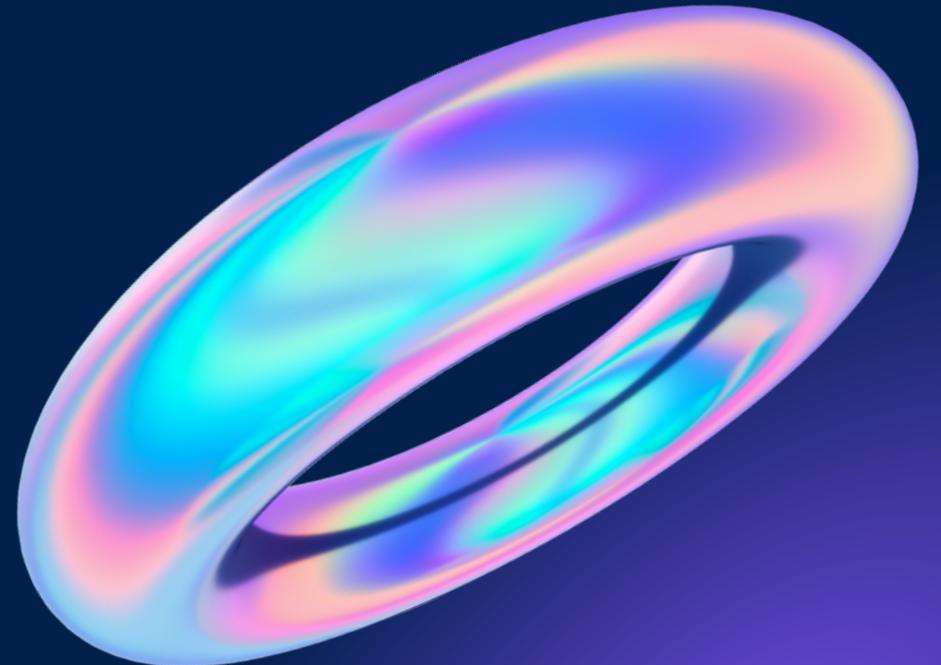




# ESERCITAZIONE

Brigaglia Emmanuela



# CONSEGNA

Simulazione di un'architettura client server  
(192.168.32.101 Win7) in laboratorio virtuale

Dove tramite web browser viene richiesta una risorsa all'hostname  
epicode.internal (192.168.32.100 Kali)

Successivamente, si vuole monitorare la comunicazione  
utilizzando Wireshark.

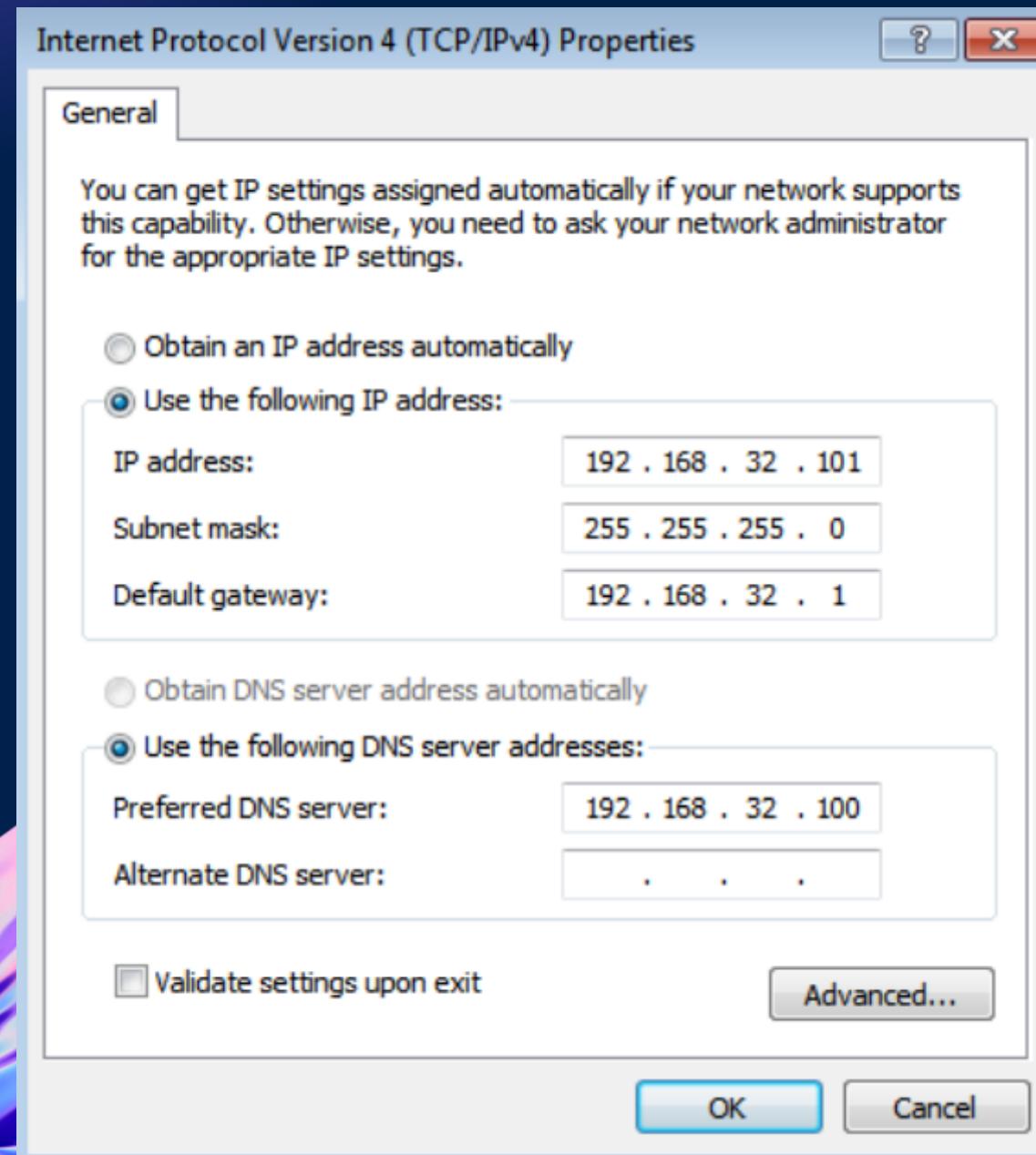
L'obiettivo è evidenziare gli indirizzi MAC di sorgente e destinazione,  
insieme al contenuto della richiesta HTTPS.

L'esercizio viene quindi ripetuto sostituendo il server HTTPS con un server  
HTTP, e si procede a intercettare nuovamente il traffico. L'intento è mettere in  
luce eventuali differenze tra il traffico catturato in HTTP e quello  
precedentemente catturato in HTTPS.

Per una comprensione più approfondita, si richiede di spiegare e motivare le  
principali differenze, se presenti.

# STEP - 1

## Impostazione indirizzi IP Windows 7 e 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 in
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
gateway 192.168.32.0
network 192.168.32.0
netmask 255.255.255.0
broadcast 192.168.32.255
gateway 192.168.32.1

^K G Help ^O Write Out ^W Where Is
^X Exit    ^R Read File ^\ Replace ^K Cut
                                         ^U Paste
```

A screenshot of a terminal window titled 'GNU nano 7.2' showing the contents of the '/etc/network/interfaces' file. The file contains the following configuration:

```
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
address 192.168.32.100
gateway 192.168.32.0
network 192.168.32.0
netmask 255.255.255.0
broadcast 192.168.32.255
gateway 192.168.32.1
```

At the bottom of the terminal window, there is a menu bar with 'File', 'Actions', 'Edit', 'View', and 'Help'. Below the menu bar, there is a toolbar with various keyboard shortcut icons.

# STEP - 2

Entriamo nel file "inetsim.conf" partendo da Kali.

Inseriamo il comando (sudo nano /etc/inetsim/inetsim.conf).

Abilitando i servizi DNS, HTTP e HTTPS, rimuovendo il cancelletto (#) davanti a ciascuna voce, come mostrato nell'esempio dello screenshot qui sotto.

```
#####
# start_service
#
# File System
# 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
# start_service smtp
# start_service smtps
# start_service pop3
# start_service pop3s
# start_service fts
```

# STEP - 3

Impostazione “service\_bind\_address  
con l'IP “0.0.0.0”

```
GNU nano 1.2                               /etc/inetsim/inetsim.conf

#####
# service_bind_address
#
# IP address to bind services to
#
# Syntax: service_bind_address <IP address>
# File System
# Default: 127.0.0.1
#
service_bind_address 0.0.0.0
```

# STEP - 4

Associamo l'hostname “epicode.internal”  
a “dns\_static” con IP di Kali

```
# 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 ftp.bar.net 10.10.20.30
dns_static epicode.internal 192.168.32.100

#####
# dns_version
#
# DNS version
#
# Syntax: dns_version <version>
#
# Default: "INetSim DNS Server"
#           the more you become, the more you are able to he
#
#dns_version "9.2.4"
```

# STEP - 5

INETSIM

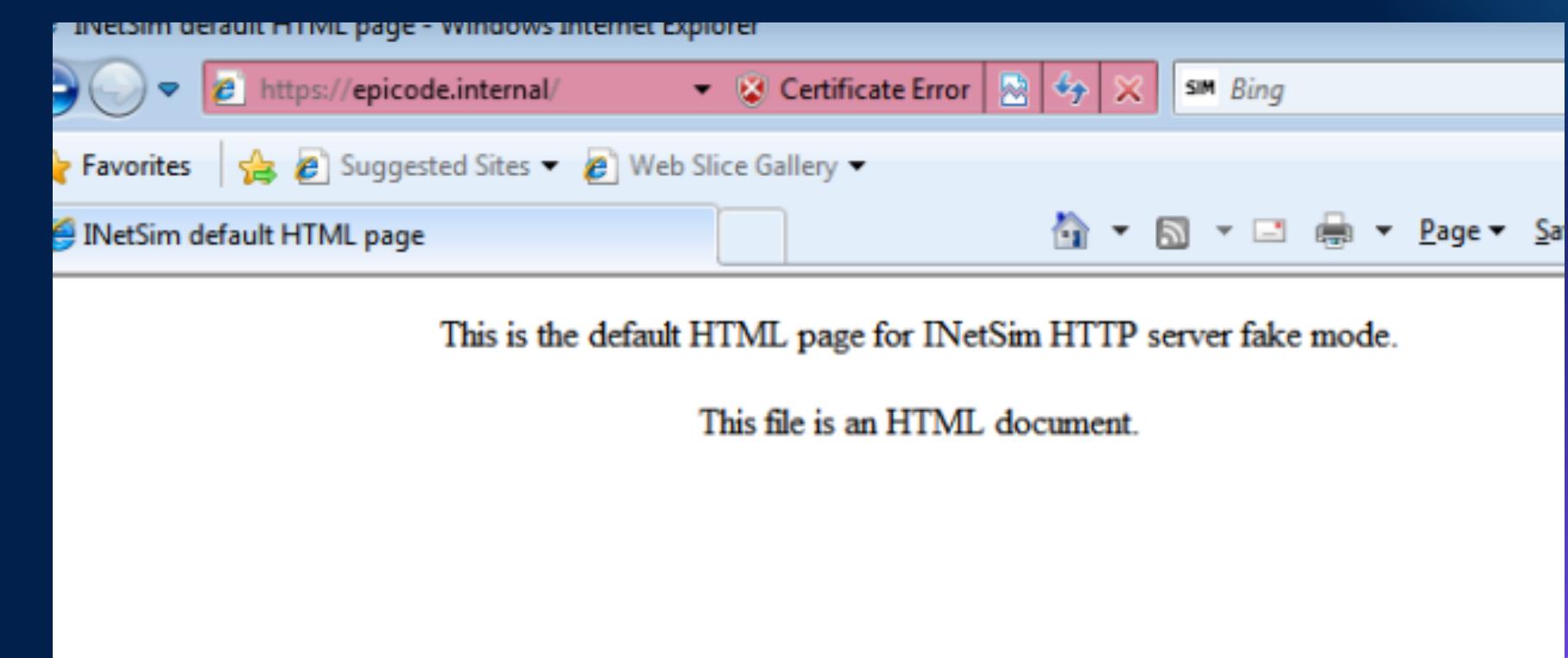
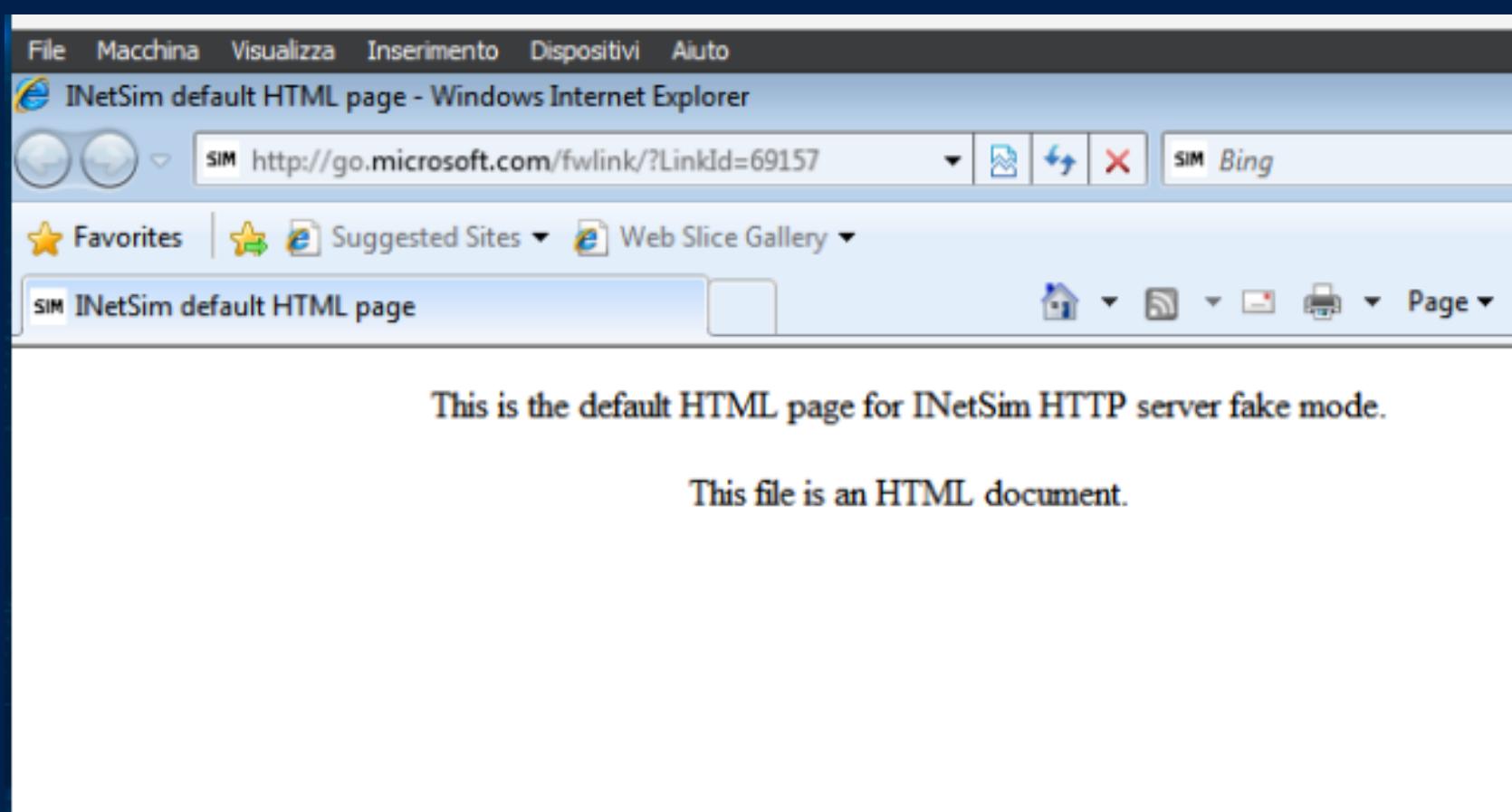
```
File Actions Edit View Help
zsh: corrupt history file /home/emma/.zsh_history
[emma@kali] ~
$ sudo nano /etc/inetsim/inetsim.conf
[sudo] password for emma :

[emma@kali] ~
$ sudo inetsim
[sudo] password for emma :
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 24135) ==
Session ID: 24135
Listening on: 0.0.0.0
Real Date/Time: 2023-11-18 20:09:54
Fake Date/Time: 2023-11-18 20:09:54 (Delta: 0 seconds)
Forking services ...
 * dns_53_tcp_udp - started (PID 24145)
print() on closed filehandle MLOG at /usr/share/perl5/Net/DNS/Nameserver.pm line
print() on closed filehandle MLOG at /usr/share/perl5/Net/DNS/Nameserver.pm line
```

# STEP - 6

Ora tramite Win7 collegiamoci a Explorer con HTTP e con HTTPS ai seguenti indirizzi:

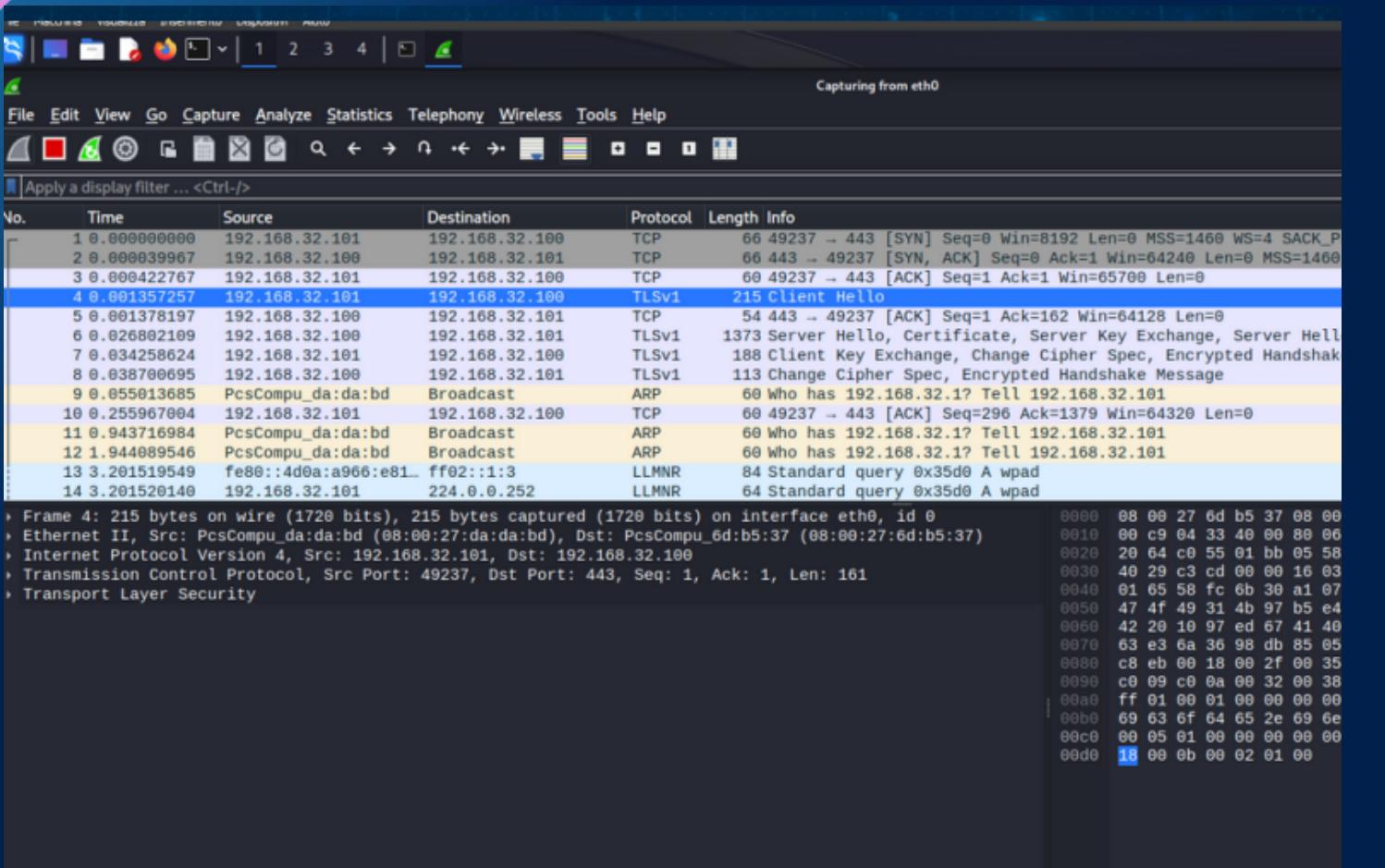
- <https://epicode.internal/>
- <http://epicode.internal/>



# STEP - 7

## WIRESHARK - HTTPS

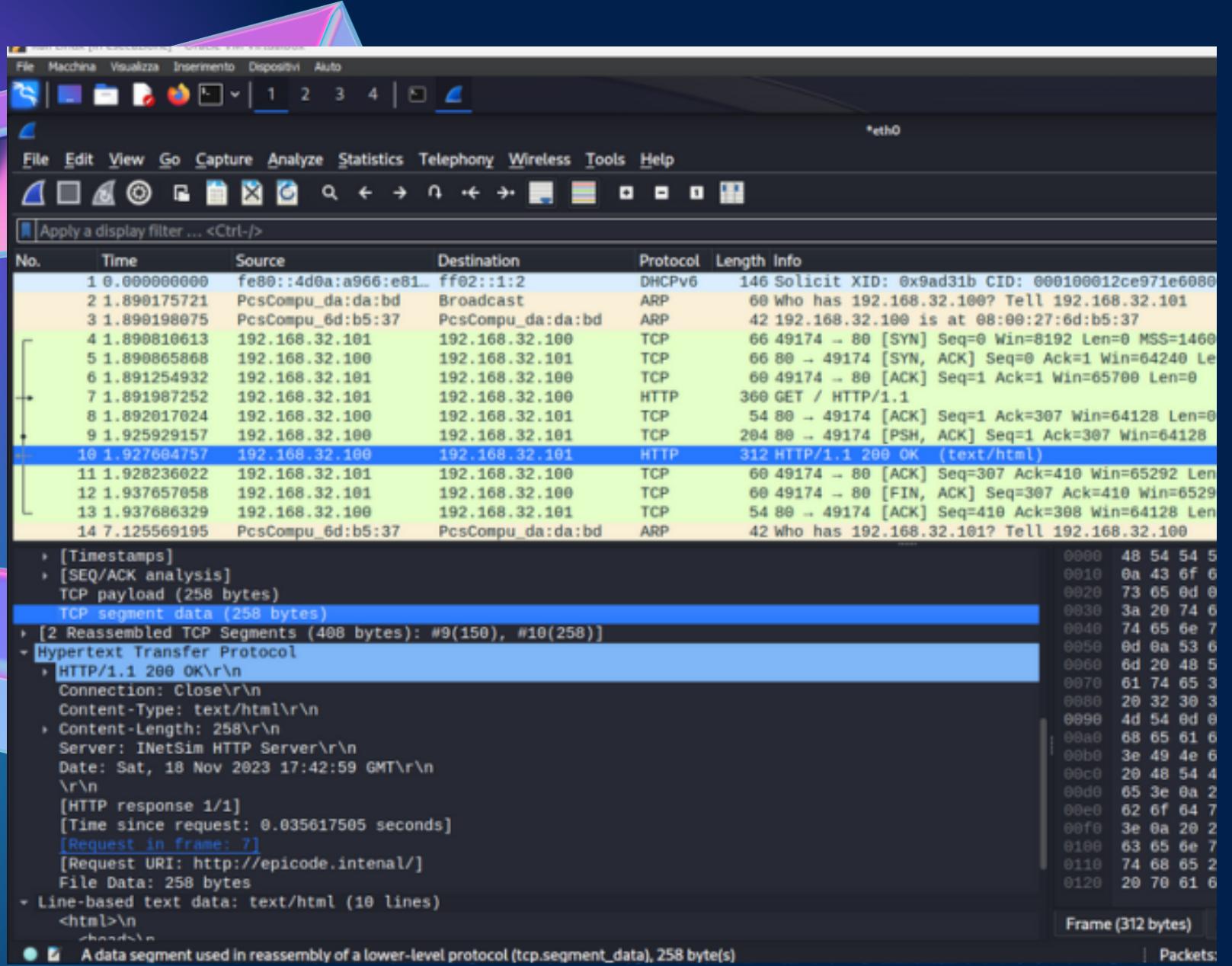
Tramite Wireshark effettuiamo il packet capture con rete eth0  
Per verificare che Win7 e Kali trasmettano i pacchetti correttamente tra loro



Nel protocollo HTTPS, le informazioni sono crittografate. Questo è evidente durante la "three-way handshake", un processo in cui avviene uno scambio di pacchetti TLS. Questa procedura è fondamentale per garantire la sicurezza dei dati sensibili.

# STEP - 8

## WIRESHARK - HTTP



Abbiamo un'osservazione di uno scambio di pacchetti ARP. Nel primo pacchetto, il computer Windows invia una richiesta broadcast a tutti i dispositivi sulla rete. Nel secondo pacchetto, Kali risponde confermando di avere l'indirizzo IP richiesto (192.168.32.101). Successivamente, si verifica la "three-way handshake". Poco dopo, c'è una richiesta GET da parte di Windows 7 a un server HTTP, con una risposta di codice 200 che conferma la trasmissione riuscita al computer Windows.

FINE