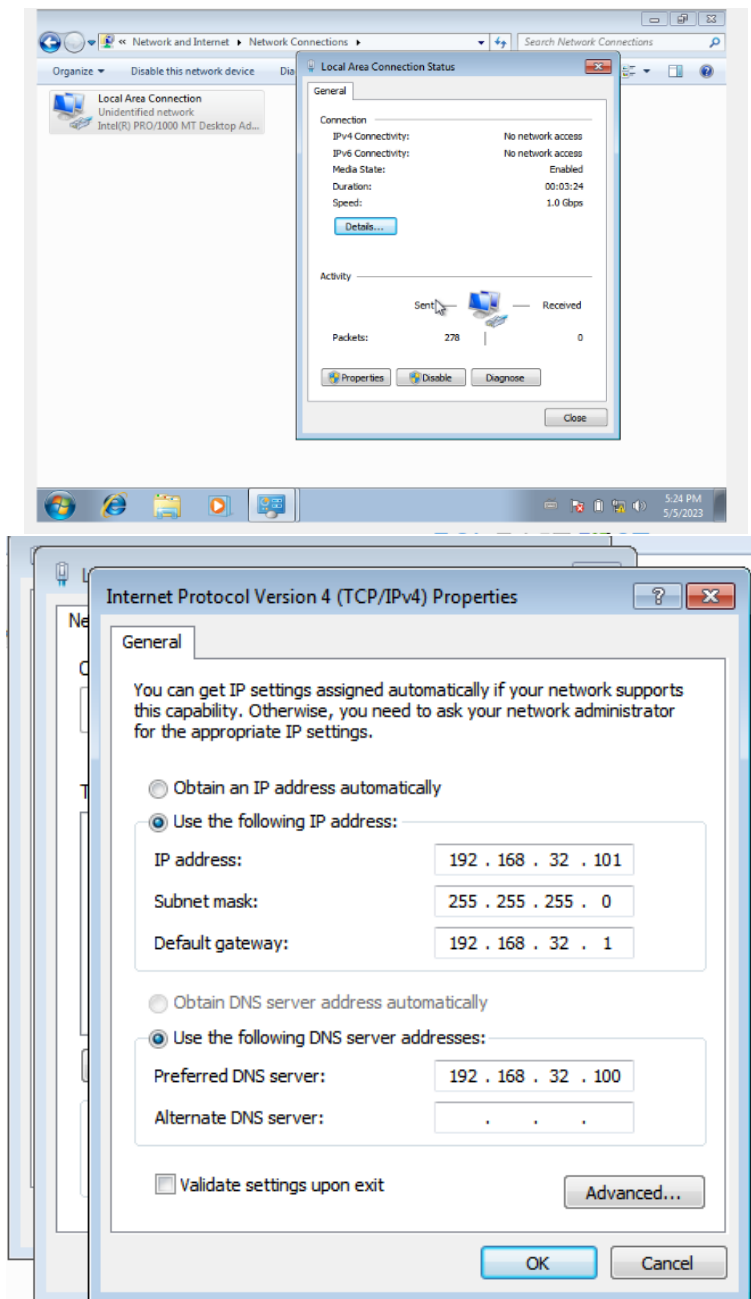


ESERCIZIO PRATICO EPICODE

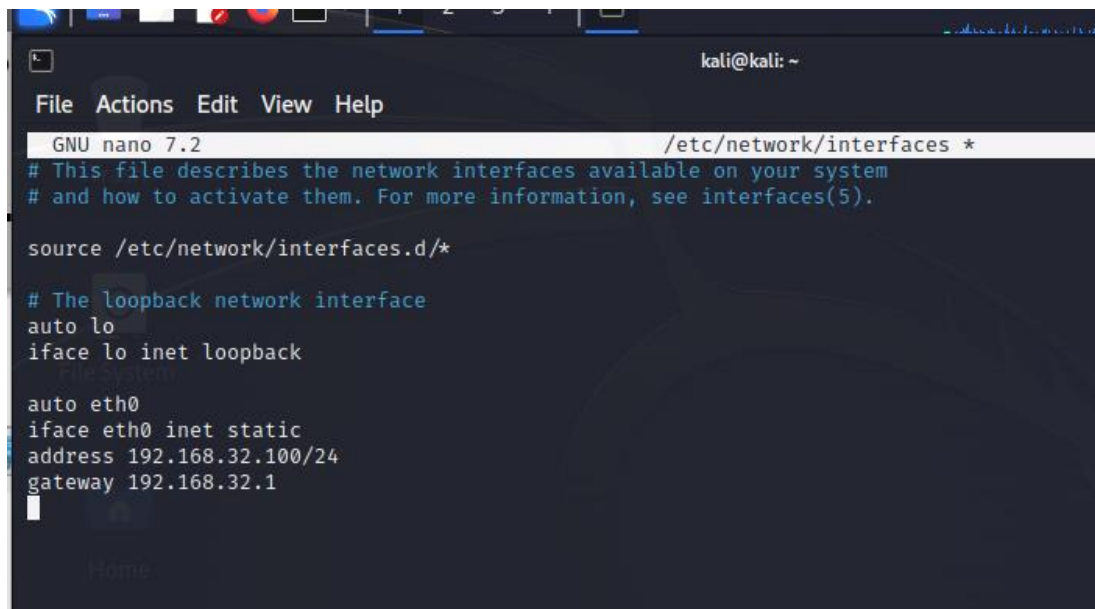
Punto 1. Variazione indirizzi Ip sulle due Macchine.

Tramite le impostazioni di rete di **Windows 7**, sono andato sulle proprietà dell'ipv4, e ho modificato l'indirizzo Ip come richiesto dall'esercizio in **192.168.32.101**;



Per comodità ho anche già impostato l'indirizzo Ip del Dns così da non doverlo rifare in seguito.

Per **Kali Linux** ho fatto la stessa operazione tramite terminale, e modificato l'indirizzo Ip della rete mantenendolo **statico** in **192.168.32.100**;

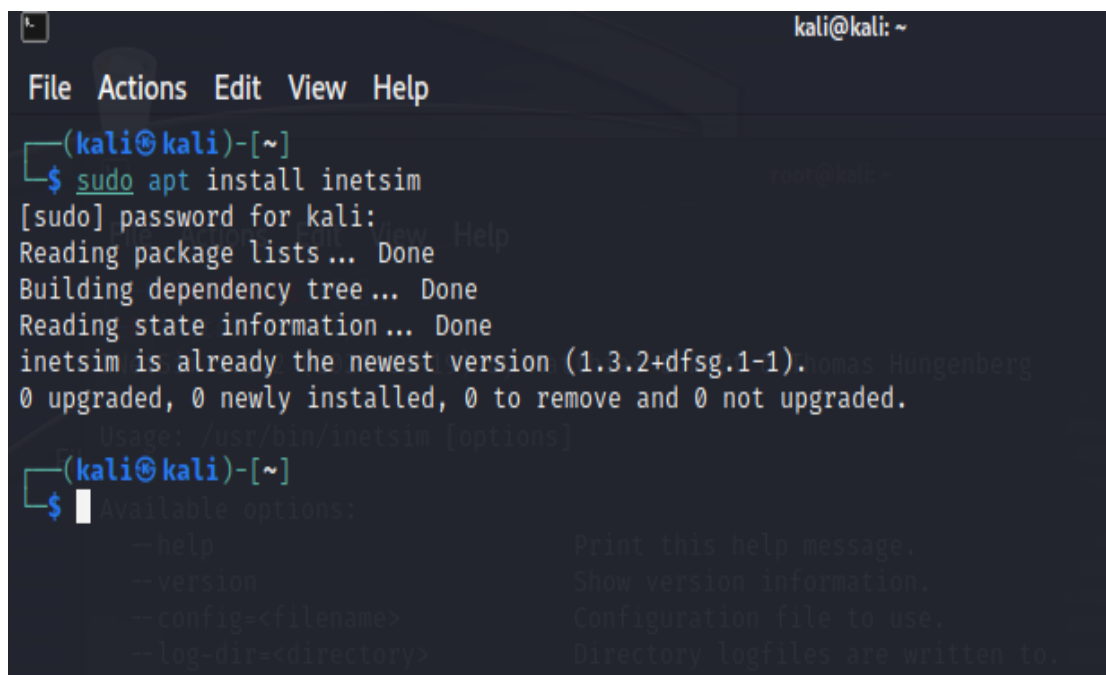


```
kali@kali: ~  
File Actions Edit View Help  
GNU nano 7.2 /etc/network/interfaces *  
# This file describes the network interfaces available on your system  
# and how to activate them. For more information, see interfaces(5).  
  
source /etc/network/interfaces.d/*  
  
# The loopback network interface  
auto lo  
iface lo inet loopback  
  
auto eth0  
iface eth0 inet static  
address 192.168.32.100/24  
gateway 192.168.32.1
```

In seguito per verificare il funzionamento ho fatto un **Ping** da Windows a Kali e viceversa;

Punto 2. Download di Inetsim su Kali e configurazione di un server Dns e uno Https.

Ho trovato il file di installazione di inetsim già presente su **Kali**, di conseguenza non ho dovuto scaricarlo, ho solamente utilizzato il comando **sudo apt install inetsim**, per installarlo sulla macchina;



```
kali@kali: ~  
File Actions Edit View Help  
  
(kali@kali)-[~]  
$ sudo apt install inetsim  
[sudo] password for kali:  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
inetsim is already the newest version (1.3.2+dfsg.1-1).  
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.  
Usage: /usr/bin/inetsim [options]  
  
(kali@kali)-[~]  
$ Available options:  
-help Print this help message.  
-version Show version information.  
-config=<filename> Configuration file to use.  
-log-dir=<directory> Directory logfiles are written to.
```

Successivamente come prima cosa ho modificato i parametri di Inetsim con il comando **nano /etc/inetsim/inetsim.conf**.

Poi dallo stesso menu di configurazione ho attivato anche il server **https**.

```
#####
#
# INetSim configuration file
#
#####

#####
# Main configuration
#####

#####
# start_service
#
# The services to start
#
# Syntax: start_service <service name>
#
# Default: none
#
# Available service names are:
# dns, http, smtp, pop3, tftp, ftp, ntp, time_tcp,
# time_udp, daytime_tcp, daytime_udp, echo_tcp,
# echo_udp, discard_tcp, discard_udp, quotd_tcp,
# quotd_udp, chargen_tcp, chargen_udp, finger,
# ident, syslog, dummy_tcp, dummy_udp, smtps, pop3s,
# ftps, irc, https
#
```

```
start_service dns
#start_service http
start_service https
```

Punto 3. Configurazione Dns

Per configurare il **Dns** in modo che risponda all'indirizzo **Ip** richiesto e cambiato il nome default del dominio in **epicode.internal**;

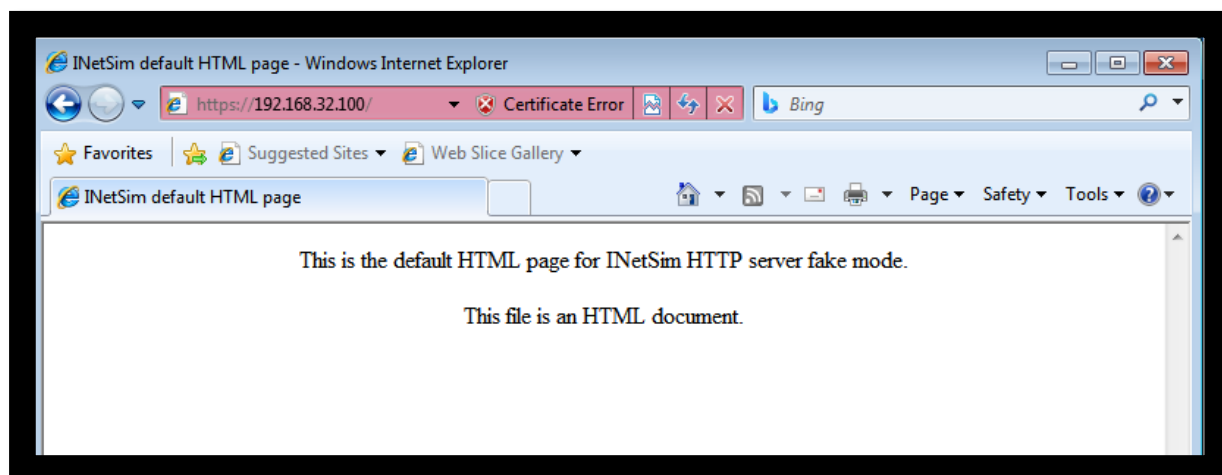
```
# Syntax: dns_default_ip <IP address>
#
# Default: 127.0.0.1
#
dns_default_ip 192.168.32.100

#####
# dns_default_hostname
#
# Default hostname to return with DNS replies
#
# Syntax: dns_default_hostname <hostname>
#
# Default: www
#
#dns_default_hostname somehost

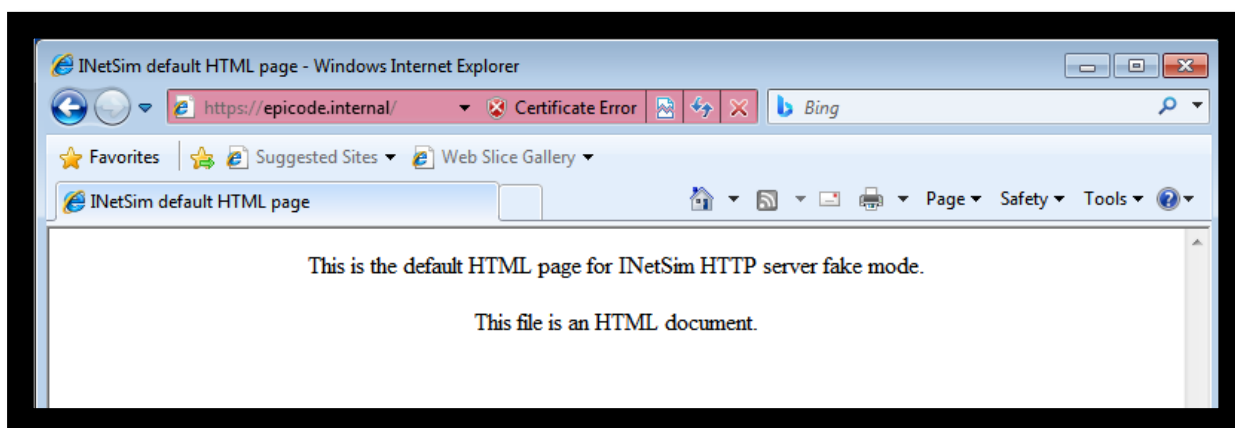
#####
# dns_default_domainname
#
# Default domain name to return with DNS replies
#
# Syntax: dns_default_domainname <domain name>
#
# Default: inetsim.org
#
dns_default_domainname epicode.internal

#####
# dns_static
#
```

Dopo aver configurato l'indirizzo Ip e il Dns, il sistema Windows si collega correttamente al nostro **falso server Https**, digitando l'**ip** :



E anche digitando sulla barra di ricerca **epicode.internal**:



Punto 4. Catture con Wireshark

Per provare la cattura dei pacchetti ho per prima cosa aperto **Wireshark** su Kali linux, poi ho selezione la cattura dei pacchetti sulla porta **eth0**, di seguito ho riaperto il browser su Windows e riprovato a collegarmi a **epicode.internal** quelle di seguito sono le foto dei risultati delle catture di **Wireshark**;

34	10.790086604	192.168.32.101	192.168.32.255	NBNS	92 Name query NB WPAD<00>
35	11.549845642	192.168.32.101	192.168.32.100	TCP	66 49279 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM
36	11.549940091	192.168.32.100	192.168.32.101	TCP	54 80 → 49279 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
37	12.072232727	192.168.32.101	192.168.32.100	TCP	66 [TCP Retransmission] [TCP Port numbers reused] 49279 → 80 [SYN]
38	12.072261136	192.168.32.100	192.168.32.101	TCP	54 80 → 49279 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
39	12.588178408	192.168.32.101	192.168.32.100	TCP	62 [TCP Retransmission] [TCP Port numbers reused] 49279 → 80 [SYN]
40	12.588216718	192.168.32.100	192.168.32.101	TCP	54 80 → 49279 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
41	12.624984469	192.168.32.101	192.168.32.100	TCP	60 49278 → 443 [FIN, ACK] Seq=264 Ack=1379 Win=64320 Len=0
42	12.625826017	192.168.32.100	192.168.32.101	TLSv1	91 Encrypted Alert
43	12.626585195	192.168.32.101	192.168.32.100	TCP	60 49278 → 443 [RST, ACK] Seq=265 Ack=1416 Win=0 Len=0
44	18.585525630	192.168.32.101	192.168.32.100	TCP	66 49280 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=4 SACK_PERM
45	18.585578189	192.168.32.100	192.168.32.101	TCP	66 443 → 49280 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM
46	18.586411157	192.168.32.101	192.168.32.100	TCP	60 49280 → 443 [ACK] Seq=1 Ack=1 Win=65700 Len=0
47	18.590552360	192.168.32.101	192.168.32.100	TLSv1	215 Client Hello
48	18.590580411	192.168.32.100	192.168.32.101	TCP	54 443 → 49280 [ACK] Seq=1 Ack=162 Win=64128 Len=0
49	18.638770112	192.168.32.100	192.168.32.101	TLSv1	1373 Server Hello, Certificate, Server Key Exchange, Server Hello Do
50	18.647056369	192.168.32.101	192.168.32.100	TLSv1	188 Client Key Exchange, Change Cipher Spec, Encrypted Handshake Me
51	18.647755195	192.168.32.100	192.168.32.101	TLSv1	113 Change Cipher Spec, Encrypted Handshake Message
52	18.673568964	192.168.32.101	192.168.32.100	TCP	60 49280 → 443 [FIN, ACK] Seq=296 Ack=1379 Win=64320 Len=0
53	18.673797962	192.168.32.100	192.168.32.101	TLSv1	91 Encrypted Alert
54	18.675169141	192.168.32.101	192.168.32.100	TCP	60 49280 → 443 [RST, ACK] Seq=297 Ack=1416 Win=0 Len=0
55	18.677249546	192.168.32.101	192.168.32.100	TCP	66 49281 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=4 SACK_PERM
56	18.677274778	192.168.32.100	192.168.32.101	TCP	66 443 → 49281 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM
57	18.680085562	192.168.32.101	192.168.32.100	TCP	60 49281 → 443 [ACK] Seq=1 Ack=1 Win=65700 Len=0
58	18.680085725	192.168.32.101	192.168.32.100	TLSv1	215 Client Hello

64	18.783520255	192.168.32.100	192.168.32.101	TLSv1	235 Application Data
65	18.787726908	192.168.32.100	192.168.32.101	TLSv1	384 Application Data, Encrypted Alert
66	18.788456262	192.168.32.101	192.168.32.100	TCP	60 49281 → 443 [ACK] Seq=749 Ack=1891 Win=65700 Len=0
67	18.789163062	192.168.32.101	192.168.32.100	TCP	60 49281 → 443 [FIN, ACK] Seq=749 Ack=1891 Win=65700 Len=0
68	18.789743015	192.168.32.100	192.168.32.101	TCP	54 443 → 49281 [ACK] Seq=1891 Ack=750 Win=64128 Len=0
69	18.850568917	192.168.32.101	192.168.32.100	TCP	66 49282 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=4 SACK_PERM
70	18.850633331	192.168.32.100	192.168.32.101	TCP	66 443 → 49282 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM
71	18.851784321	192.168.32.101	192.168.32.100	TCP	60 49282 → 443 [ACK] Seq=1 Ack=1 Win=65700 Len=0
72	18.851784583	192.168.32.101	192.168.32.100	TLSv1	184 Client Hello
73	18.851969543	192.168.32.100	192.168.32.101	TCP	54 443 → 49282 [ACK] Seq=1 Ack=131 Win=64128 Len=0
74	18.862559448	192.168.32.101	192.168.32.100	TCP	66 49283 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=4 SACK_PERM
75	18.862589494	192.168.32.100	192.168.32.101	TCP	66 443 → 49283 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM
76	18.863662275	192.168.32.101	192.168.32.100	TCP	60 49283 → 443 [ACK] Seq=1 Ack=1 Win=65700 Len=0
77	18.865406420	192.168.32.100	192.168.32.101	TLSv1	184 Client Hello

93	18.937441301	192.168.32.101	192.168.32.100	TCP	66 49284 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=4 SACK_PERM
94	18.937459046	192.168.32.100	192.168.32.101	TCP	66 443 → 49284 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM
95	18.938255691	192.168.32.101	192.168.32.100	TCP	60 49283 → 443 [RST, ACK] Seq=266 Ack=1416 Win=0 Len=0
96	18.938255883	192.168.32.101	192.168.32.100	TCP	60 49282 → 443 [RST, ACK] Seq=266 Ack=1416 Win=0 Len=0
97	18.938255999	192.168.32.101	192.168.32.100	TCP	60 49285 → 443 [ACK] Seq=1 Ack=1 Win=65700 Len=0
98	18.938740364	192.168.32.101	192.168.32.100	TLSv1	216 Client Hello
99	18.938740518	192.168.32.101	192.168.32.100	TCP	60 49284 → 443 [ACK] Seq=1 Ack=1 Win=65700 Len=0
100	18.938751784	192.168.32.100	192.168.32.101	TCP	54 443 → 49285 [ACK] Seq=1 Ack=163 Win=64128 Len=0
101	18.939813503	192.168.32.101	192.168.32.100	TLSv1	216 Client Hello
102	18.939824033	192.168.32.100	192.168.32.101	TCP	54 443 → 49284 [ACK] Seq=1 Ack=163 Win=64128 Len=0
103	18.993742185	192.168.32.100	192.168.32.101	TLSv1	1373 Server Hello, Certificate, Server Key Exchange, Server Hello Do
104	18.996933472	192.168.32.100	192.168.32.101	TLSv1	1373 Server Hello, Certificate, Server Key Exchange, Server Hello Do
105	19.003754270	192.168.32.101	192.168.32.100	TLSv1	188 Client Key Exchange, Change Cipher Spec, Encrypted Handshake Me
106	19.005018395	192.168.32.100	192.168.32.101	TLSv1	113 Change Cipher Spec, Encrypted Handshake Message
107	19.006033052	192.168.32.101	192.168.32.100	TLSv1	188 Client Key Exchange, Change Cipher Spec, Encrypted Handshake Me
108	19.006055709	192.168.32.100	192.168.32.101	TCP	54 443 → 49284 [ACK] Seq=1320 Ack=297 Win=64128 Len=0
109	19.007468805	192.168.32.100	192.168.32.101	TLSv1	113 Change Cipher Spec, Encrypted Handshake Message
110	19.013633266	192.168.32.101	192.168.32.100	TCP	60 49285 → 443 [FIN, ACK] Seq=297 Ack=1379 Win=64320 Len=0
111	19.014293816	192.168.32.100	192.168.32.101	TLSv1	91 Encrypted Alert
112	19.014998325	192.168.32.101	192.168.32.100	TCP	60 49285 → 443 [RST, ACK] Seq=298 Ack=1416 Win=0 Len=0
113	19.019161966	192.168.32.101	192.168.32.100	TCP	60 49284 → 443 [FIN, ACK] Seq=297 Ack=1379 Win=64320 Len=0
114	19.019980808	192.168.32.100	192.168.32.101	TLSv1	91 Encrypted Alert
115	19.021756866	192.168.32.101	192.168.32.100	TCP	60 49284 → 443 [RST, ACK] Seq=298 Ack=1416 Win=0 Len=0

Punto 5. Conclusioni Finali

Le consegne del compito sono tutte state eseguite e la simulazione ha funzionato correttamente, ho notato che essendo un server Https falso(fake), il sistema di Windows 7 lo riconosceva e ci segnalava il sito come potenzialmente pericoloso, ma comunque cliccando su “procedi la navigazione” la pagina si apriva correttamente e il sito veniva visualizzato.

Inoltre, mi sono accorto che durante la simulazione di Wireshark, durante la raccolta dei pacchetti, il browser web mandava al nostro server su Kali dei pacchetti con richieste Ack e Kali rispondeva con i pacchetti Syn; appena ho ricaricato la pagina ho notato che i pacchetti catturati da Wireshark aumentavano sempre con altri dati inviati dal mio browser di Windows 7.

Ho anche notato che la maggioranza dei pacchetti inviati da Windows erano tramite il protocollo Tcp.