

PROGETTO (PRATICA W4D4)

Nell'esercizio di oggi metteremo insieme le competenze acquisite finora.
Lo studente verrà valutato sulla base della risoluzione al problema seguente.

Requisiti e servizi:

- Kali Linux ☐ IP 192.168.32.100
- Windows 7 ☐ IP 192.168.32.101
- HTTPS server: attivo
- Servizio DNS per risoluzione nomi di dominio: attivo

Traccia:

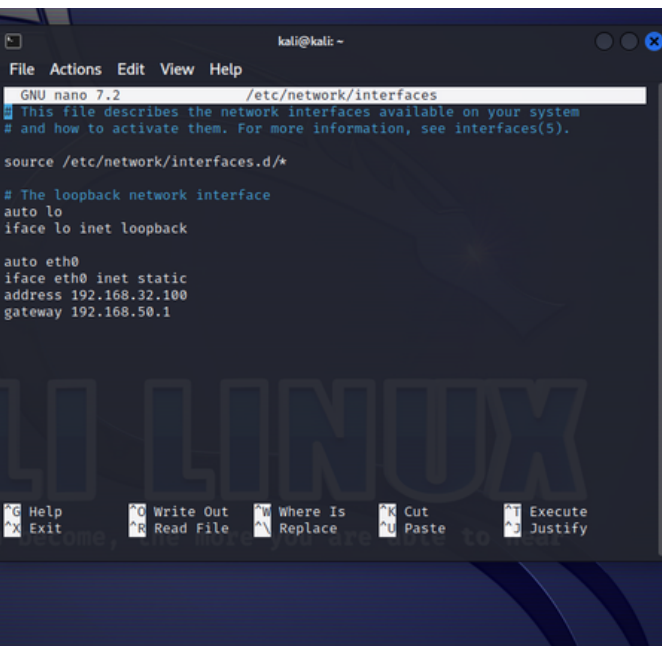
Simulare, in ambiente di laboratorio virtuale, un'architettura client server in cui un client con indirizzo 192.168.32.101 (Windows 7) richiede tramite web browser una risorsa all'hostname `epicode.internal` che risponde all'indirizzo 192.168.32.100 (Kali).

Si intercetti poi la comunicazione con Wireshark, evidenziando i MAC address di sorgente e destinazione ed il contenuto della richiesta HTTPS.

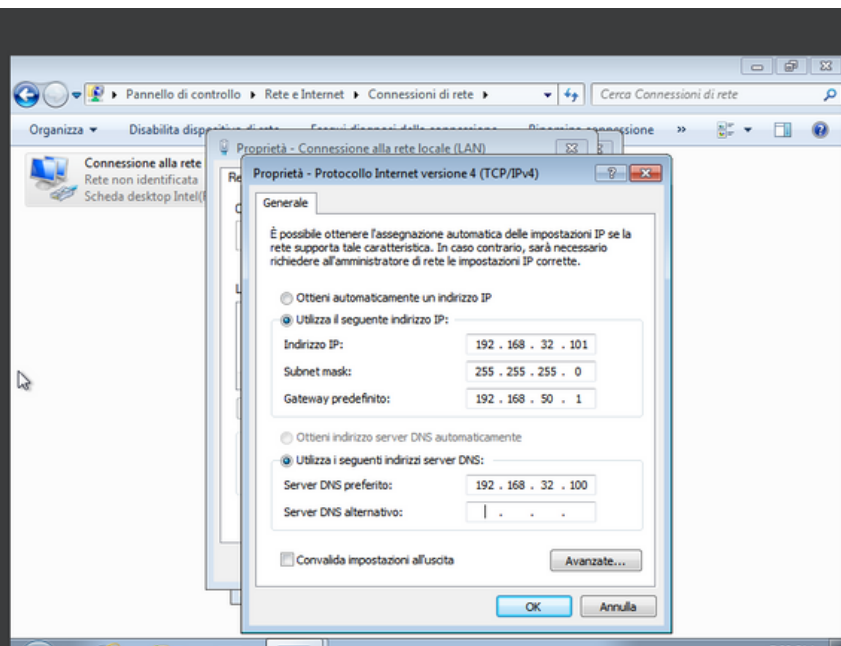
Ripetere l'esercizio, sostituendo il server HTTPS, con un server HTTP. Si intercetti nuovamente il traffico, evidenziando le eventuali differenze tra il traffico appena catturato in HTTP ed il traffico precedente in HTTPS. Spiegare, motivandole, le principali differenze se presenti.

2

1) Ho configurato gli indirizzi IP statici della consegna dell'esercizio, e nella macchina virtuale di WINDOWS 7 ho cambiato anche il puntamento del DNS, (con l'indirizzo IP di KALI).



```
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  
  
# The ethernet network interface  
auto eth0  
iface eth0 inet static  
address 192.168.32.100  
gateway 192.168.50.1
```



2) ho configurato su InetSim i servizi DNS e HTTPS.



```
File Actions Edit View Help
GNU nano 7.2 /etc/inetsim/inetsim.conf
#start_service dummy_udp

#####
# service_bind_address
#
# IP address to bind services to
#
# Syntax: service_bind_address <IP address>
#
# Default: 127.0.0.1
#
service_bind_address 192.168.32.100

#####
# service_run_as_user
#
# User to run services
#
# Syntax: service_run_as_user <username>
#
# Default: inetsim
#
# quieter you become, the more you are able to hear

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute
^X Exit      ^R Read File  ^N Replace    ^U Paste      ^J Justify

File Actions Edit View Help
GNU nano 7.2 /etc/inetsim/inetsim.conf
# 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 ftp
#start_service ftps
#start_service tftp
#start_service irc
#start_service ntp
#start_service finger
#start_service ident
#start_service syslog

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute
^X Exit      ^R Read File  ^N Replace    ^U Paste      ^J Justify

File Actions Edit View Help
GNU nano 7.2 /etc/inetsim/inetsim.conf
# 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 epicode.internal 192.168.32.100
#dns_static ns1.foo.com 10.70.50.30
#dns_static ftp.bar.net 10.10.20.30

#####
# quieter you become, the more you are able to hear

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute
^X Exit      ^R Read File  ^N Replace    ^U Paste      ^J Justify

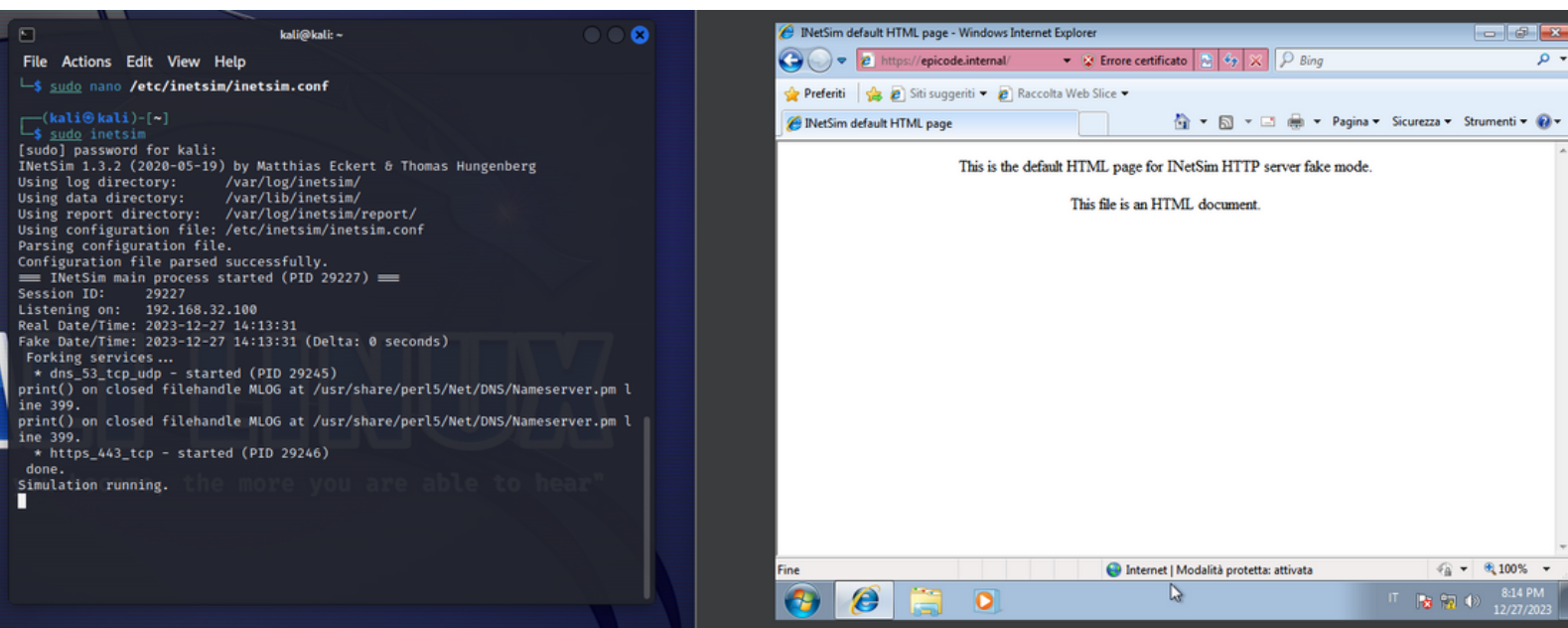
File Actions Edit View Help
GNU nano 7.2 /etc/inetsim/inetsim.conf
# Port number to bind DNS service to
#
# Syntax: dns_bind_port <port number>
#
# Default: 53
#
#dns_bind_port 53

#####
# 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
# quieter you become, the more you are able to hear

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute
^X Exit      ^R Read File  ^N Replace    ^U Paste      ^J Justify
```

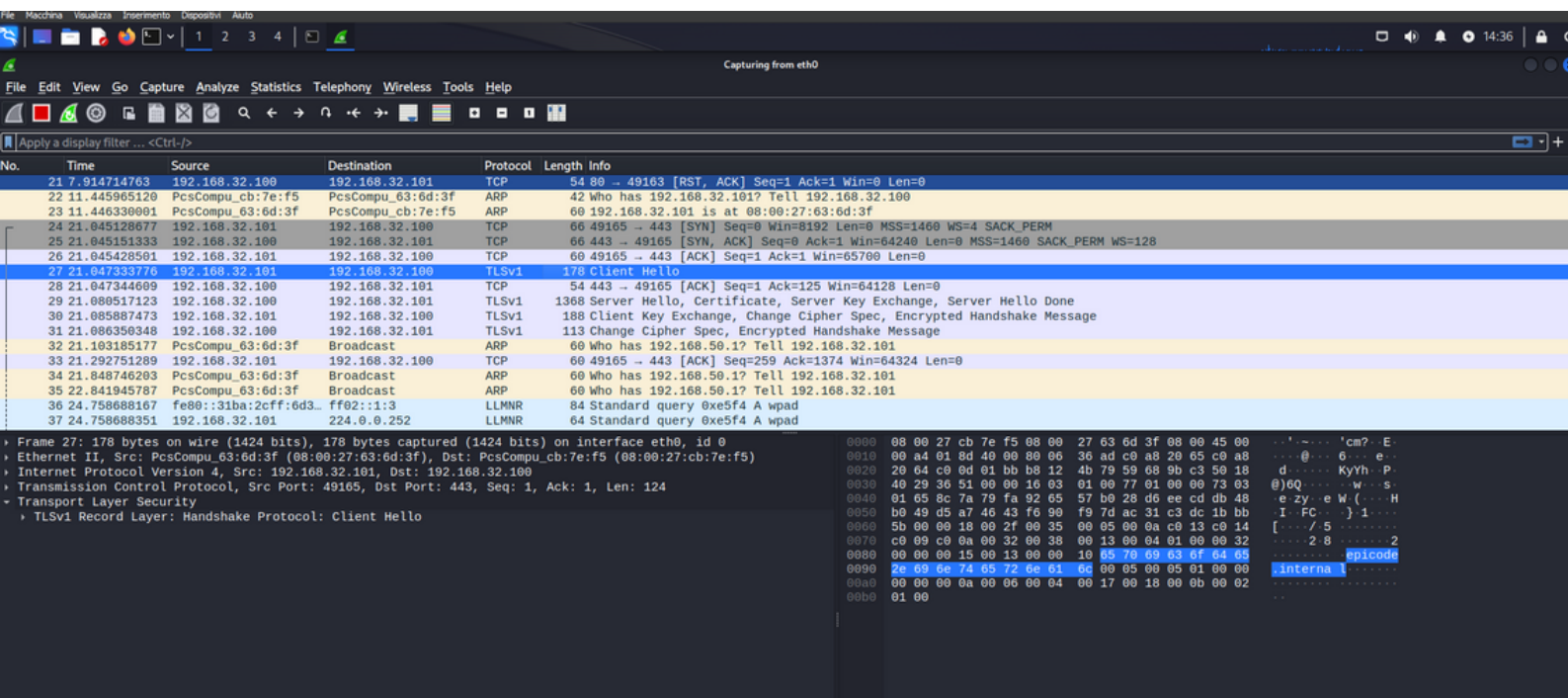
3) Ho avviato la simulazione con InetSim e ho verificato la connessione con WINDOWS 7 aprendo la pagina <https://epicode.internal/>.



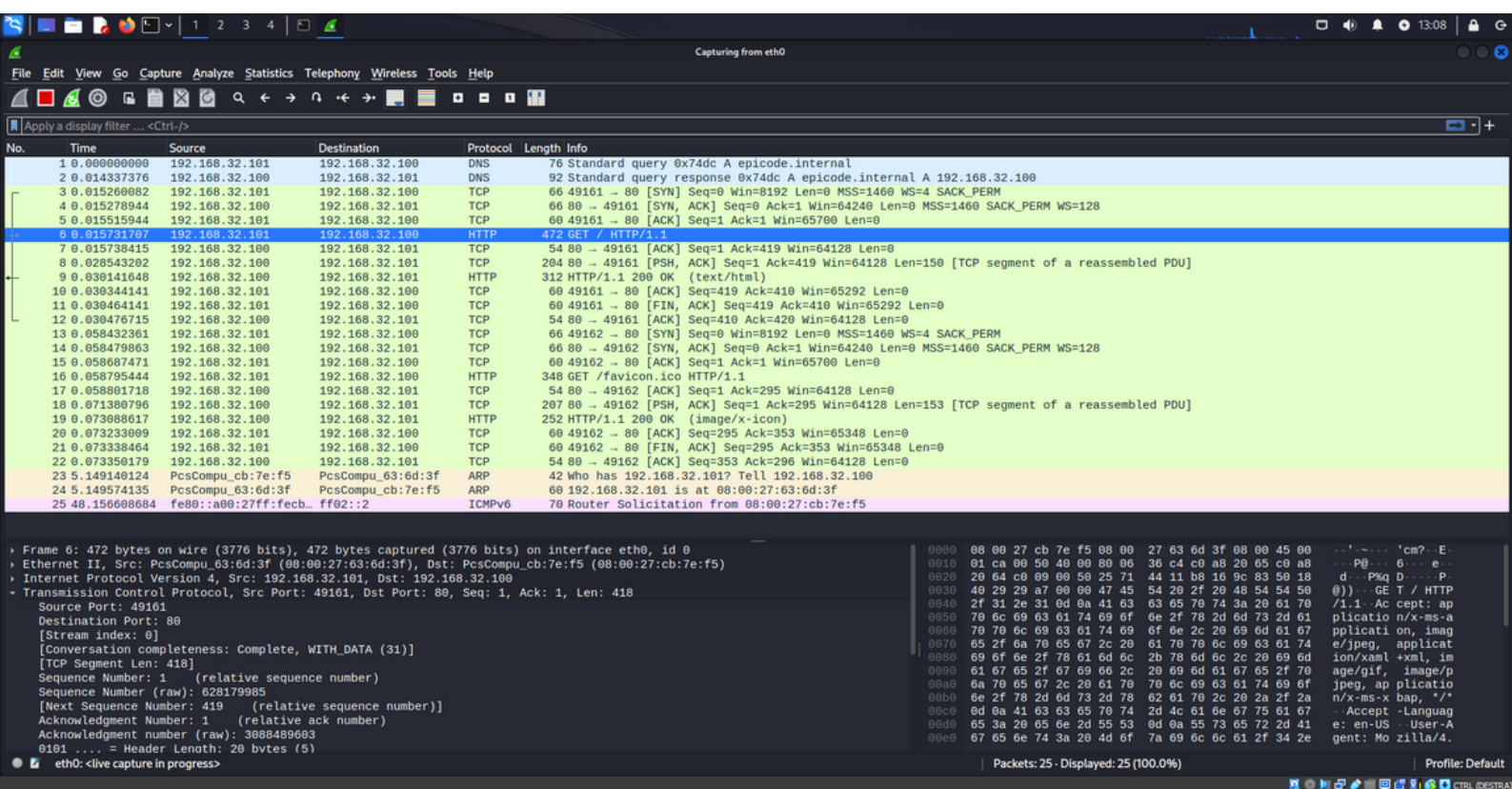
```
File Actions Edit View Help
$ sudo nano /etc/inetsim/inetsim.conf
(kali@kali)-[~]
$ sudo inetsim
[sudo] password for kali:
InetSim 1.3.2 (2020-05-19) by Matthias Eckert & Thomas Hungenberg
Using log directory: /var/log/inetsim/
Using data directory: /var/lib/inetsim/
Using report directory: /var/log/inetsim/report/
Using configuration file: /etc/inetsim/inetsim.conf
Parsing configuration file.
Configuration file parsed successfully.
== InetSim main process started (PID 29227) ==
Session ID: 29227
Listening on: 192.168.32.100
Real Date/Time: 2023-12-27 14:13:31
Fake Date/Time: 2023-12-27 14:13:31 (Delta: 0 seconds)
Forking services...
* dns_53_tcp_udp - started (PID 29245)
print() on closed filehandle MLOG at /usr/share/perl5/Net/DNS/Nameserver.pm l
ine 399.
print() on closed filehandle MLOG at /usr/share/perl5/Net/DNS/Nameserver.pm l
ine 399.
* https_443_tcp - started (PID 29246)
done.
Simulation running. the more you are able to hear

File Actions Edit View Help
InetSim default HTML page - Windows Internet Explorer
https://epicode.internal/ Errore certificato Bing
Preferiti Siti suggeriti Raccolta Web Svice
InetSim default HTML page
Pagina Sicurezza Strumenti
This is the default HTML page for InetSim HTTP server fake mode.
This file is an HTML document.
Fine
Internet | Modalità protetta: attivata
8:14 PM 12/27/2023
```

4) Cattura dei pacchetti con Wireshark



5) Dopo aver cambiato le impostazioni su InetSim, disattivando i servizi HTTPS e abilitando i servizi HTTP, ho fatto lo stesso procedimento e ho catturato i pacchetti con Wireshark.



CONCLUSIONI) Come si può notare dalle immagini sopra ci sono delle differenze tra le due catture di Wireshark, in HTTPS c'è il protocollo di sicurezza TLS che fornisce un meccanismo di crittografia nascondendo il contenuto del pacchetto. Mentre in HTTP c'è il contenuto del pacchetto mostrato in chiaro.