REPORT SIMULAIZONE

Ho impostato l'IP 192.168.32.100 su Kali Linux tramite il comando *sudo nano /etc/network/interfaces*, salvato le modifiche e riavviato la macchina.

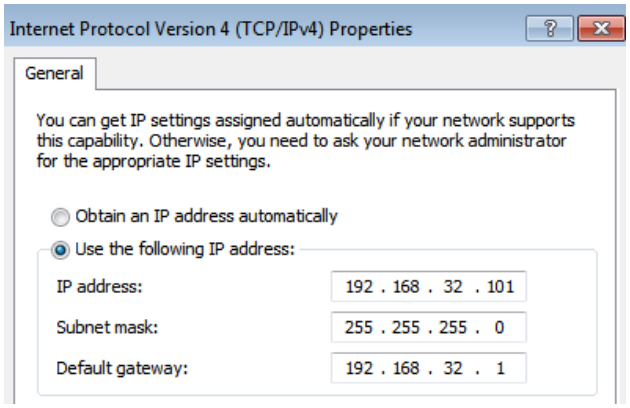
```
source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

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

```
    link/ether 08:00:27:c7:e1:36 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
```

Per impostare l'IP 192.168.32.101 su Windows 7 sono andata a cambiare le proprietà della Local Area Connection (percorso: *Control Panel\Network and Internet\Network Connections*) agendo nelle proprietà *dell'Internet Protocol Version 4 (TCP/IPv4).*




```
C:\Users\Monia>ipconfig

Windows IP Configuration

Ethernet adapter Local Area Connection:

   Connection-specific DNS Suffix  . :
   Link-local IPv6 Address . . . . . : fe80::a4df:b307:1ec1:a6e7%11
   IPv4 Address. . . . . . . . . . . : 192.168.32.101
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 192.168.32.1
```

Da specificare che entrambe le macchine lavorano in assenza di connessione di rete, impostata su *rete interna*, e a Windows 7 sono disabilitati i firewall.

Provando su entrambi i sistemi si riescono a pingare senza problemi.

```
┌──(kali㉿kali)-[~]
└─$ ping 192.168.32.101
PING 192.168.32.101 (192.168.32.101) 56(84) bytes of data.
64 bytes from 192.168.32.101: icmp_seq=1 ttl=128 time=1.03 ms
64 bytes from 192.168.32.101: icmp_seq=2 ttl=128 time=0.382 ms
64 bytes from 192.168.32.101: icmp_seq=3 ttl=128 time=0.552 ms
```

```
C:\Users\Monia>ping 192.168.32.100

Pinging 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
```

Poi ho avviato InetSim su Kali e ho inserito i dati per preparare la simulazione, con il comando *sudo mousepad /etc/inetsim/inetsim.conf* andando poi a configurare i dati di *Service_bind_address*, *DNS_default_IP* e *DNS_static*, savato e riavviato.

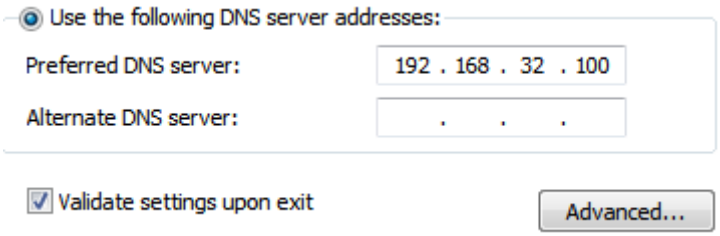
```
┌──(kali㉿kali)-[~]
└─$ sudo mousepad /etc/inetsim/intesim.conf
```

```
69 service_bind_address 192.168.32.100
70
```

```
207 dns_default_ip 192.168.32.100
208
```

```
242 dns_static epicode.internal 192.168.32.100
243
```

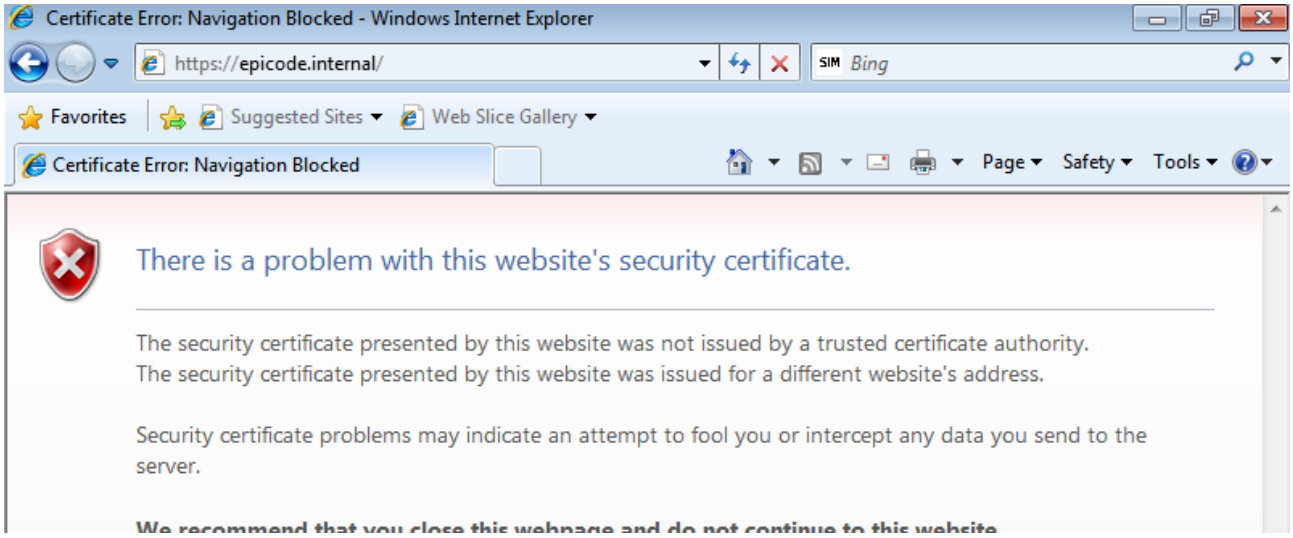
Tornando nelle impostazioni di Windows 7, già precedentemente modificate, ho cambiato il *Preferred DNS server* in 192.168.32.100, e riavviato la macchina.



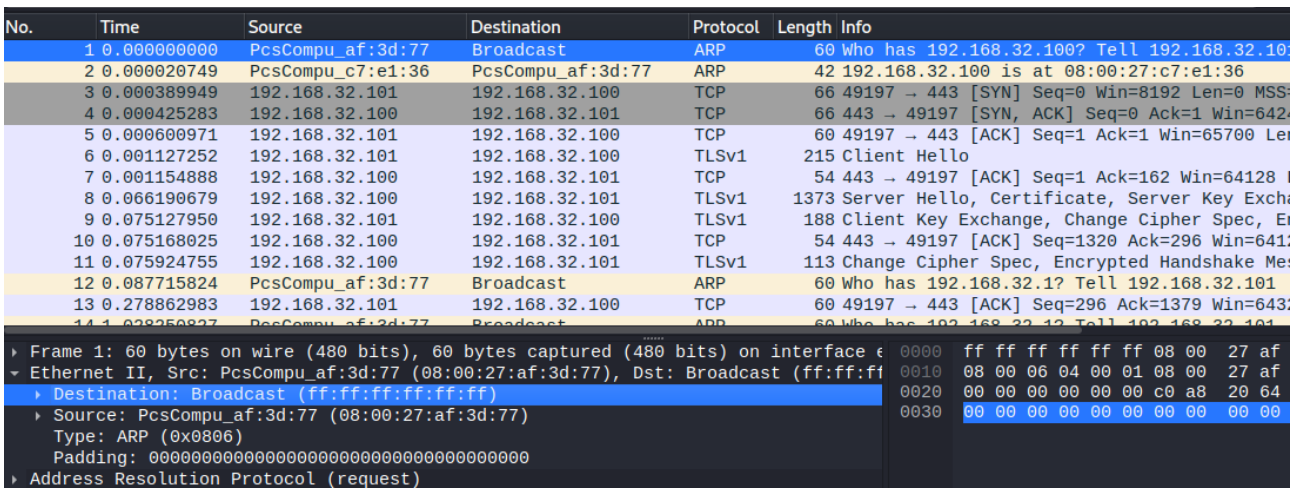
Su Kali ho avviato la simulazione con il comando *sudo inetsim*.

```
┌──(kali㉿kali)-[~]
└─$ sudo inetsim
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 2069) ===
Session ID:     2069
```

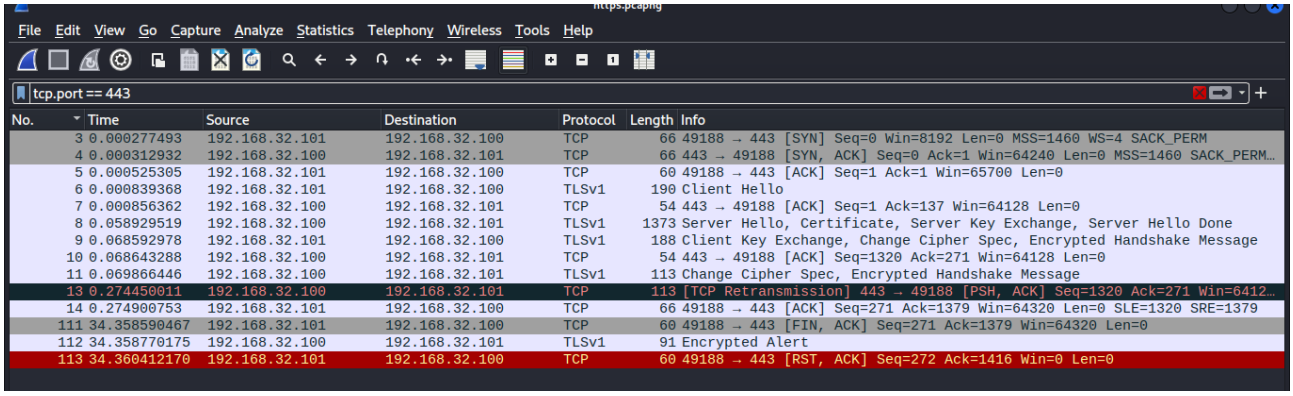
Su kali avvio il programma Wireshark e subito dopo su Windows7 inserisco l'indirizzo https://epi-code.internal che segnale un problema di sicurezza in quanto non presente il certificato.



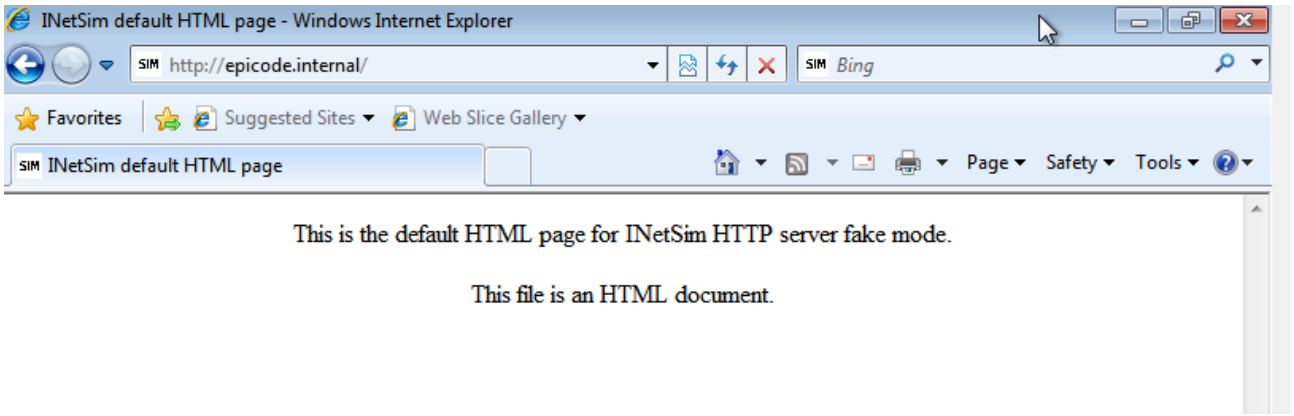
Quindi su Wireshark, un packet sniffer cioè un programma che analizza "catturando" i pacchetti che passano sulla rete, si può visualizzare l'indirizzo MAC selezionando il frame n.01 dove con "Who has 192.168.32.100? Tell 192.168.32.101" si chiede *Chi è il server e di dare risposta al Client*, nella sezione sottostante ci viene quindi indicato il MAC in quanto il protocollo ARP è quello incaricato di assegnarlo.



Filtrando, invece, tramite il numero della porta 443, cioè la porta dedicata all'https, vediamo sia la procedura di Three way handshake nel quale il client invia un pacchetto dati SYN, gli viene risposto con una ricevuta SYN/ACK dal server e il client risponde di nuovo con un pacchetto ACK, da qui si crea la connessione, che il protocollo TLSv1 che indica la crittografia prevista per l'https.



Ripetendo il procedimento ma inserendo sul browser di Windows 7 l'indirizzo http://epicode.internal accertando il funzionamento del collegamento.



In questo caso nella cattura di Wideshark invece del protocollo TLSv1 vediamo il protocollo HTTP che non prevede la crittografia.