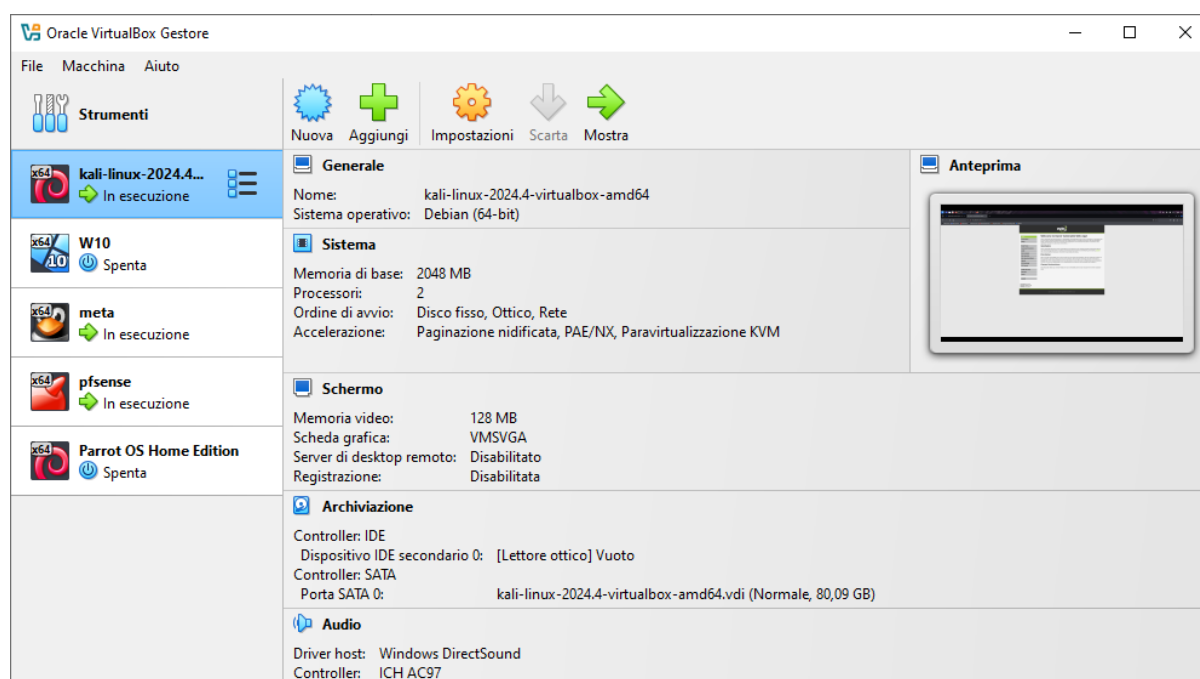


Creazione policy pfsense

Introduzione

Nell'esercitazione di oggi abbiamo il compito di configurare un firewall **pfsense**, e creare diverse regole. Nella nostra configurazione usiamo **Virtualbox** come sistema di virtualizzazione.



Configurazione Pfsense, kali linux e metasploitable

Pfsense è stato configurato con 3 schede di rete:

- Scheda 1 connessa a : **Scheda con bridge.**
 - Scheda 2 connessa a: Rete interna (**intnet**, connessa alla scheda di rete **kali linux**).
 - Scheda 3 connessa a: Rete interna (**meta**, connessa alla scheda di rete di **metasploitable**).
- (La rete bridge consente la comunicazione con l'host fisico, mentre le reti interne consentono per la comunicazione tra VM.)

Kali linux è stato configurato con IP statico: **192.168.50.100**

Editing Statica50

Connection name: Statica50

General Ethernet 802.1X Security DCB Proxy **IPv4 Settings** IPv6 Settings

Method: Manual

Addresses

Address	Netmask	Gateway
192.168.50.100	24	192.168.50.1

DNS servers: 192.168.50.1

Search domains:

DHCP client ID:

☒ Require IPv4 addressing for this connection to complete

Routes...

Cancel Save

Metasploitable configurato con IP statico: **192.168.40.101**

```
GNU nano 2.0.7 File: /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto eth0
iface eth0 inet static
address 192.168.40.101
netmask 255.255.255.0
gateway 192.168.40.1

[Wrote 13 lines]
```

msfadmin@metasploitable:~\$ _

Riavviamo la macchina per rendere effettive le modifiche.

Da questo momento utilizzeremo solo **kali linux**, iniziando dalla configurazione di **pfsense**.

Accediamo alla pagina **192.168.50.1** e si aprirà **pfsense**, dalla sezione Interfaces abilitiamo la **LAN2** che appartiene a **metasploitable**.

Adesso le macchine possono comunicare tra di loro, facciamo un test **ping** verso la macchina **metasploitable**:

```
└─$ ping 192.168.40.101
PING 192.168.40.101 (192.168.40.101) 56(84) bytes of data.
64 bytes from 192.168.40.101: icmp_seq=1 ttl=63 time=10.8 ms
64 bytes from 192.168.40.101: icmp_seq=2 ttl=63 time=7.13 ms
64 bytes from 192.168.40.101: icmp_seq=3 ttl=63 time=1.12 ms
```

Creazione regole firewall

Ci è stato chiesto di creare una regola firewall che blocchi l'accesso alla **DVWA** (su **metasploitable**) dalla macchina **Kali Linux** e ne impedisca di conseguenza lo scan.

Lanciamo uno scan **nmap** per vedere le porte aperte, in questo caso per semplificare abbiamo scansionato dalla porta **20** alla **80**.

```
└─(kali㉿kali)-[~]
└─$ nmap -v -p20-80 192.168.40.101
Starting Nmap 7.95 ( https://nmap.org ) at 2025-02-14 08:26 EST
Initiating Ping Scan at 08:26
Scanning 192.168.40.101 [4 ports]
Completed Ping Scan at 08:26, 0.02s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 08:26
Completed Parallel DNS resolution of 1 host. at 08:26, 0.00s elapsed
Initiating SYN Stealth Scan at 08:26
Scanning 192.168.40.101 [61 ports]
Discovered open port 53/tcp on 192.168.40.101
Discovered open port 25/tcp on 192.168.40.101
Discovered open port 80/tcp on 192.168.40.101
Discovered open port 23/tcp on 192.168.40.101
Discovered open port 22/tcp on 192.168.40.101
Discovered open port 21/tcp on 192.168.40.101
Completed SYN Stealth Scan at 08:26, 0.06s elapsed (61 total ports)
Nmap scan report for 192.168.40.101
Host is up (0.0046s latency).
Not shown: 55 closed tcp ports (reset)
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
53/tcp    open  domain
80/tcp    open  http

Read data files from: /usr/share/nmap
Nmap done: 1 IP address (1 host up) scanned in 0.23 seconds
Raw packets sent: 65 (2.836KB) | Rcvd: 62 (2.492KB)
```

Per fare in modo che il firewall blocchi l'accesso alla **DVWA** potremmo bloccare le connessioni provenienti da tutte le porte, in questo modo:

Firewall / Rules / Edit

Edit Firewall Rule

Action Block
Choose what to do with packets that match the criteria specified below.
Hint: the difference between block and reject is that with reject, a packet (TCP RST or ICMP port unreachable for UDP) is returned to the sender, whereas with block the packet is dropped silently. In either case, the original packet is discarded.

Disabled ☐ Disable this rule
Set this option to disable this rule without removing it from the list.

Interface LAN
Choose the interface from which packets must come to match this rule.

Address Family IPv4
Select the Internet Protocol version this rule applies to.

Protocol TCP
Choose which IP protocol this rule should match.

Source

Source ☐ Invert match Address or Alias 192.168.50.100 /

[Display Advanced](#)

The **Source Port Range** for a connection is typically random and almost never equal to the destination port. In most cases this setting must remain at its default value, any.

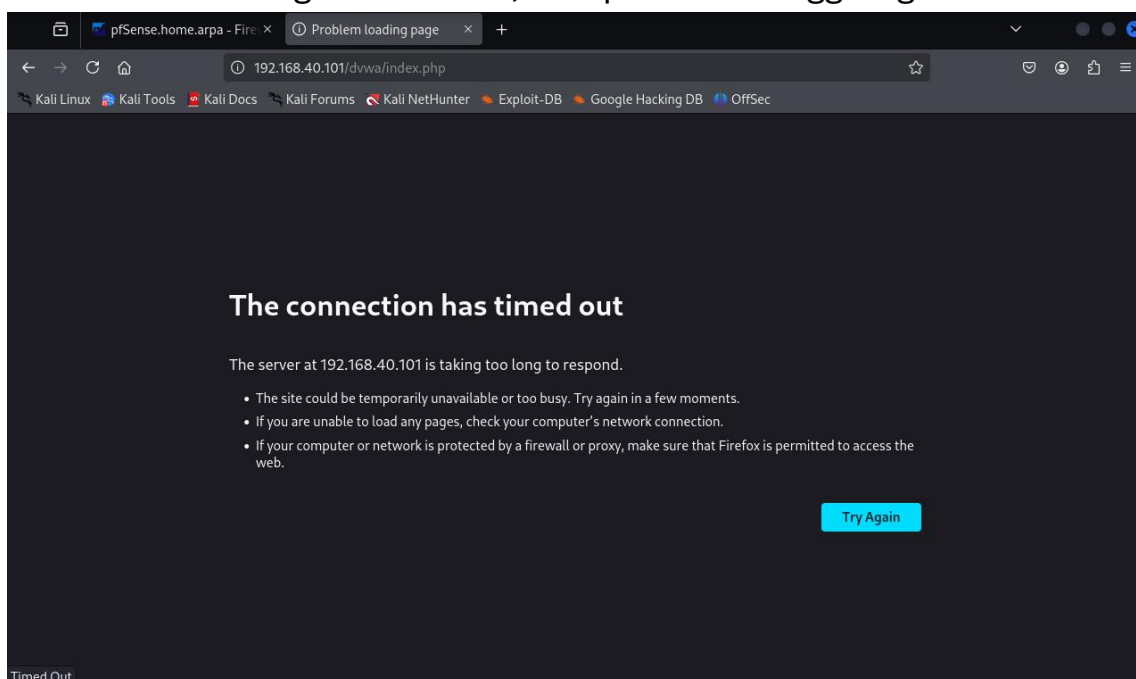
Destination

Destination ☐ Invert match Address or Alias 192.168.40.101 /

Destination Port Range any From Custom To Custom

Specify the destination port or port range for this rule. The "To" field may be left empty if only filtering a single port.

Con questa regola blocchiamo il traffico su tutte le porte e come possiamo vedere nell'immagine in basso, non possiamo raggiungere la **DVWA**.



Adesso apportiamo alcune modifiche, vogliamo fare in modo che il firewall blocchi le porte dalla 20 alla 80, tranne la 53. Rimuoviamo quindi la regola appena aggiunta che blocca le connessioni a tutte le porte da **kali** a **metasploitable**, e modifichiamola con un range di porte o una porta singola, in questo caso inseriamo un range di porte che va dalla **20** alla **80**.

Firewall / Rules / Edit

Edit Firewall Rule

Action Block
Choose what to do with packets that match the criteria specified below.
Hint: the difference between block and reject is that with reject, a packet (TCP RST or ICMP port unreachable for UDP) is returned to the sender, whereas with block the packet is dropped silently. In either case, the original packet is discarded.

Disabled ☐ Disable this rule
Set this option to disable this rule without removing it from the list.

Interface LAN
Choose the interface from which packets must come to match this rule.

Address Family IPv4
Select the Internet Protocol version this rule applies to.

Protocol TCP
Choose which IP protocol this rule should match.

Source

Source ☐ Invert match Address or Alias 192.168.50.100 /

[Display Advanced](#)

The **Source Port Range** for a connection is typically random and almost never equal to the destination port. In most cases this setting must remain at its default value, **any**.

Destination

Destination ☐ Invert match Address or Alias 192.168.40.101 /

Destination Port Range (other) 20 HTTP (80) Custom
From Custom To Custom

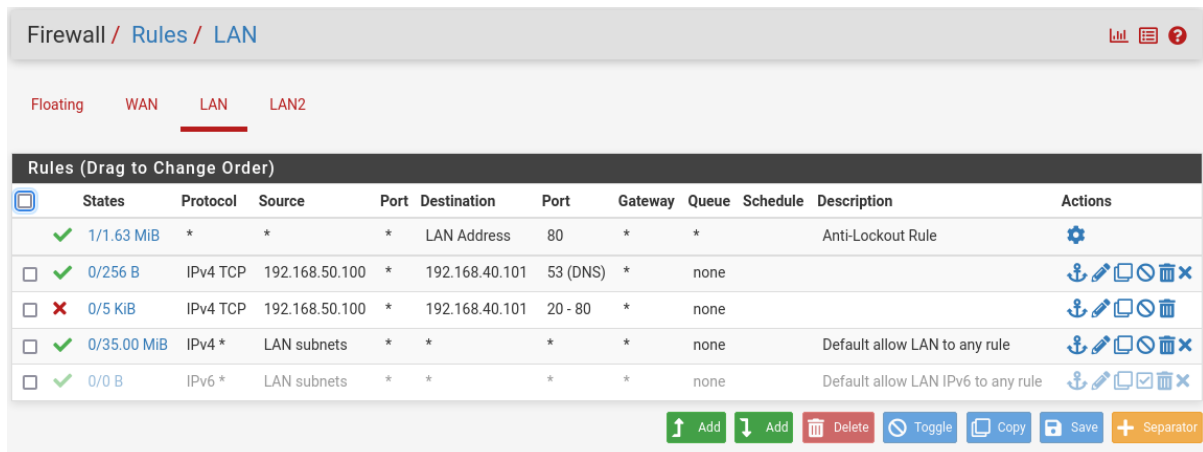
Specify the destination port or port range for this rule. The "To" field may be left empty if only filtering a single port.

Facciamo un test con **nmap**, scansionando dalla porta **20** alla **80**.

```
(kali㉿kali)-[~]
$ nmap -v -p20-80 192.168.40.101
Starting Nmap 7.95 ( https://nmap.org ) at 2025-02-14 08:58 EST
Initiating Ping Scan at 08:58
Scanning 192.168.40.101 [4 ports]
Completed Ping Scan at 08:58, 0.06s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 08:58
Completed Parallel DNS resolution of 1 host. at 08:58, 0.00s elapsed
Initiating SYN Stealth Scan at 08:58
Scanning 192.168.40.101 [61 ports]
Completed SYN Stealth Scan at 08:58, 3.82s elapsed (61 total ports)
Nmap scan report for 192.168.40.101
Host is up (0.040s latency).
All 61 scanned ports on 192.168.40.101 are in ignored states.
Not shown: 61 filtered tcp ports (no-response)

Read data files from: /usr/share/nmap
Nmap done: 1 IP address (1 host up) scanned in 3.99 seconds
Raw packets sent: 126 (5.520KB) | Rcvd: 1 (28B)
```

Come possiamo vedere sono state bloccate tutte le porte dalla **20** alla **80**, ma per completare il nostro obiettivo dobbiamo poter comunicare con la porta **53**, andiamo quindi a creare un'ulteriore regola di pass sulla porta **53** e la inseriamo sopra a quella che blocca le connessioni.



The screenshot shows the Mikrotik WinBox Firewall Rules configuration for the LAN interface. The 'LAN' tab is selected. The rules table is as follows:

Rules (Drag to Change Order)	States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
<input checked="" type="checkbox"/>	1/1.63 MiB	*	*	*	LAN Address	80	*	*		Anti-Lockout Rule	
<input type="checkbox"/>	<input checked="" type="checkbox"/> 0/256 B	IPv4 TCP	192.168.50.100	*	192.168.40.101	53 (DNS)	*	none			
<input type="checkbox"/>	<input checked="" type="checkbox"/> 0/5 KiB	IPv4 TCP	192.168.50.100	*	192.168.40.101	20 - 80	*	none			
<input type="checkbox"/>	<input checked="" type="checkbox"/> 0/35.00 MiB	IPv4 *	LAN subnets	*	*	*	*	none		Default allow LAN to any rule	
<input type="checkbox"/>	<input checked="" type="checkbox"/> 0/0 B	IPv6 *	LAN subnets	*	*	*	*	none		Default allow LAN IPv6 to any rule	

At the bottom of the table are buttons: Add (up arrow), Add (down arrow), Delete, Toggle, Copy, Save, and Separator.

Effettuiamo quindi un test finale con **nmap** per vedere quali porte risultano aperte.

```
(kali㉿kali)-[~]
$ nmap -v -p20-80 192.168.40.101
Starting Nmap 7.95 ( https://nmap.org ) at 2025-02-14 08:58 EST
Initiating Ping Scan at 08:58
Scanning 192.168.40.101 [4 ports]
Completed Ping Scan at 08:58, 0.03s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 08:58
Completed Parallel DNS resolution of 1 host. at 08:58, 0.00s elapsed
Initiating SYN Stealth Scan at 08:58
Scanning 192.168.40.101 [61 ports]
Discovered open port 53/tcp on 192.168.40.101
Completed SYN Stealth Scan at 08:58, 1.55s elapsed (61 total ports)
Nmap scan report for 192.168.40.101
Host is up (0.011s latency).
Not shown: 60 filtered tcp ports (no-response)
PORT      STATE SERVICE
53/tcp    open  domain

Read data files from: /usr/share/nmap
Nmap done: 1 IP address (1 host up) scanned in 1.70 seconds
Raw packets sent: 126 (5.520KB) | Rcvd: 3 (116B)
```

L'unica porta aperta mostrata da **nmap** è la **53**, tutte le altre dalla **20** alla **80** non vengono rilevate dallo scan di **kali linux**.

Conclusioni

Nell'esercitazione di oggi abbiamo configurato un firewall **pfsense** utilizzando **VirtualBox**, con due reti interne e una rete esterna. Abbiamo configurato le macchine virtuali **Kali Linux** e **Metasploitable** con indirizzi IP statici e verificato la loro connettività tramite **ping**.

Successivamente, abbiamo creato le regole firewall personalizzate che controllano il traffico tra le macchine. In particolare, abbiamo aggiunto una regola che blocca l'accesso a tutte le porte dalla 20 alla 80 e quindi alla **DVWA** presente sulla macchina **Metasploitable**, e una regola che consente solo il traffico sulla **porta 53**. La configurazione potrebbe infine essere migliorata tramite l'aggiunta di ulteriori regole o tramite l'aggiunta dei log.