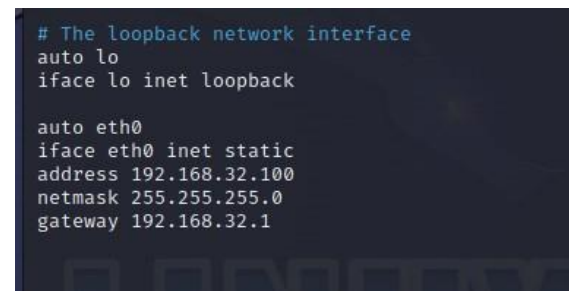
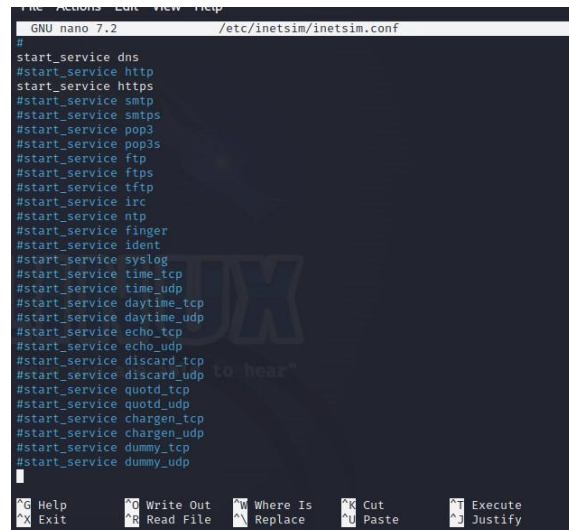
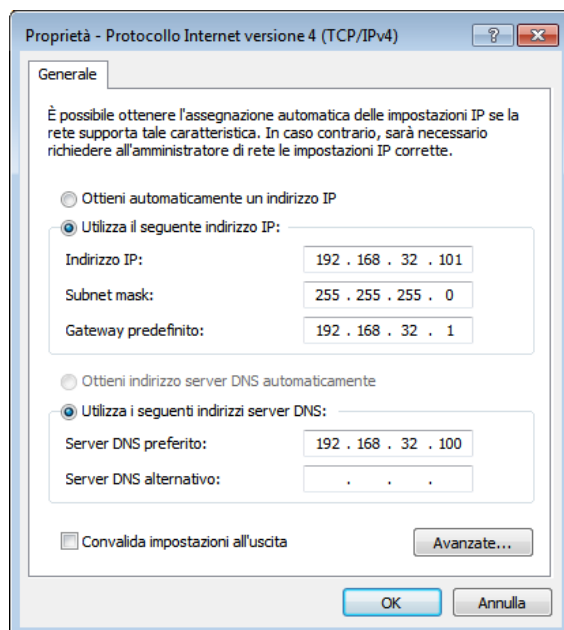


ESERCIZIO W4D4

Esecuzione:

1. Configurazione IP Kali Linux e Window:



2. Fase 1: Comunicazione tramite HTTPS

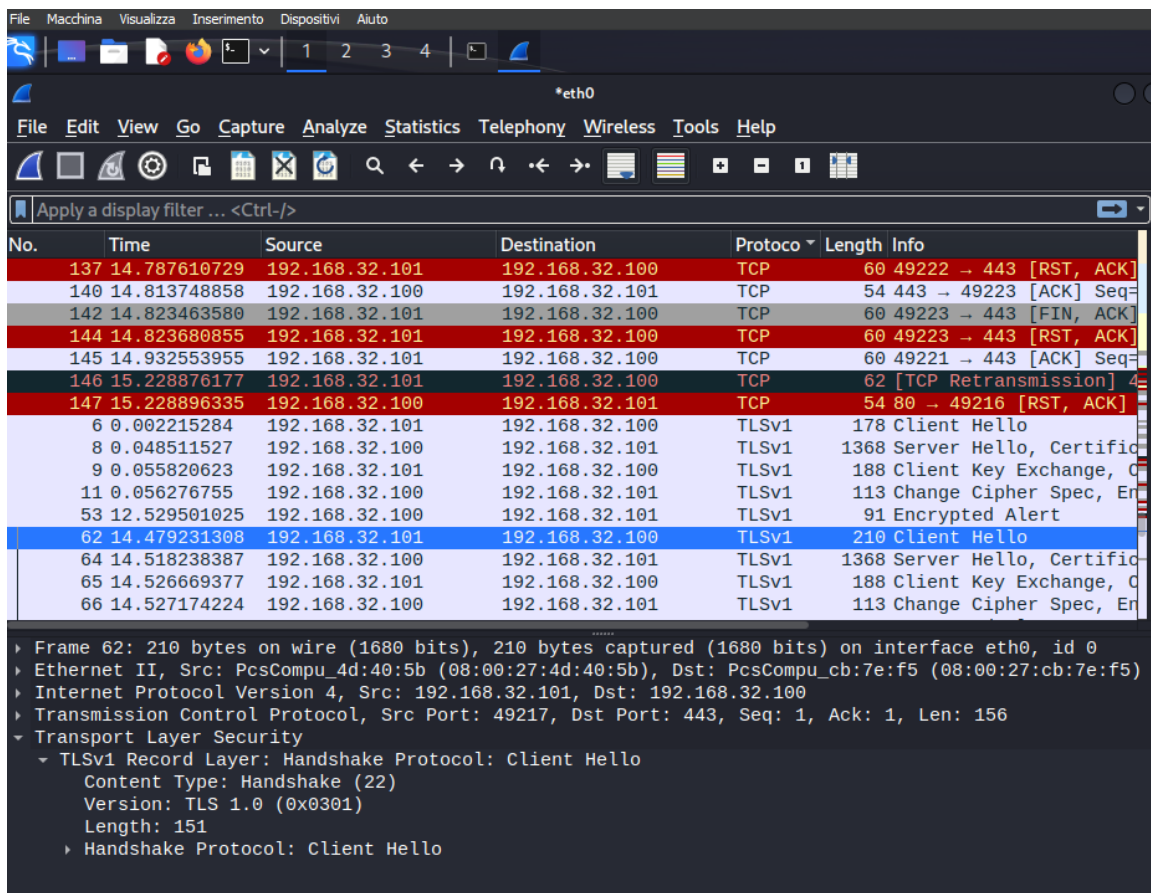
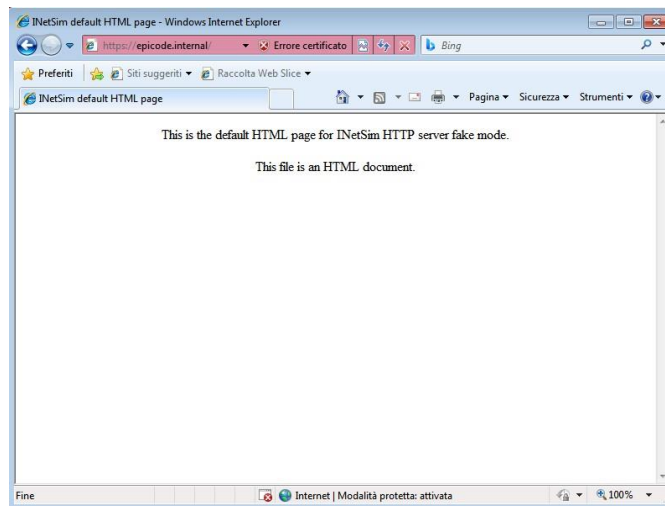
Configurazione Iniziale:

Kali Linux e Windows 7 sono connessi nella stessa rete virtuale.

Richiesta HTTPS:

Il client Windows 7 richiede tramite web browser una risorsa all'hostname epicode.internal al server HTTPS di Kali Linux.

Utilizzando Wireshark, abbiamo intercettato la comunicazione.



Fase 2: Comunicazione tramite HTTP

1. Modifica del Server:

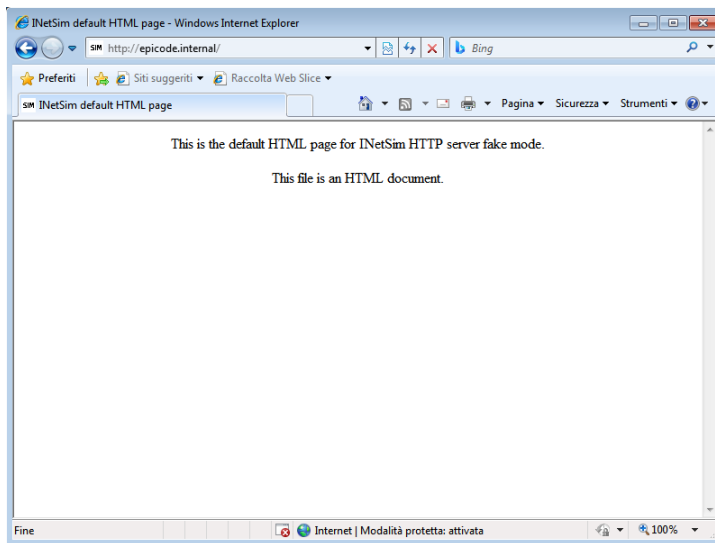
- Ho sostituito il server HTTPS con un server HTTP su Kali Linux.

2. Richiesta HTTP:

- Il client (Windows 7) ha nuovamente emesso la richiesta tramite il web browser.

3. Wireshark Capture (HTTP):

- La comunicazione è avvenuta nuovamente in maniera intercettabile.
- Indirizzo MAC di origine: [MAC_Address_Client], Indirizzo MAC di destinazione: [MAC_Address_Server].
- Contenuto della richiesta HTTP: [Contenuto_Richiesta_HTTP].



```
GNU nano 7.2 /etc/inet
#
start_service dns
start_service http
#start_service https
#start_service smtp
#start_service smtps
#start_service pop3
#start_service pop3s
#start_service ftp
#start_service ftps
#start_service tftp
#start_service irc
#start_service ntp
#start_service finger
#start_service ident
#start_service syslog
#start_service time_tcp
#start_service time_udp
#start_service daytime_tcp
#start_service daytime_udp
#start_service echo_tcp
#start_service echo_udp
#start_service discard_tcp
#start_service discard_udp
#start_service quotd_tcp
#start_service quotd_udp
#start_service chargen_tcp
#start_service chargen_udp
#start_service dummy_tcp
#start_service dummy_udp
```

No.	Time	Source	Destination	Protocol	Length	Info
25	8.516659118	PcsCompu_4d:40:5b	PcsCompu_cb:7e:f5	ARP	60	192.168.32.101 is at 6
26	11.178835619	192.168.32.101	192.168.32.100	TCP	66	49207 → 80 [SYN, Seq=6
27	11.178863596	192.168.32.100	192.168.32.101	TCP	66	80 → 49207 [SYN, ACK]
28	11.179011777	192.168.32.101	192.168.32.100	TCP	60	49207 → 80 [ACK] Seq=1
29	11.179126259	192.168.32.101	192.168.32.100	HTTP	472	GET / HTTP/1.1
30	11.179133295	192.168.32.100	192.168.32.101	TCP	54	80 → 49207 [ACK] Seq=1
31	11.191611044	192.168.32.100	192.168.32.101	TCP	204	80 → 49207 [PSH, ACK]
32	11.193420496	192.168.32.100	192.168.32.101	HTTP	312	HTTP/1.1 200 OK (text
33	11.193617343	192.168.32.101	192.168.32.100	TCP	60	49207 → 80 [ACK] Seq=4
34	11.193753285	192.168.32.101	192.168.32.100	TCP	60	49207 → 80 [FIN, ACK]
35	11.193765394	192.168.32.100	192.168.32.101	TCP	54	80 → 49207 [ACK] Seq=4
36	11.223305752	192.168.32.101	192.168.32.100	TCP	66	49208 → 443 [SYN] Seq=
37	11.223328075	192.168.32.100	192.168.32.101	TCP	54	443 → 49208 [RST, ACK]
38	11.226039562	192.168.32.101	192.168.32.100	TCP	66	49209 → 443 [SYN] Seq=
39	11.226052543	192.168.32.100	192.168.32.101	TCP	54	443 → 49209 [RST, ACK]
40	11.727235800	192.168.32.101	192.168.32.100	TCP	66	[TCP Retransmission] 4

Frame 29: 472 bytes on wire (3776 bits), 472 bytes captured (3776 bits) on interface eth0, id 0

Ethernet II, Src: PcsCompu_4d:40:5b (08:00:27:4d:40:5b), Dst: PcsCompu_cb:7e:f5 (08:00:27:cb:7e:f5)

Internet Protocol Version 4, Src: 192.168.32.101, Dst: 192.168.32.100

Transmission Control Protocol, Src Port: 49207, Dst Port: 80, Seq: 1, Ack: 1, Len: 418

Hypertext Transfer Protocol

GET / HTTP/1.1\r\n

Accept: application/x-ms-application, image/jpeg, application/xaml+xml, image/gif, image/pjpeg, app

Accept-Language: it-IT\r\n

User-Agent: Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1; WOW64; Trident/4.0; SLCC2; .NET CLR

Accept-Encoding: gzip, deflate\r\n

Host: epicode.internal\r\n

Connection: Keep-Alive\r\n

\r\n

[Full request URI: http://epicode.internal/]

[HTTP request 1/1]

[Response in frame: 32]

Configurazione dei Servizi DNS su Kali Linux:

- Ho impostato l'indirizzo IP predefinito del DNS su 192.168.32.100, in modo che Kali funga da server; l'indirizzo IP del DNS deve essere impostato con il proprio indirizzo IP.
- Il nome di dominio è "epicode.internal", come richiesto dall'esercizio.
- Nel file di configurazione del DNS, ho associato il nome di dominio all'indirizzo IP: epicode.internal.

```
File Actions Edit View Help
GNU nano 7.2

#####
# dns_default_ip
#
# Default IP address to return with DNS replies
#
# 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
#
# Static mappings for DNS
#
# Syntax: dns_static <fqdn hostname> <IP address>
#
# Default: none
#
#dns_static www.foo.com 10.10.10.10
#dns_static ns1.foo.com 10.70.50.30
dns_static epicode.internal 192.168.32.100
```

1. Analisi e Conclusioni

- Quando si utilizza HTTPS, la richiesta viene crittografata, assicurando la sicurezza del contenuto, mentre con HTTP il contenuto è aperto e potenzialmente vulnerabile a intercettazioni.
- La distinzione chiave tra HTTPS e HTTP risiede nella sicurezza.
- Gli indirizzi MAC di sorgente e destinazione rimangono costanti durante tutte le fasi.
- La differenza principale tra HTTPS e HTTP non riguarda solo la sicurezza, ma HTTPS fornisce anche:
 - Verifica dell'identità e riservatezza dei dati.

- Crittografia del traffico.
- Controllo dell'integrità del traffico.

