CONFIGURAZIONE DI UNA POLICY SUL FIREWALL WINDOWS + PACKET CAPTURE CON WIRESHARK

1) CONFIGURARE POLICY PER PERMETTERE IL PING DA MACCHINE LINUX A MACCHINA WINDOWS 7 NEL NOSTRO LABORATORIO VM (WINDOWS FIREWALL)

#KALI LINUX IP: 192.168.50.100
#METASPLOITABLE IP: 192.168.50.101
#WINDOWS 7 IP: 192.168.50.102

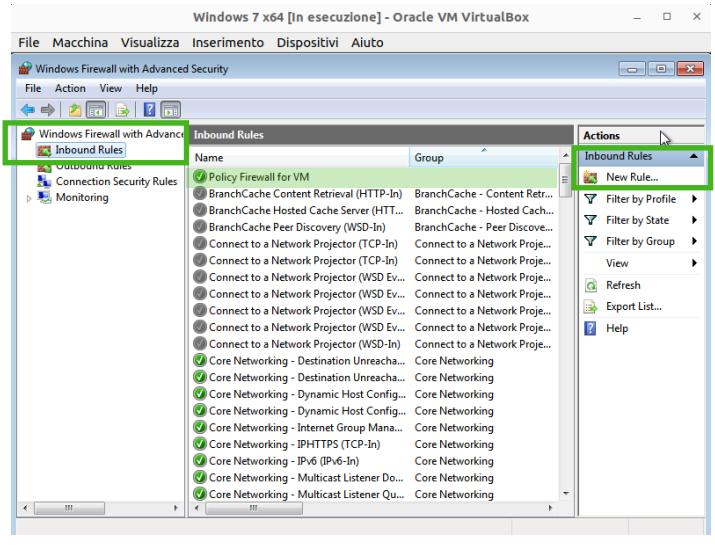
2) UTILIZZO DELL'UTILITY InetSim PER L'EMULAZIONE DI SERVIZI INTERNET

3) CATTURA DI PACCHETTI CON WIRESHARK (DIFFERENZA TRA HTTP e HTTPS)

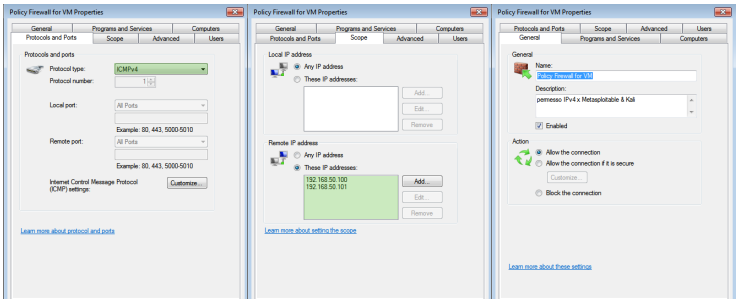
#FIREWALL POLICY

SI DEVONO MODIFICARE LE IMPOSTAZIONI AVANZATE DI WINDOWS FIREWALL PER PERMETTERE IL TRAFFICO DA UN DEVICE DELLA STESSA RETE, ALTRIMENTI BLOCCA IL PING

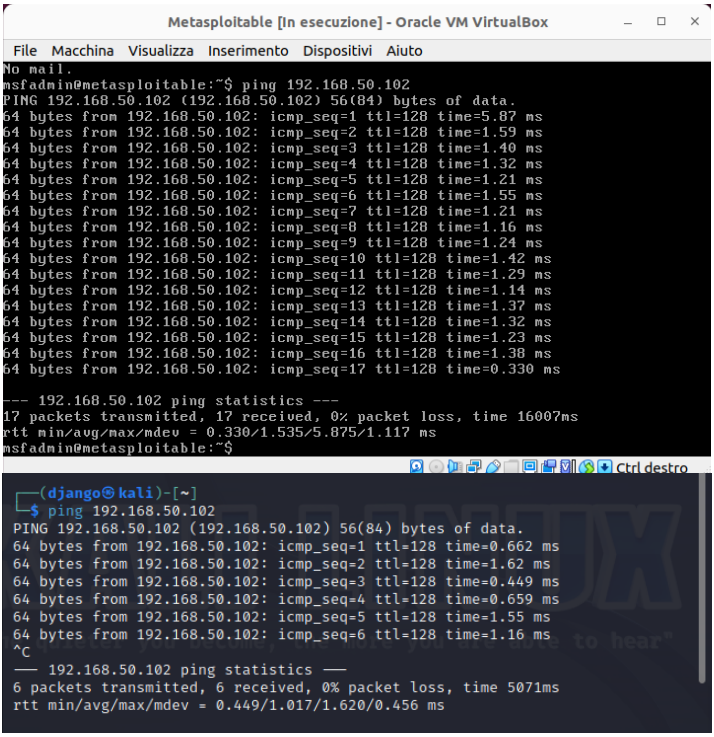
IMPOSTAZIONI AVANZATE -> INBOUND RULES -> NEW



CAMPI NECESSARI:
#PROTOCOLS AND PORTS SETTANDO IL CAMPO "PROTOCOL TYPE" SU ICMPv4
#SCOPE SETTANDO GLI IP A CUI CONCEDIAMO DI INVIARE PING
#GENERAL (FACOLTATIVO) INSERENDO UN RIFERIMENTO AL PROTOCOLLO PER DISTINGUERLO DA ALTRI



PING VS WINDOWS 7 RICEVUTO DA #METASPLOITABLE E #KALI

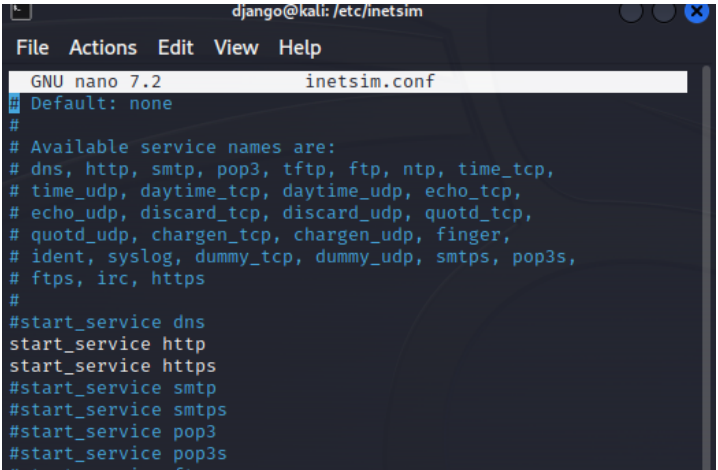


#INETSIM

InetSim È UN SOFTWARE CHE SIMULA LE FEATURES DELLA RETE/SERVIZI INTERNET, DA CONFIGURARE ALL'INTERNO DELLA VM DI KALI LINUX (CHE RICORDIAMO ESSERE IL NOSTRO OS ATTACCANTE). ALL'INTERNO DI "NANO" (EDITOR DI TESTO CHE CONSENTE DI MODIFICARE FILE) COMMENTEREMO CON "#" DAVANTI I SERVIZI CHE NON UTILIZZEREMO LASCIANDO INTATTI HTTP/HTTPS. FINGEREMO SUCESSIVAMENTE DI ACCEDERE AD UN SITO FAKE SU LOCAL HOST (IP - 127.0.0.1 - detto LOOPBACK) SIMULANDO UNA CONNESSIONE INTERNET PROTETTA E NON PROTETTA.

PASSAGGI DI CONFIGURAZIONE DI INETSIM:

- cd /etc/inetsim (dir)
- ls (file presenti nella dir)
- sudo nano inetsim.conf
- # per commentare i servizi da "spegnere" (tranne http/https)
- nella riga "service_bind_address" inseriamo il local host e sovrascriviamo con ^O
- sudo inetsim (per avviare la simulazione)



IP DI ACCESSO AL SITO INTERNET SIMULATO:

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

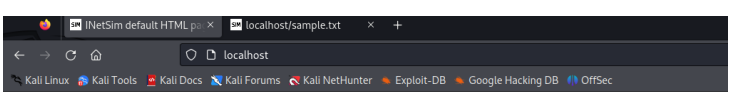
FILE FAKE ACCESSIBILI NELLA SIMULAZIONE:

```
# Syntax: https_fakefile <extension> <filename> <mime-type>
#
# Default: none
#
https_fakefile txt  sample.txt     text/plain
https_fakefile htm  sample.html    text/html
https_fakefile html sample.html    text/html
https_fakefile php  sample.html    text/html
https_fakefile gif  sample.gif     image/gif
https_fakefile jpg  sample.jpg     image/jpeg
https_fakefile jpeg sample.jpg     image/jpeg
https_fakefile png  sample.png     image/png
https_fakefile bmp  sample.bmp     image/x-ms-bmp
https_fakefile ico  favicon.ico    image/x-icon
https_fakefile exe  sample_gui.exe x-msdos-program
https_fakefile com  sample_gui.exe x-msdos-program
```

SIMULAZIONE AVVIATA CORRETTAMENTE:

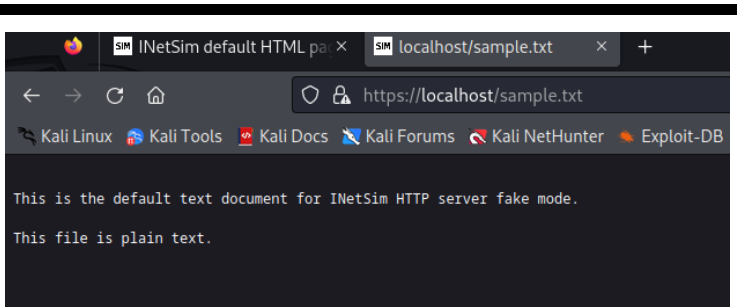
```
┌──(django㉿kali)-[~]
└─$ cd /etc/inetsim

┌──(django㉿kali)-[/etc/inetsim]
└─$ sudo nano inetsim.conf

┌──(django㉿kali)-[/etc/inetsim]
└─$ sudo inetsim
INetSim 1.3.2 (2020-05-19) by Matthias Eckert & Thomas Hungenb
erg
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 5337) ===
Session ID:     5337
Listening on:   127.0.0.1
Real Date/Time: 2023-11-13 10:00:43
Fake Date/Time: 2023-11-13 10:00:43 (Delta: 0 seconds)
 Forking services...
  * http_80_tcp - started (PID 5347)
  * https_443_tcp - started (PID 5348)
 done.
Simulation running.
```

FAKE SITO INTERNET DA BROWSER + FAKE FILE .TXT DA WEB:



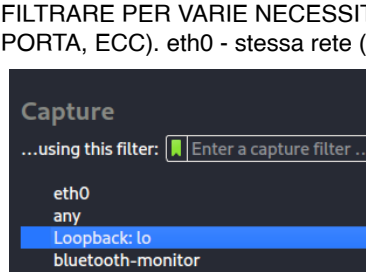
This is the default HTML page for INetSim HTTP server fake mode.

This file is an HTML document.



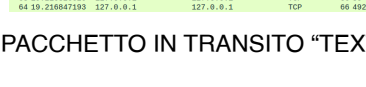
#WIRESHARK

SI UTILIZZA PER SNIFFARE AKA CATTURARE ED ANALIZZARE I PACCHETTI IN TRANSITO SU UNA DETERMINATA RETE. ATTRAVERSO LA SELEZIONE DI UNA SCHEDA DI RETE NE VIENE CATTURATO IL TRAFFICO CHE SI PUÒ SUCESSIVAMENTE FILTRARE PER VARIE NECESSITÀ/INTERESSE (IP / DNS / PORTA, ECC). eth0 - stessa rete (LE NOSTRA VM) / Loopback



HTTP://LOCALHOST: PORTA 80 - CONNESSIONE NON PROTETTA

RICHIESTA IN LOOPBACK SULLA PORTA 80 (HTTP) DA “GET” A “OK” DEL SERVER CON SEQUENZA DI THREE-WAY-HANDSHAKE. STRINGHE DA SYN/SYN ACK fino a FIN/ACK ACK.



PACCHETTO IN TRANSITO “TEXT/PLAIN”.

HTTPS://LOCALHOST: PORTA 443 - CONNESSIONE PROTETTA

RICHIESTA IN LOOPBACK SULLA PORTA 443 (HTTPS) DA “Client Hello” A “Server Hello” CON SEQUENZA DI THREE-WAY-HANDSHAKE CON CIFRATURA PIÙ COMPLESSA.