

## CONSEGNA CORSO CYBER SECURITY W3D4

STUDENTE: ORAZIO MORGILLO

L'obiettivo dell'esercizio è quello di simulare con le macchine virtuali un'architettura in cui un client server (windows) richiede tramite browser una risorsa all'hostname (kali) `epicode.internal`. successivamente bisogna intercettare le comunicazioni e il contenuto delle richieste https e http in entrata ed uscita su whireshark

Premessa: come visto a lezione di pratica e come riscontrato da altri colleghi, ho problemi con il dns server e quindi non riesco a visualizzare la pagina `epicode.internal`.

Da browser su windows sono riuscito a richiedere le pagine <https://192.168.50.100> e <http://192.168.50.100>

Requisiti:

Kali: 192.168.50.100

Windows: 192.168.50.102

HTTPS server: attivo

HTTP server: attivo

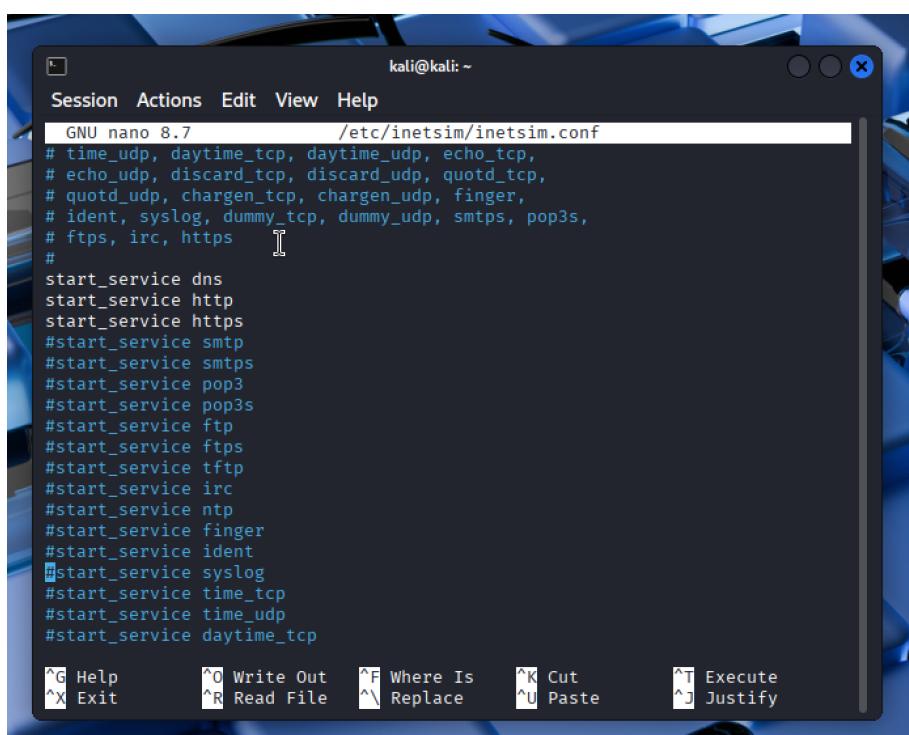
DNS server: attivo

Svolgimento:

per prima cosa, su Kali ho controllato che le due macchine fossero in comunicazione tramite il comando ping 192.168.50.102.

successivamente, ho attivato i servizi https, http e dns dal terminale con il comando sudo nano `/etc/inetsim/inetsim.conf` e configurato l'indirizzo ip 192.168.50.100 su `service_bind_address` e `dns_default_ip`.

Inetsim è un simulatore di servizi internet preconfigurato su Kali linux.

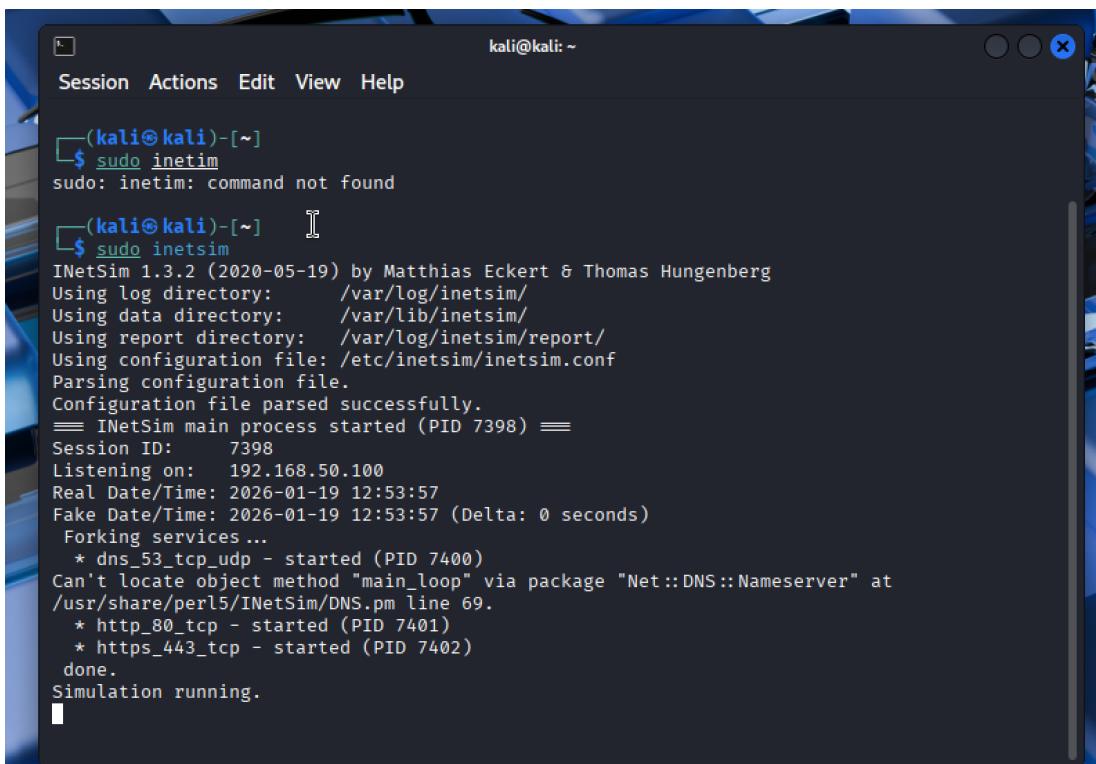


The screenshot shows a terminal window titled "kali@kali: ~" with the command "GNU nano 8.7 /etc/inetsim/inetsim.conf" running. The content of the file is as follows:

```
# time_udp, daytime_tcp, daytime_udp, echo_tcp,
# echo_udp, discard_tcp, discard_udp, quod_tcp,
# quod_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
#start_service time_tcp
#start_service time_udp
#start_service daytime_tcp
```

At the bottom of the terminal, there is a menu bar with "Session", "Actions", "Edit", "View", and "Help". Below the menu, there are several keyboard shortcuts: "G Help", "X Exit", "O Write Out", "R Read File", "F Where Is", "Replace", "K Cut", "U Paste", "T Execute", and "J Justify".

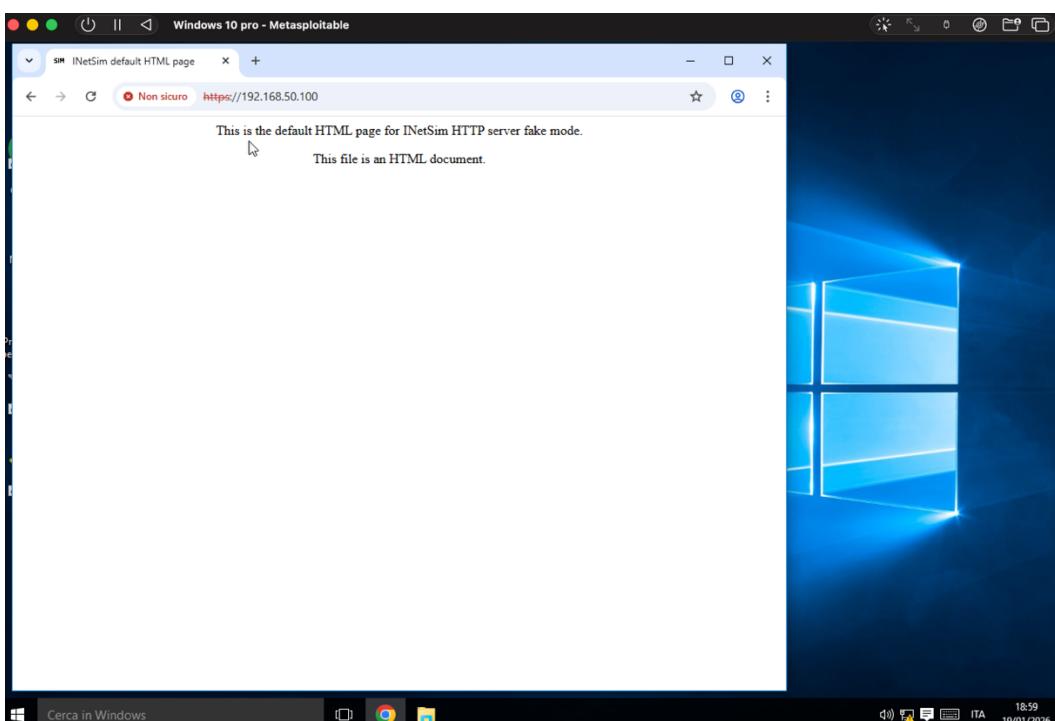
Ho fatto partire poi il comando sudo inetsim che ci da come output la schermata di seguito e che vuol dire che il servizio https è in ascolto sulla porta 443 e il servizio http sulla porta 80.



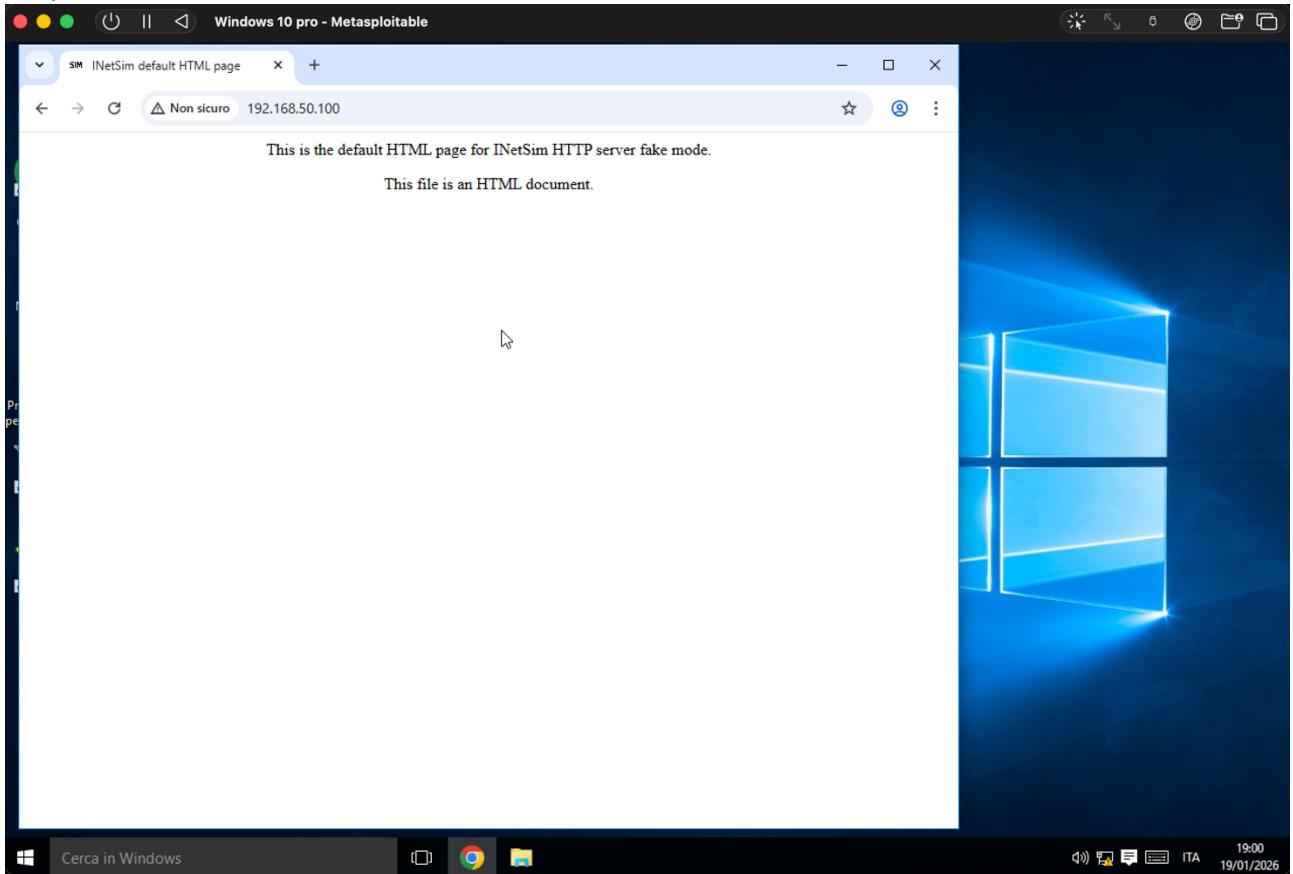
```
kali@kali: ~
Session Actions Edit View Help
(kali㉿kali)-[~]
$ sudo inetsim
sudo: inetsim: command not found
(kali㉿kali)-[~]
$ sudo inetsim
INetSim 1.3.2 (2020-05-19) by Matthias Eckert & Thomas Hungenberg
Using log directory: /var/log/inetsim/
Using data directory: /var/lib/inetsim/
Using report directory: /var/log/inetsim/report/
Using configuration file: /etc/inetsim/inetsim.conf
Parsing configuration file.
Configuration file parsed successfully.
== INetSim main process started (PID 7398) ==
Session ID: 7398
Listening on: 192.168.50.100
Real Date/Time: 2026-01-19 12:53:57
Fake Date/Time: 2026-01-19 12:53:57 (Delta: 0 seconds)
Forking services ...
* dns_53_tcp_udp - started (PID 7400)
Can't locate object method "main_loop" via package "Net::DNS::Nameserver" at
/usr/share/perl5/INetSim/DNS.pm line 69.
* http_80_tcp - started (PID 7401)
* https_443_tcp - started (PID 7402)
done.
Simulation running.
```

Tenendo aperto sudo inetsim su Kali, ci spostiamo su Windows e ci connettiamo via browser agli indirizzi <https://192.168.50.100> e <http://192.168.50.100> che restituiscono le schermate di seguito che ci indicano che il servizio è attivo.

https:

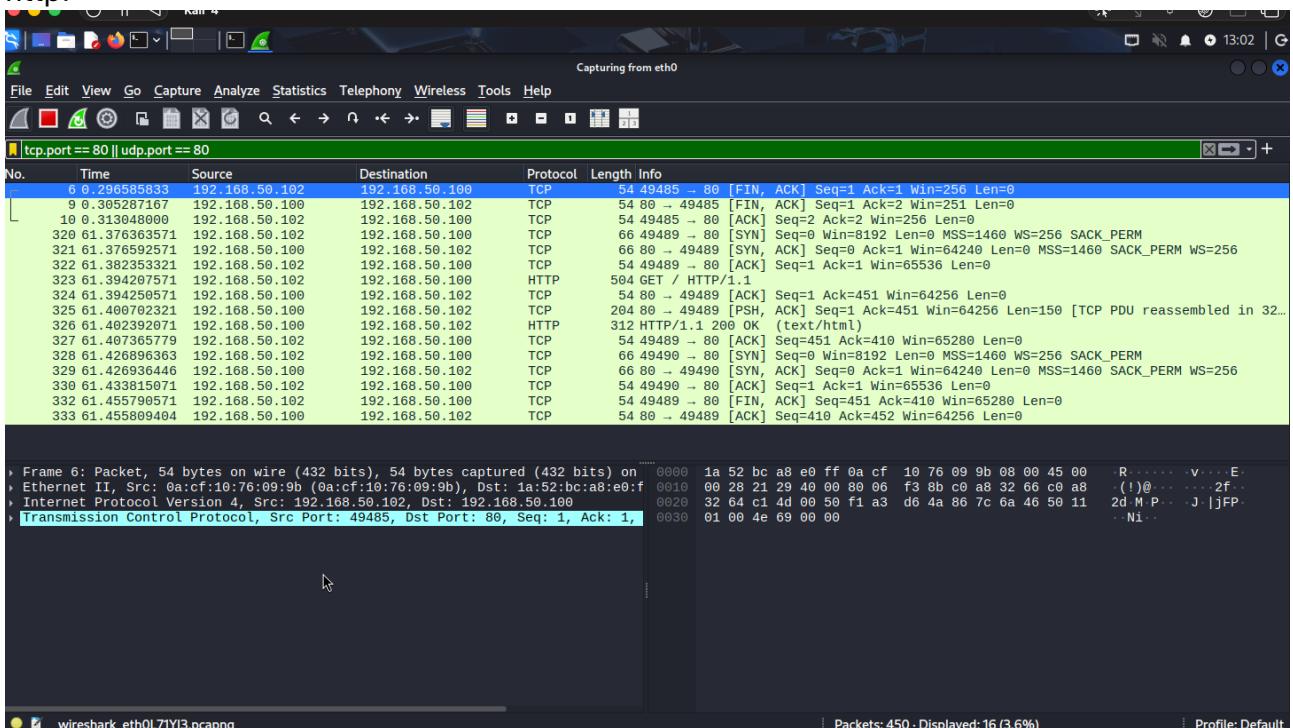


http:

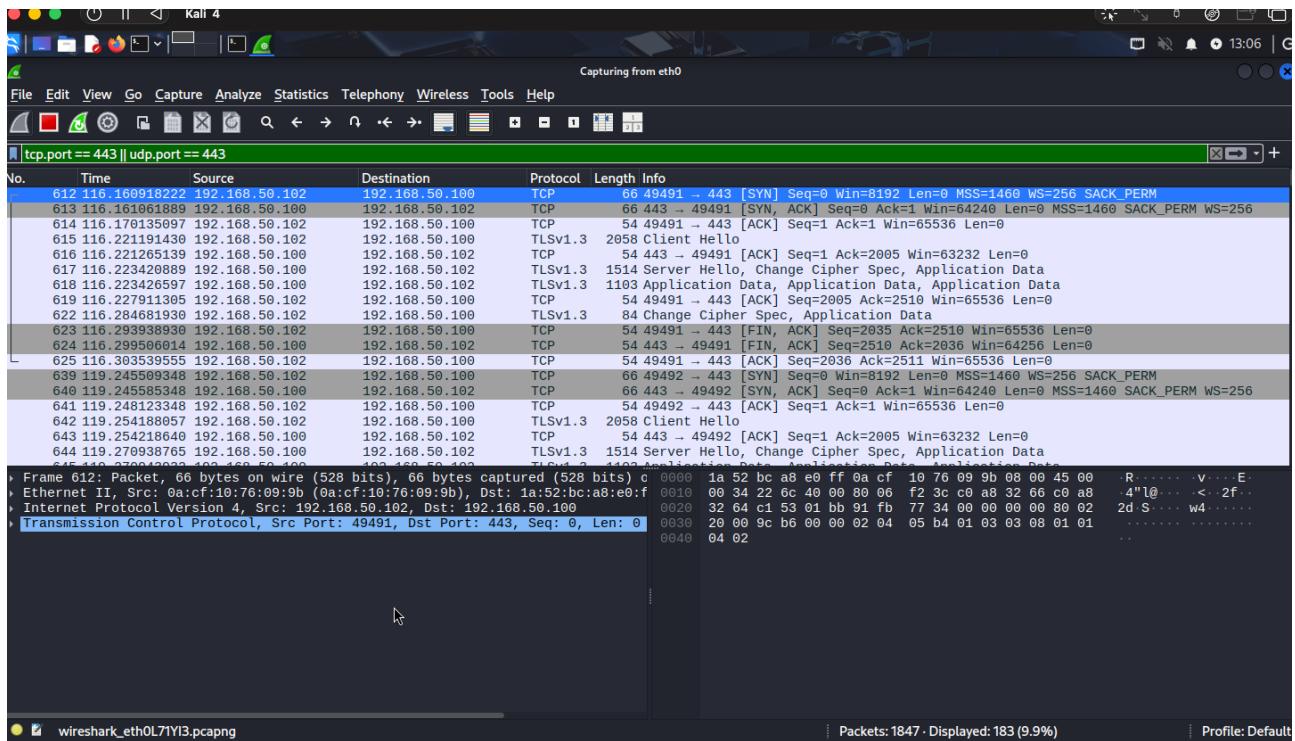


Apriamo ora whireshark e intercettiamo le comunicazioni in entrata ed uscita.

http:



https:



Whireshark mostra il traffico HTTP sulla porta 80 mostrando intestazioni, URL e payload non crittografati, filtrabili con "http" o "tcp.port == 80".

HTTPS sulla porta 443 appare come pacchetti TLS crittografati, con soli dettagli come indirizzi IP, porte e fasi di handshake senza dati leggibili.