

ESERCITAZIONE 8

FIREWALL-INETSIM-

WIRESHARK



Andrea Ferrantino

L'esercizio consiste nella configurazione del firewall di WIN 7 e in una packet capture con Wireshark attraverso l'utilizzo di InetSim su Kali Linux.

Configurazione win 7 firewall

Per permettere il ping tra le due macchine è necessario stabilire una policy su Win 7, per prima cosa andiamo nel pannello di controllo ed eseguiamo questi passaggi:

Control Panel System and Security Firewall Advanced Settings Inbound Rules New Rule Custom Rule

**Fatto ciò impostiamo la seguente configurazione:*

epicode Properties



Protocols and Ports

Programs and Services

Computers

Scope

Advanced

Users

Protocols and ports



Protocol type:

ICMPv4

Protocol number:

1

Local port:

All Ports

Example: 80, 443, 5000-5010

Remote port:

All Ports

Example: 80, 443, 5000-5010

Internet Control Message Protocol
(ICMP) settings:

Customize...

[Learn more about protocol and ports](#)

OK

Cancel

Apply



epicode Properties



General	Programs and Services	Computers
Protocols and Ports	Scope	Advanced
		Users

Protocols and ports



Protocol type:

ICMPv4

Protocol number:

1

Local port:

All Ports

Example: 80, 443, 5000-5010

Remote port:

All Ports

Example: 80, 443, 5000-5010

Internet Control Message Protocol
(ICMP) settings:

Customize...

[Learn more about protocol and ports](#)

OK

Cancel

Apply



Settaggio Inetsim

Apriamo il terminale di Kali Linux e seguiamo questi passaggi per aprire il file inetsim.conf:

```
andrea@kali: /etc/inetsim
File Actions Edit View Help
zsh: corrupt history file /home/andrea/.zsh_history
└─(andrea㉿kali)-[~]
$ cd /etc/inetsim

└─(andrea㉿kali)-[/etc/inetsim]
$ ls
inetsim.conf

└─(andrea㉿kali)-[/etc/inetsim]
$ sudo nano inetsim.conf
[sudo] password for andrea: ■
```

Home



Inseriamo la password ci verrà data la possibilità il file "inetsim.conf" settandolo lascialndo attivi "http e https" inserendo il carattere "#"

The screenshot shows a Kali Linux desktop environment with a terminal window open. The terminal window title is "Kali Linux [In esecuzione] - Oracle VM VirtualBox". The window contains the file "/etc/inetsim.conf" edited with the nano text editor. The file content includes comments about service starting and a list of available service names like dns, http, smtp, etc. At the bottom of the file, there is a line starting with "#". The terminal window has a standard Linux-style menu bar (File, Actions, Edit, View, Help) and a toolbar with icons for file operations. A status bar at the bottom shows keyboard shortcuts for various functions.

```
GNU nano 7.2                                inetsim.conf

#####
# start_service
#
# 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, smtsp, pop3s,
# ftsp, irc, https
#
# start_service dns
start_service http
start_service https
# start_service smtp
# start_service smtsp
# start_service pop3
# start_service pop3s
# start_service ftp
# start_service ftsp
# start_service tftp
# start_service irc
# start_service ntp
# start_service finger
# start_service ident
# start_service syslog
# start_service time_tcp
```

^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location M-U Undo
^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^/ Go To Line M-E Redo

Andremo a impostare l'indirizzo 127.0.0.1 come IP address per il fake service che andremo ad utilizzare:

Kali Linux [In esecuzione] - Oracle VM VirtualBox

File Macchina Visualizza Inserimento Dispositivi Aiuto

andrea@kali: /etc/inetsim

File Actions Edit View Help

GNU nano 7.2 inetsim.conf

```
# 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

Home #####
# 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

#####
# service_run_as_user
#
# User to run services
#
# Syntax: service_run_as_user <username>
#
# Default: inetsim

^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute
^X Exit ^R Read File ^V Replace ^U Paste ^J Justify ^C Local
^/ Go To
```

Salviamo il file con “Ctrl+X” , “Y” e digitiamo il comando
“sudo inetsim”

Kali Linux [In esecuzione] - Oracle VM VirtualBox

File Macchina Visualizza Inserimento Dispositivi Aiuto

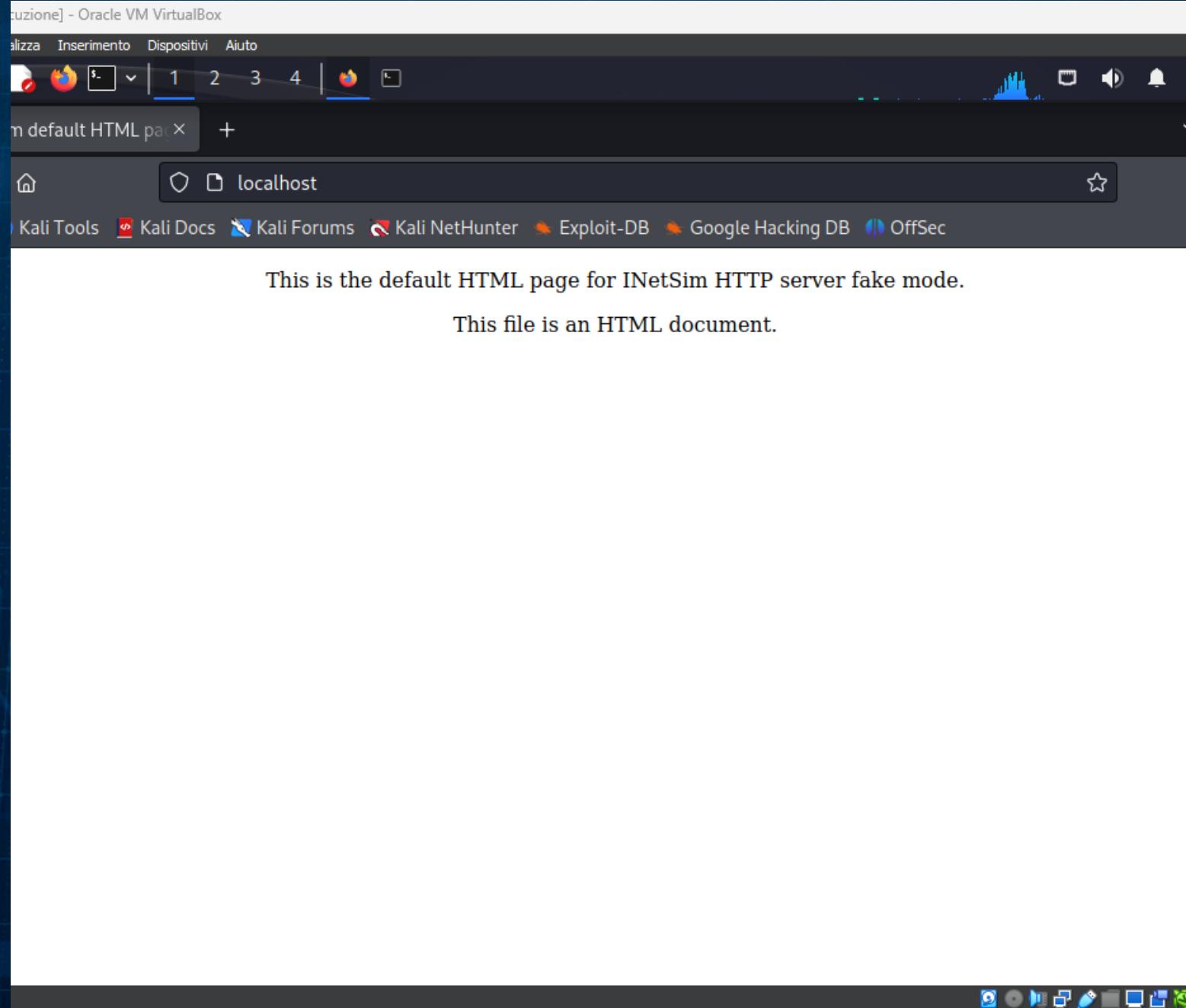
andrea@kali: ~

File Actions Edit View Help

```
zsh: corrupt history file /home/andrea/.zsh_history
[andrea@kali)-[~]
$ sudo inetsim
[sudo] password for andrea:
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 22072) ==
Session ID:    22072
Listening on:  127.0.0.1
Real Date/Time: 2023-11-18 00:51:08
Fake Date/Time: 2023-11-18 00:51:08 (Delta: 0 seconds)
Forking services ...
 * http_80_tcp - started (PID 22074)
 * https_443_tcp - started (PID 22075)
done.
Simulation running.
```

"the quieter you become, the more you

Ora apriamo Mozilla da Kali Linux sulla pagina “<http://localhost>” così possiamo vedere che la risposta verrà visualizzata sulla pagina.



Packet Capture con Wireshark

Avviamo adesso wireshark su kali linux scegliendo di utilizzare la rete LOOPBACK. Andremo a richiamare delle magine su mozilla con il comando get "http://localhost/sample.txt"

The screenshot shows the Wireshark interface capturing traffic from the Loopback interface (lo). The main pane displays a list of network frames, primarily TCP and ICMP packets, between the local host (127.0.0.1) and another host (192.168.50.100). The traffic includes an HTTP request for 'sample.txt' followed by several ICMP destination unreachable responses. The bottom pane shows the raw hex and ASCII representations of the captured data.

No.	Time	Source	Destination	Protocol	Length	Info
69	23.812990582	127.0.0.1	127.0.0.1	TCP	66	80 → 49544 [ACK] Seq=1 Ack=1
70	23.827161505	127.0.0.1	127.0.0.1	TCP	216	80 → 49544 [PSH, ACK] Seq=1 Ack=1
71	23.827267226	127.0.0.1	127.0.0.1	TCP	66	49544 → 80 [ACK] Seq=432 Ack=80
72	23.827288175	127.0.0.1	127.0.0.1	HTTP	324	HTTP/1.1 200 OK (text/html)
73	23.827291770	127.0.0.1	127.0.0.1	TCP	66	49544 → 80 [ACK] Seq=432 Ack=80
74	23.827699323	127.0.0.1	127.0.0.1	TCP	66	49544 → 80 [FIN, ACK] Seq=433 Ack=80
75	23.829000167	127.0.0.1	127.0.0.1	TCP	66	80 → 49544 [FIN, ACK] Seq=433 Ack=80
76	23.829023730	127.0.0.1	127.0.0.1	TCP	66	49544 → 80 [ACK] Seq=433 Ack=80
77	24.575914203	192.168.50.100	192.168.50.100	ICMP	141	Destination unreachable (Host unreachable)
78	24.575922766	192.168.50.100	192.168.50.100	ICMP	138	Destination unreachable (Host unreachable)
79	24.575929085	192.168.50.100	192.168.50.100	ICMP	138	Destination unreachable (Host unreachable)
80	24.575931929	192.168.50.100	192.168.50.100	ICMP	141	Destination unreachable (Host unreachable)
81	24.575936004	192.168.50.100	192.168.50.100	ICMP	140	Destination unreachable (Host unreachable)
82	24.575942183	192.168.50.100	192.168.50.100	ICMP	140	Destination unreachable (Host unreachable)

Frame 1: 141 bytes on wire (1128 bits), 141 bytes captured
Ethernet II, Src: 00:00:00_00:00:00 (00:00:00:00:00:00), Dst: 192.168.50.100 (192.168.50.100)
Internet Protocol Version 4, Src: 192.168.50.100, Dst: 192.168.50.100
Internet Control Message Protocol
Domain Name System (query)

Hex	Dec
0000	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0010	00 7f 92 6f 00 00 40 01 01 36 c0 a8
0020	32 64 03 01 68 fc 00 00 00 00 45 00
0030	40 00 40 11 38 06 c0 a8 32 64 c0 a8
0040	00 35 00 4f b5 16 9c b5 01 00 00 01
0050	00 00 07 63 6f 6e 74 69 6c 65 08 73
0060	63 65 73 07 6d 6f 7a 69 6c 6c 61 03
0070	68 6f 6d 65 6e 65 74 0d 74 65 6c 65
0080	74 61 6c 69 61 02 69 74 00 00 01 00

Nella stessa maniera andremo a scegliere ANY anziche LOOPBACK come in precedenza e saremo in grado attraverso il comando ip.src == 127.0.0.1 verranno visualizzati i pacchetti provenienti da quel IP sorgente.

