

CONFIGURAZIONE DI UNA POLICY SU FIREWALL IN AMBIENTE WINDOWS

L'esercizio di oggi è suddiviso in due parti:

FASE 1 - Permettere la comunicazione tra Windows e Kali Linux.

FASE 2 - Packet capture tramite l'utility Wireshark.

FASE 1

Nella FASE 1 era richiesto di configurare una policy su firewall Windows che permettesse la comunicazione tra due distinti dispositivi (Windows e Kali Linux)

Aperto le impostazioni avanzate del Firewall Windows siamo andati a configurare una nuova regola che permettesse la comunicazione tra i dispositivi. Avendo aggiunto un range di IP Address (fig.1) la comunicazione è avvenuta correttamente. La nuova regola al Firewall Windows (fig.2) non bloccava più in entrata e in uscita tra i due dispositivi.

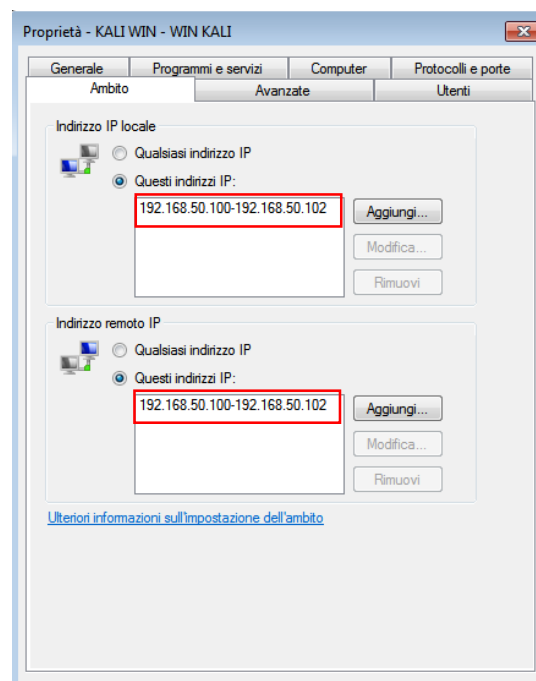


FIG.1

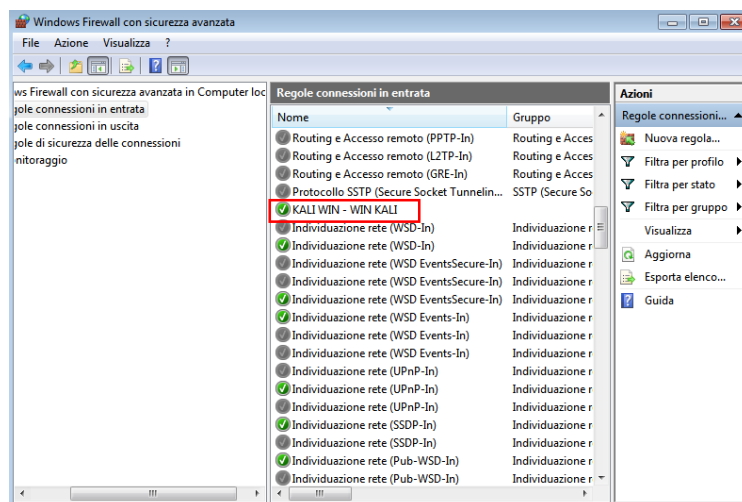


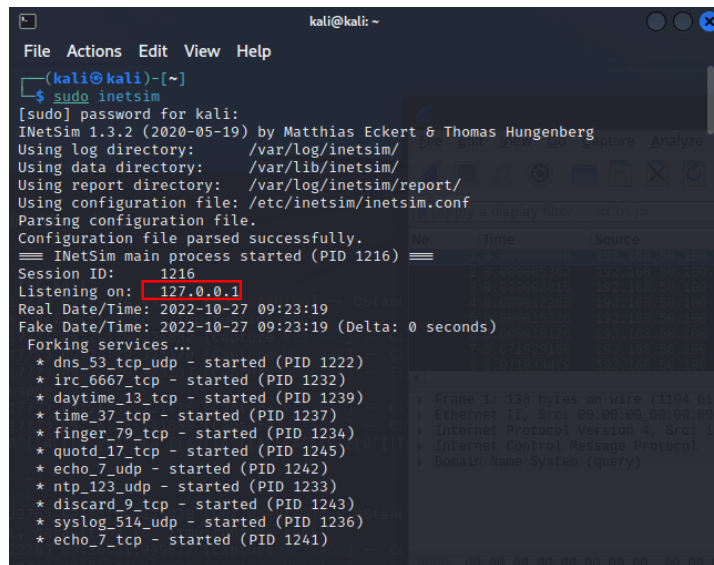
FIG.2

Lista contenente la policy di comunicazione

FASE 2

Nella Fase 2 era richiesto un Packet capture tramite l'utility Wireshark in ambiente Kali Linux.

Inserendo il comando **sudo inetsim** si è potuto accedere ad alcuni servizi di rete(fig.3).



```
kali@kali: ~  
File Actions Edit View Help  
~(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 1216) ==  
Session ID: 1216  
Listening on: 127.0.0.1  
Real Date/Time: 2022-10-27 09:23:19  
Fake Date/Time: 2022-10-27 09:23:19 (Delta: 0 seconds)  
Forking services ...  
* dns_53_tcp_udp - started (PID 1222)  
* irc_6667_tcp - started (PID 1232)  
* daytime_13_tcp - started (PID 1239)  
* time_37_tcp - started (PID 1237)  
* finger_79_tcp - started (PID 1234)  
* quotd_17_tcp - started (PID 1245)  
* echo_7_udp - started (PID 1242)  
* ntp_123_udp - started (PID 1233)  
* discard_9_tcp - started (PID 1243)  
* syslog_514_udp - started (PID 1236)  
* echo_7_tcp - started (PID 1241)
```

FIG.3

Usando il browser abbiamo raggiunto l'indirizzo IP che inetsim ci forniva(fig.4)

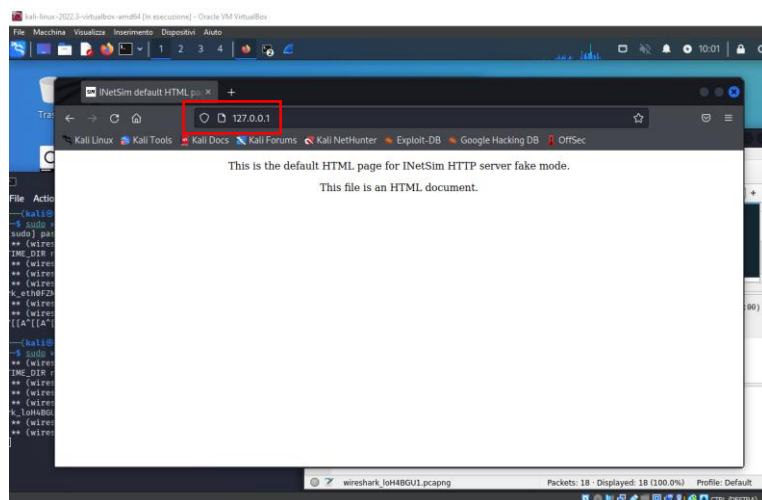


FIG.4

Successivamente inserendo il comando (in una nuova finestra) **sudo wireshark** abbiamo lanciato il tool Wireshark(fig.4) che ci ha permesso di visualizzare il Packet capture(fig.5)

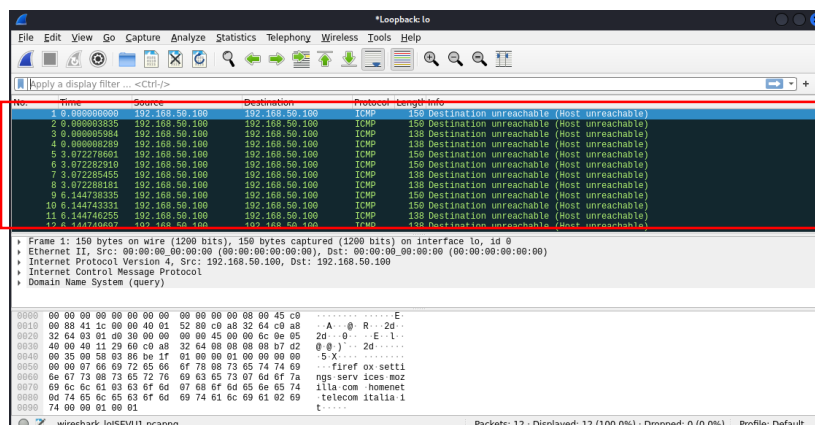
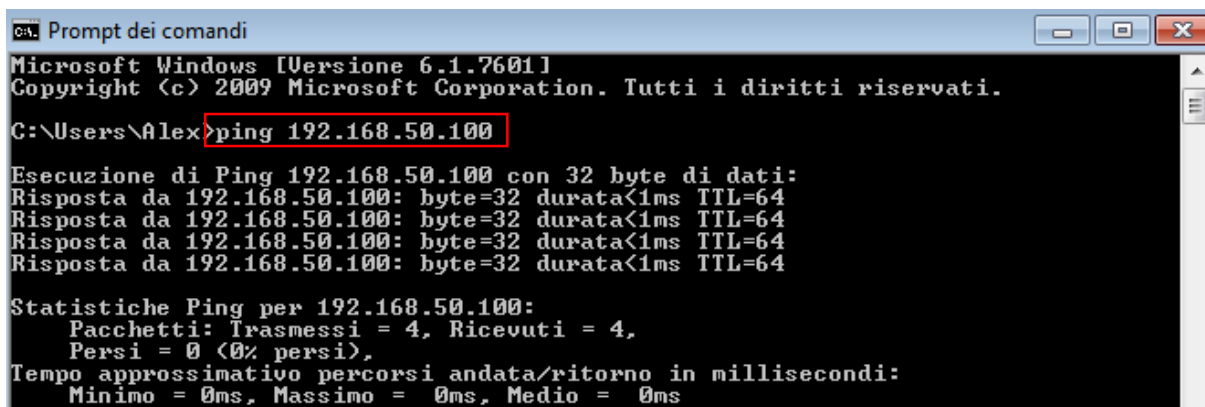


FIG.5

CONCLUSIONI

Aperto il prompt dei comandi sia in ambiente Windows che in Ambiente Kali Linux entrambi riuscivano a comunicare tra di loro tramite il comando ping IP_ADRESS (fig. 6 7)



```
C:\> Prompt dei comandi
Microsoft Windows [Versione 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Tutti i diritti riservati.

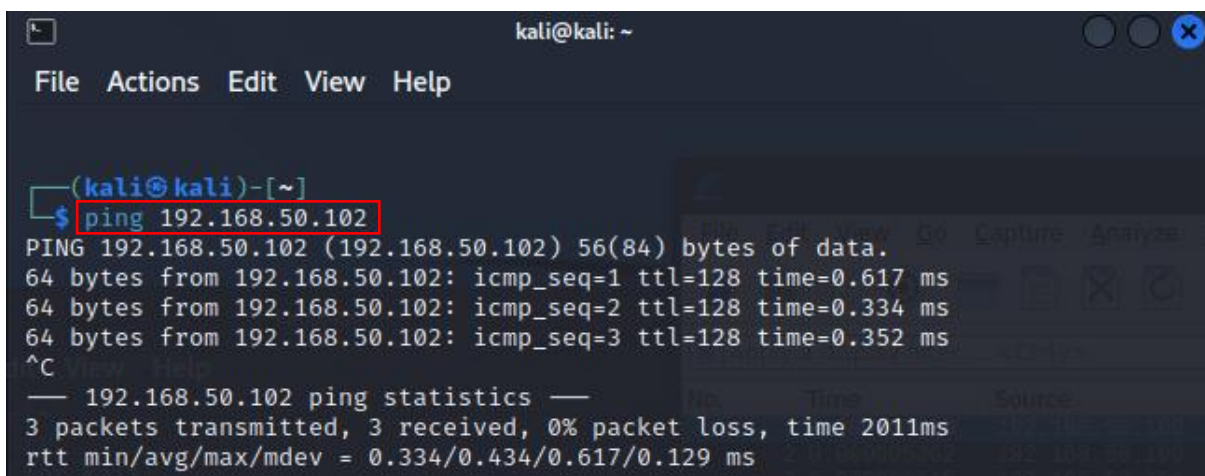
C:\Users\Alex>ping 192.168.50.100

Esecuzione di Ping 192.168.50.100 con 32 byte di dati:
Risposta da 192.168.50.100: byte=32 durata<1ms TTL=64
Risposta da 192.168.50.100: byte=32 durata<1ms TTL=64
Risposta da 192.168.50.100: byte=32 durata<1ms TTL=64
Risposta da 192.168.50.100: byte=32 durata<1ms TTL=64

Statistiche Ping per 192.168.50.100:
    Pacchetti: Trasmessi = 4, Ricevuti = 4,
    Persi = 0 (0% persi),
    Tempo approssimativo percorsi andata/ritorno in millisecondi:
    Minimo = 0ms, Massimo = 0ms, Medio = 0ms
```

FIG.6

PING IN AMBIENTE WINDOWS



```
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=0.617 ms
64 bytes from 192.168.50.102: icmp_seq=2 ttl=128 time=0.334 ms
64 bytes from 192.168.50.102: icmp_seq=3 ttl=128 time=0.352 ms
^C
— 192.168.50.102 ping statistics —
3 packets transmitted, 3 received, 0% packet loss, time 2011ms
rtt min/avg/max/mdev = 0.334/0.434/0.617/0.129 ms
```

FIG.7

PING IN AMBIENTE KALI LINUX