ACK] Seq=1320 Ack=264 Win=64128 Len=0
92	145.266003629	192.168.32.100	192.168.32.101	TLSv1	113	Change Cipher Spec, Encrypted Handshake Message
97	145.472324529	192.168.32.100	192.168.32.101	TCP	113	[TCP Retransmission] 443 → 49170 [PSH, ACK] Seq=1320 Ack=264 Win=64128 Len=59
98	145.472791992	192.168.32.101	192.168.32.100	TCP	66	49170 → 443 [ACK] Seq=264 Ack=1379 Win=64320 Len=0 SLE=1320 SRE=1379
110	151.411614108	192.168.32.101	192.168.32.100	TCP	66	49170 → 443 [FIN, ACK] Seq=264 Ack=1379 Win=64320 Len=0
111	151.411730263	192.168.32.100	192.168.32.101	TLSv1	91	Encrypted Alert
112	151.412049289	192.168.32.101	192.168.32.100	TCP	60	49170 → 443 [RST, ACK] Seq=265 Ack=1416 Win=0 Len=0

```
> Frame 111: 91 bytes on wire (728 bits), 91 bytes captured (728 bits) on interface eth0, id 0
  Ethernet II, Src: PcsCompu_53:0c:ba (08:00:27:53:0c:ba), Dst: PcsCompu_10:02:a6 (08:00:27:10:02:a6)
  > Internet Protocol Version 4, Src: 192.168.32.100, Dst: 192.168.32.101
  > Transmission Control Protocol, Src Port: 443, Dst Port: 49170, Seq: 1379, Ack: 265, Len: 37
  > Transport Layer Security
```

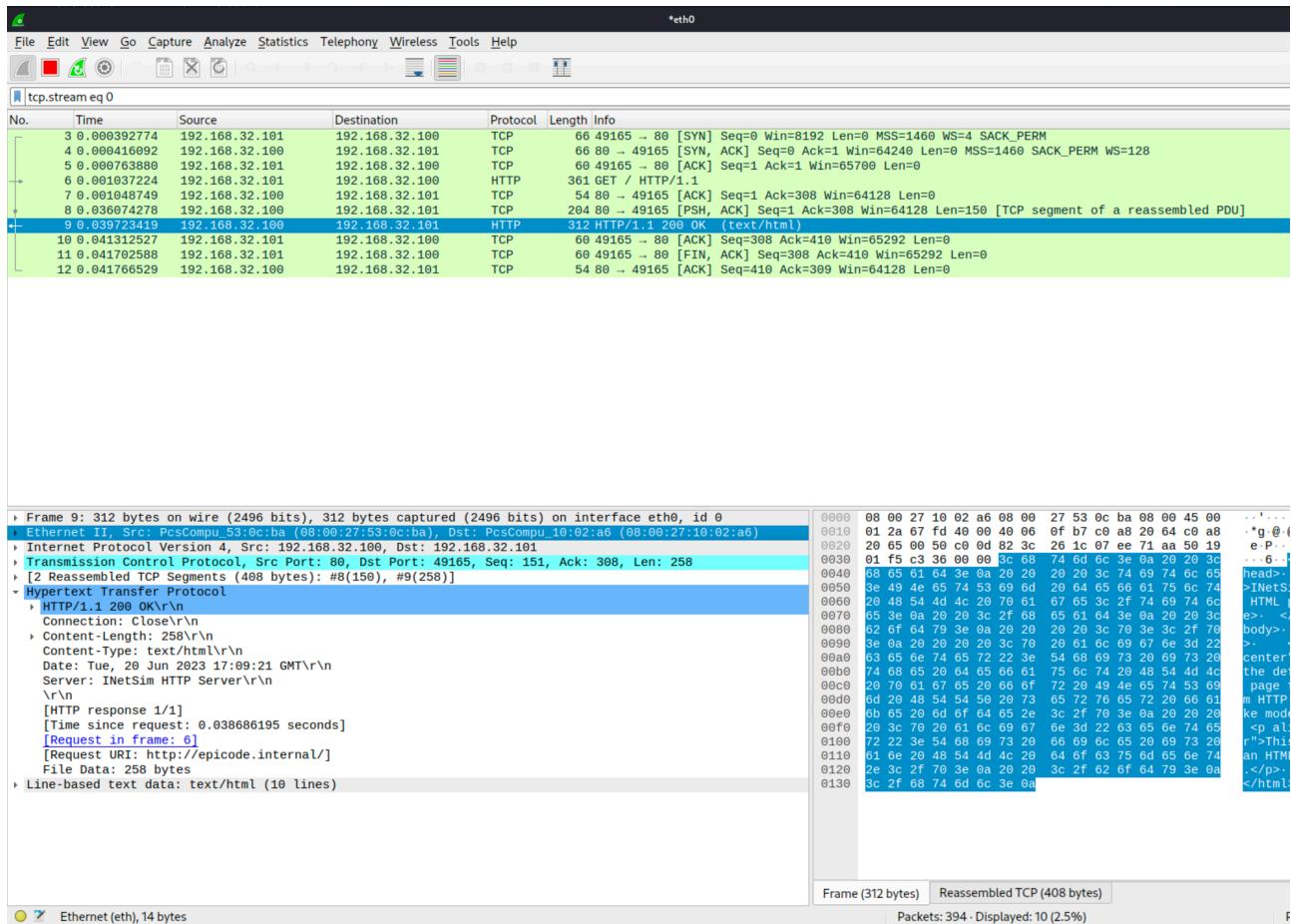
0000	08 00 27 10 02 a6 08 00 27 53 0c ba 08 00 45 00
0010	00 4d 73 2a 49 00 00 06 05 67 c0 a8 20 04 c0 a8
0020	20 65 01 bb c0 12 82 de 45 73 df a8 b0 03 50 18
0030	01 f5 c2 59 00 00 15 03 01 00 20 ac 26 b1 31 7c
0040	66 c6 88 a0 aa 92 0b ed 06 d6 be 8c 33 f1 b3 03
0050	70 ca af 8c e3 d8 01 c1 fe a7 c1

Wireshark - Packet 68 · eth0

Frame 68: 395 bytes on wire (3160 bits), 395 bytes captured (3160 bits) on interface eth0, id 0
Ethernet II, Src: PcsCompu_10:02:a6 (08:00:27:10:02:a6), Dst: PcsCompu_53:0c:ba (08:00:27:53:0c:ba)
> Destination: PcsCompu_53:0c:ba (08:00:27:53:0c:ba)
> Source: PcsCompu_10:02:a6 (08:00:27:10:02:a6)
Type: IPv4 (0x0800)
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: 381
Identification: 0x0244 (580)
> 010. = Flags: 0x2, Don't fragment
...0 0000 0000 0000 = Fragment Offset: 0
Time to Live: 128
Protocol: TCP (6)
Header Checksum: 0x351d [validation disabled]
[Header checksum status: Unverified]
Source Address: 192.168.32.101
Destination Address: 192.168.32.100
Transmission Control Protocol, Src Port: 49188, Dst Port: 443, Seq: 296, Ack: 1379, Len: 341
> Transport Layer Security

0000	08 00 27 53 0c ba 08 00 27 10 02 a6 08 00 45 00
0010	'S E.

4) Richiesta con server HTTP.



Le differenze con il protocollo HTTP e quello HTTPS sono che quest'ultimo è la versione sicura e rendere cifrato il traffico web.

Quindi vediamo che con la richiesta HTTP tutto il traffico è in chiaro, possiamo vedere tutti i pacchetti generati e il contenuto, mentre con la richiesta HTTPS il traffico catturato non è visibile, ci sono solo pacchetti generici di cui non sappiamo il contenuto, quindi mostrano solo dati criptati.