

# CSPT0524IT – W3D4 - PRATICA

## Report di Svolgimento dell'Esercizio Pratico W3D4

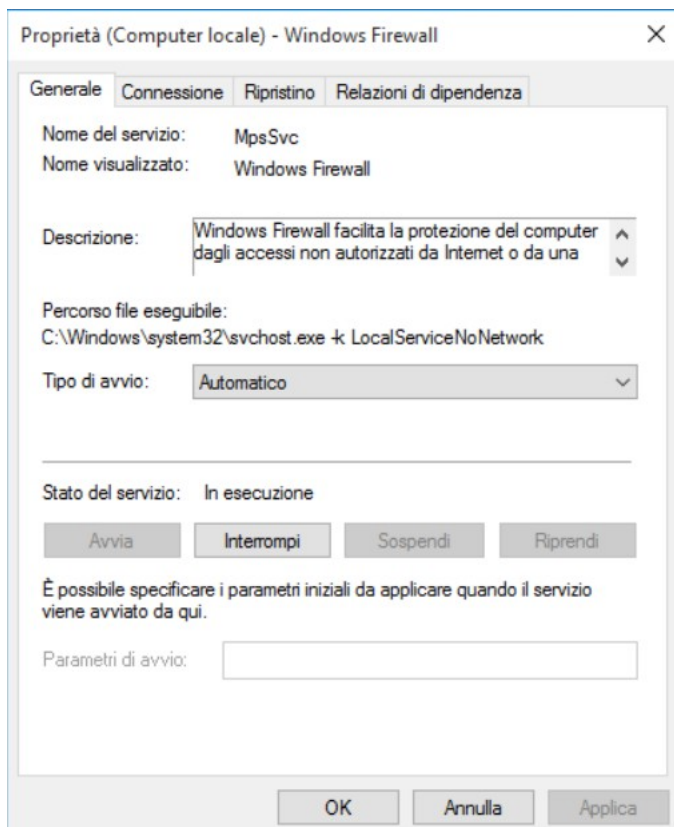
### Esercizio:

- Configurare policy per permettere il ping da macchina Linux a macchina Windows nel nostro laboratorio (Windows firewall)
- Utilizzo dell'utility InetSim per l'emulazione di servizi Internet
- Cattura di pacchetti con Wireshark

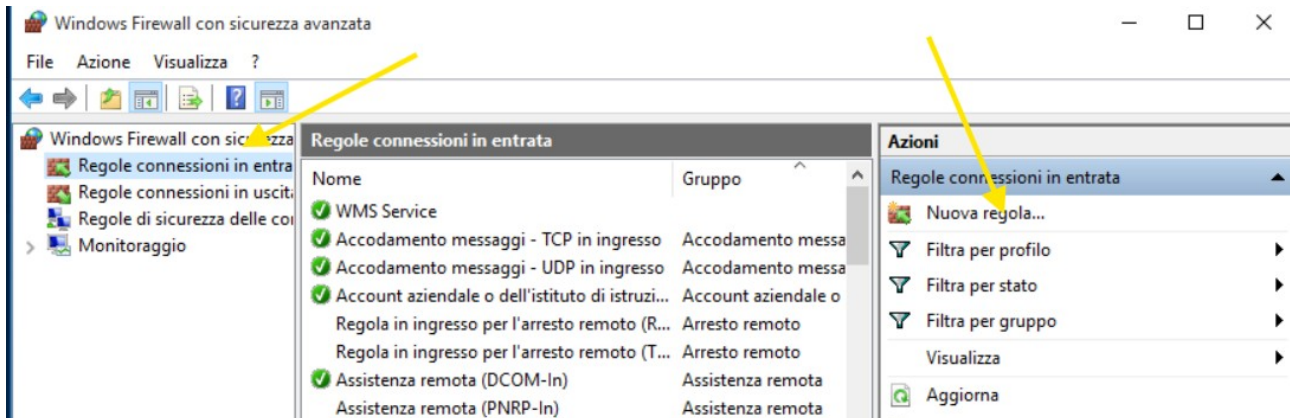
-In questo esercizio ho configurato sul firewall di WIN10 una regola in entrata per il traffico ICMP da tutte le macchine in rete; successivamente ho utilizzato inetsim per hostare un servizio HTTP in localhost per sniffare i pacchetti di rete da Kali con Wireshark.

## 1. Configurazione Firewall WIN10

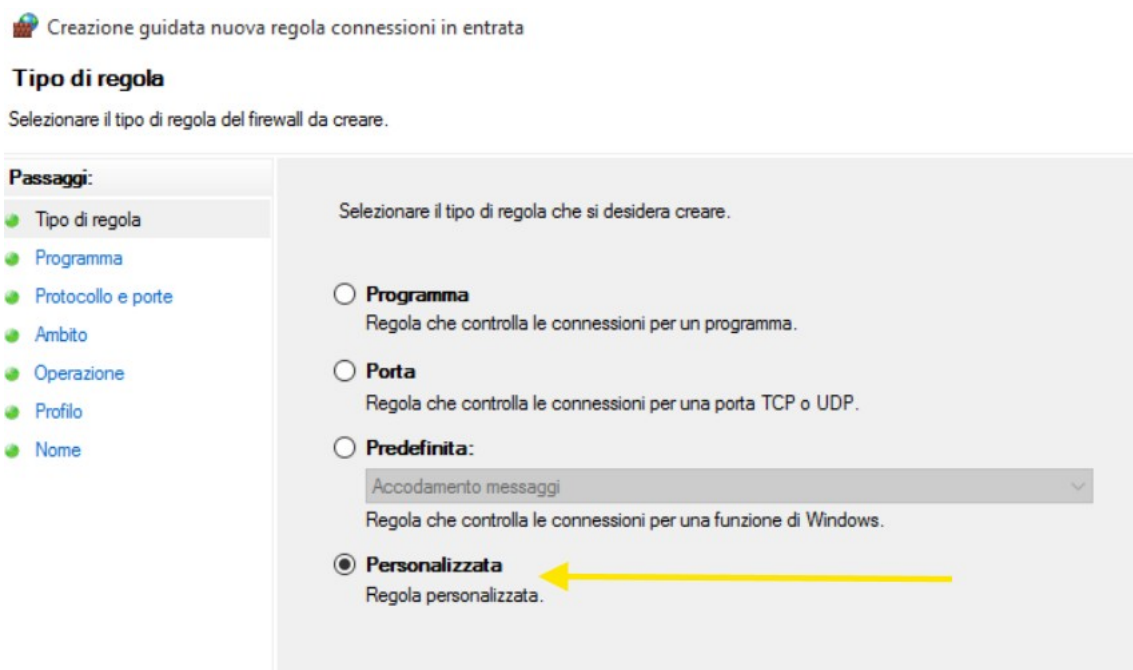
- Ho abilitato il Firewall in modo automatico dai servizi di Windows.



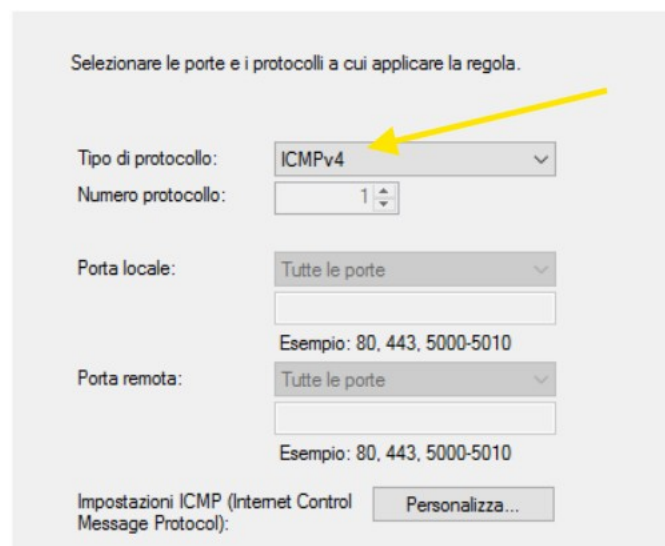
- Ho creato una Nuova Regola nelle connessioni in entrata



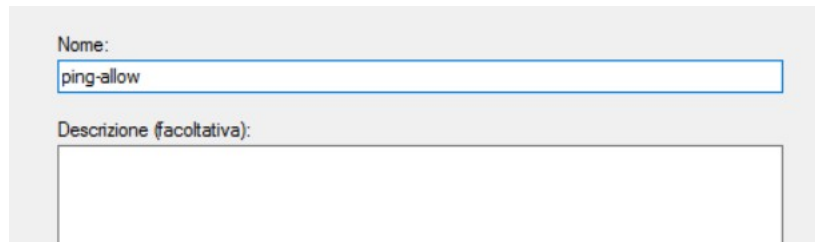
- Abbiamo scelto una regola Personalizzata



- Ho selezionato il protocollo ICMP



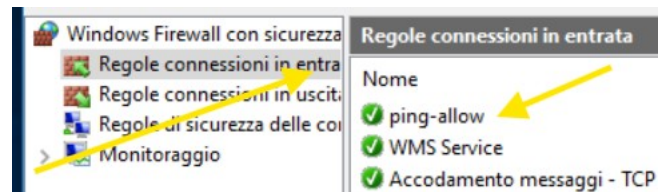
- Ho assegnato un nome identificativo alla regola e premuto fine.



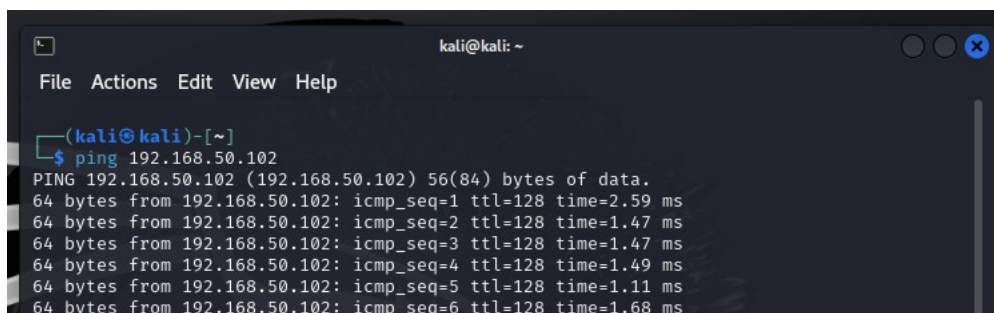
Nome:  
ping-allow

Descrizione (facoltativa):

- Mi sono assicurato che la regola è attiva



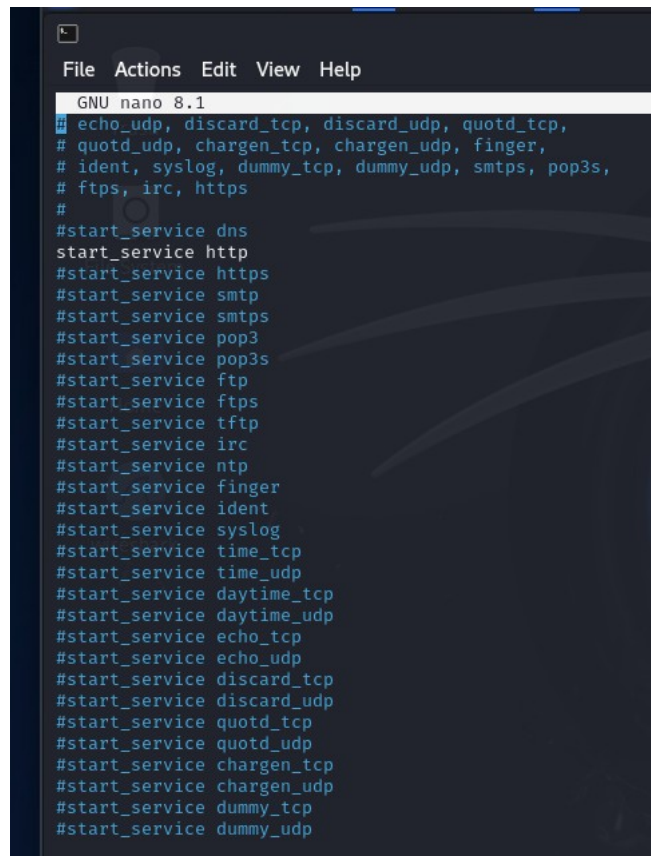
- Ho effettuato il ping dalla Kali sull'IP di WIN10 192.168.50.102 con successo.



```
kali@kali: ~  
File Actions Edit View Help  
(kali@kali)-[~]  
$ ping 192.168.50.102  
PING 192.168.50.102 (192.168.50.102) 56(84) bytes of data:  
64 bytes from 192.168.50.102: icmp_seq=1 ttl=128 time=2.59 ms  
64 bytes from 192.168.50.102: icmp_seq=2 ttl=128 time=1.47 ms  
64 bytes from 192.168.50.102: icmp_seq=3 ttl=128 time=1.47 ms  
64 bytes from 192.168.50.102: icmp_seq=4 ttl=128 time=1.49 ms  
64 bytes from 192.168.50.102: icmp_seq=5 ttl=128 time=1.11 ms  
64 bytes from 192.168.50.102: icmp_seq=6 ttl=128 time=1.68 ms
```

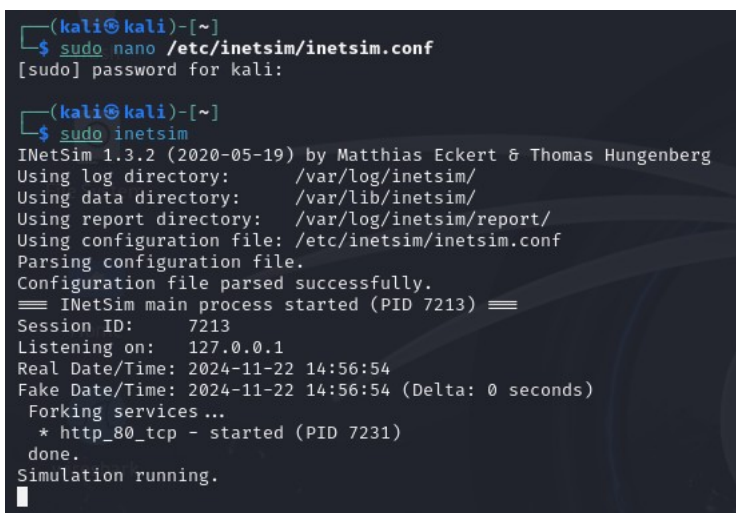
## 2. Utilizzo di InetSim e Wireshark

- Ho disabilitato tutti i servizi in /etc/inetsim/inetsim.conf e ho lasciato decommentato solo il servizio HTTP per non avere troppo traffico.



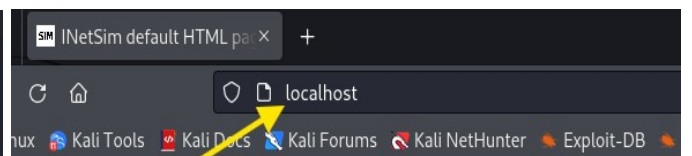
```
GNU nano 8.1
# echo_udp, discard_tcp, discard_udp, quotd_tcp,
# quotd_udp, chargen_tcp, chargen_udp, finger,
# ident, syslog, dummy_tcp, dummy_udp, smtps, pop3s,
# ftps, irc, https
#
#start_service dns
start_service http
#start_service https
#start_service smtp
#start_service smtps
#start_service 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
#start_service time_tcp
#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
```

- Ho avviato InetSim con il comando “sudo inetsim” e sono andato sul browser per chiamare “localhost” nella barra degli indirizzi



```
(kali㉿kali)-[~]
$ sudo nano /etc/inetsim/inetsim.conf
[sudo] password for kali:

(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 7213) ==
Session ID: 7213
Listening on: 127.0.0.1
Real Date/Time: 2024-11-22 14:56:54
Fake Date/Time: 2024-11-22 14:56:54 (Delta: 0 seconds)
Forking services ...
* http_80_tcp - started (PID 7231)
done.
Simulation running.
```



This is the default HTML page for I  
This file is an HTM

- Con Wireshark ho avviato una scansione per sniffare il traffico di rete in localhost e intercettare il traffico HTTP, si nota il syn, syn-ack e ack.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	127.0.0.1	127.0.0.1	TCP	74	45672 → 80 [SYN] Seq= Win=33280 Len=0 MSS=654
2	0.000029326	127.0.0.1	127.0.0.1	TCP	74	80 → 45672 [SYN, ACK] Seq=0 Ack=1 Win=33280 Len=0
3	0.000049424	127.0.0.1	127.0.0.1	TCP	66	45672 → 80 [ACK] Seq=1 Ack=1 Win=33280 Len=0
4	0.000378054	127.0.0.1	127.0.0.1	HTTP	497	GET / HTTP/1.1
5	0.000390511	127.0.0.1	127.0.0.1	TCP	66	80 → 45672 [ACK] Seq=1 Ack=432 Win=33024 Len=0
6	0.094822940	127.0.0.1	127.0.0.1	TCP	216	80 → 45672 [PSH, ACK] Seq=1 Ack=432 Win=33280 Len=0
7	0.094857732	127.0.0.1	127.0.0.1	TCP	66	45672 → 80 [ACK] Seq=132 Ack=151 Win=33152 Len=0
8	0.094895179	127.0.0.1	127.0.0.1	HTTP	324	HTTP/1.1 200 OK (text/html)
9	0.094906023	127.0.0.1	127.0.0.1	TCP	66	45672 → 80 [ACK] Seq=132 Ack=409 Win=33024 Len=0
10	0.095461143	127.0.0.1	127.0.0.1	TCP	66	45672 → 80 [FIN, ACK] Seq=432 Ack=409 Win=33280 Len=0
11	0.108141619	127.0.0.1	127.0.0.1	TCP	66	80 → 45672 [FIN, ACK] Seq=409 Ack=433 Win=33280 Len=0
12	0.108182825	127.0.0.1	127.0.0.1	TCP	66	45672 → 80 [ACK] Seq=133 Ack=410 Win=33280 Len=0

```

Frame 8: 324 bytes on wire (2592 bits), 324 bytes captured (2592 bits) on interface lo, id 0
Ethernet II, Src: 00:00:00_00:00:00 (00:00:00:00:00:00), Dst: 00:00:00_00:00:00 (00:00:00:00:00:00)
Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1
Transmission Control Protocol, Src Port: 80, Dst Port: 45672, Seq: 151, Ack: 432, Len: 258
[2 Reassembled TCP Segments (408 bytes): #6(150), #8(258)]
Hypertext Transfer Protocol
Line-based text data: text/html (10 lines)
<html>\n
<head>\n
<title>INetSim default HTML page</title>\n
</head>\n
<body>\n
<p></p>\n
<p align="center">This is the default HTML page for INetSim HTTP server fake mode.</p>\n
<p align="center">This file is an HTML document.</p>\n
</body>\n
</html>\n

```

Francesco Rinaldi