



**Avance final del proyecto**

**Integrantes:**

**Alexandra Mora Brenes**

**Isaac David Robles Meza**

**Jurgen Brenes Arce**

**Kennan Joved Sánchez Garro**

**Curso: Programación avanzada**

**Profesor: Andres Felipe Vargas Rivera**

**III Cuatrimestre 2025**

## 1. Roles del grupo

Nombre	Rol de cada uno	Pequeña descripción
Alexandra Mora Brenes	Es la coordinadora del proyecto y de toda la documentación	Se encarga de reunir toda la información, de la elaboración del documento y organiza la entrega del proyecto
Isaac David Robles Meza	Es el encargado de la parte del repositorio	Se encarga de crear la estructura del repositorio, también el README.md y de mantener el orden de archivos
Jurgen Brenes Arce	Es el encargado de la parte del firewall	Se encarga de desarrollar el script y de verificar que funcione correctamente
Kennan Joved Sánchez Garro	Es el administrador de la VM	Se encarga de configurar la maquina virtual, de ejecutar el script y de tomar evidencias
Santiago Murillo Campos	Es el encargado de la verificación y las pruebas	Se encarga de realizar el escaneo Nmap desde otra VM y de tomar evidencias

## 2. IP de la VM y puertos permitidos

- Sistema operativo utilizado: Ubuntu
- La IP asignada: 158.23.162.91
- Los puertos permitidos:
  - 22 → SSH
  - 80 → HTTP
  - 443 → HTTPS

### 3. Pasos rápidos de creación de la VM en Azure

1. Ingresar a Microsoft Azure (portal.azure.com).
2. Ingresar al servicio de creación de recursos.
3. Seleccionar la opción de "Crear" Máquina Virtual.
4. Anotar los datos básicos solicitados (Ejemplo: Grupo de recursos, Nombre de Máquina Virtual, Región (la más cercana a CR), nivel de seguridad, la imagen del sistema operativo a instalar, la arquitectura de la VM, tamaño, usuario, contraseña, puertos accesibles desde la red pública) etc.
5. Escoger el tamaño de Disco del OS, el tipo de disco (SSD o HDD), y también si se desea eliminar cuando se elimine la VM.
6. Rellenar los datos de las redes, Ejemplo: seleccionar red, ip pública, puertos de entrada públicos y si se desea equilibrio de carga.
7. Administrar todo lo que tiene que ver con: Microsoft Defender Cloud, Identidad, Apagado Automático, copia de seguridad, recuperación ante desastres y actualizaciones del OS.8.
8. Realizar la supervisión: Diagnósticos de arranque.
9. Etiquetas, si se desea.
10. Revisar y crear.

#### Datos de la creación:

**Nombre de la VM**> PrograAvanzadaG5

**Region**> Mexico Central

**Zona de disponibilidad** > Zona 2

**Tipo de seguridad**> Estandar

**Imagen** Ubuntu Server 22.04 LTS - x64 gen 2

**Arquitectura** > x64

**Tamano**> Standard\_D2ls\_v5 - 2 vcpu, 4 GiB de memoria (\$68.26/mes)

**Usuario**> Progra\_G5

**Contrasena**> Grupo5\_Progra

Discos

**Reglas de puerto de entrada**> Puertos de entrada publicos > 80, 443, 22

**Disco del SO**> 30GB

**Tipo de disco**> SSD Estandar

Redes

**Red Virtual**> Grupo5\_Progra

**Subred**> (nuevo) default (10.0.0.0/24)

**Grupo de seguridad de red de NIC**> Basico

**Puertos de entrada publicos**> 80,443,22

**Equilibrio de carga**> Ninguno

Administración

**Microsoft Defender for Cloud**> Habilitar el plan básico de forma gratuita

**Diagnostico**> Todo se deja por defecto

**Contraseña y Usuario:**

Usuario: Progra\_G5

Contraseña: Grupo5\_Progra

#### **4. Buenas prácticas**

Asegurarse de apagar la VM cuando no esté en uso.

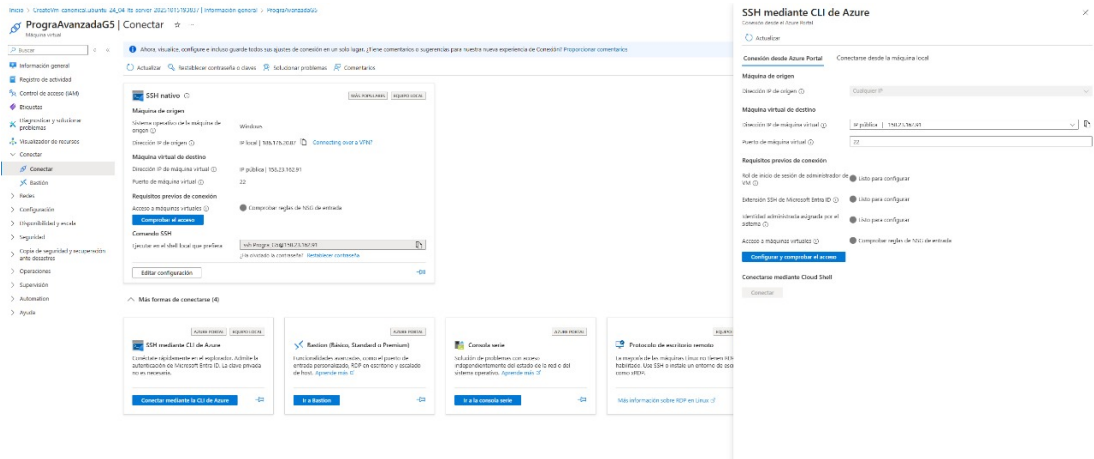
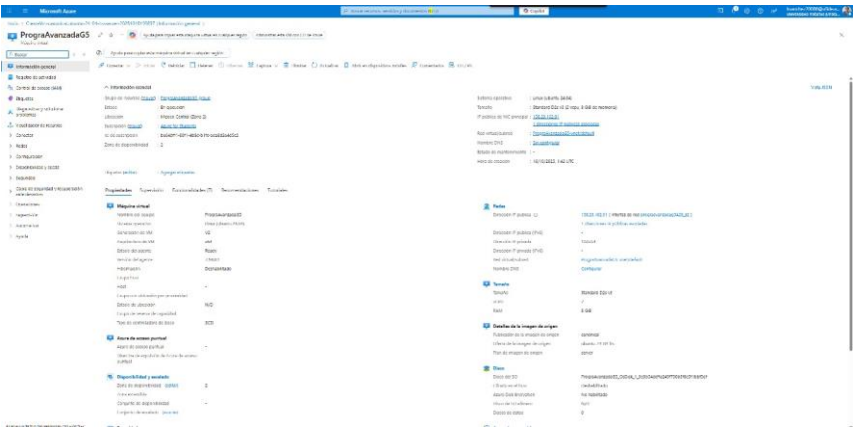
Del todo no exponer puertos innecesarios.

Siempre mantener el firewall activo y sobre todo revisado.

Solamente subir solo los archivos requeridos al repositorio.

Siempre documentar todo con capturas claras y evidencias.

# 5. Evidencias del avance



## SSH mediante CLI de Azure

Conexión desde el Azure Portal

Actualizar

Conexión desde Azure Portal

Conectarse desde la máquina local

Máquina de origen

Dirección IP de origen

Máquina virtual de destino

Dirección IP de máquina virtual

Puerto de máquina virtual

Requisitos previos de conexión

Rol de inicio de sesión de administrador de VM ☐

Extensión SSH de Microsoft Entra ID ☐

Identidad administrada asignada por el sistema ☐

Acceso a máquinas virtuales ☐

Configurar y comprobar el acceso

Conectarse mediante Cloud Shell

Conectar

```
Progra_G5@PrograAvanzada:~$ ssh Progra_G5@158.23.162.91
Warning: Permanently added '158.23.162.91' (ED25519) to the list of known hosts.
Progra_G5@158.23.162.91's password:
Permission denied, please try again.
Progra_G5@158.23.162.91's password:
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.14.0-1012-azure x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

System information as of Thu Oct 16 01:59:25 UTC 2025

System load:  0.0          Processes:      117
Usage of /:   6.4% of 28.02GB Users logged in:    0
Memory usage: 4%          IPv4 address for eth0: 10.0.0.4
Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

26 updates can be applied immediately.
13 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

Progra_G5@PrograAvanzadaG5:~$
```

```
santimur@ubuntuserver:~/blue_team$ chmod +x firewall.basic.sh
santimur@ubuntuserver:~/blue_team$ sudo ./firewall.basic.sh
Guardando una copia de las reglas actuales por seguridad
Limpiando todas las reglas existentes
Estableciendo políticas predeterminadas
Permitidos el tráfico local (loopback)
Permitiendo conexiones ya establecidas o relacionadas
Permitiendo conexión por SSH (puerto 22)
Permitiendo tráfico por web HTTP (puerto 80)
Permitiendo tráfico web seguro HTTPS (puerto 443)
santimur@ubuntuserver:~/blue_team$
```

## 6. Instrucciones rápidas para ejecutar los scripts:

Para lograr correr el primer script que es el (scanner.py) se debe hacer lo siguiente:

### 1.Requisitos principales:

Tener Python 3.x instalado al igual que tener instalado el Nmap y configurado.

### 2. Abrir una terminal, entrar a la carpeta del proyecto la cual es:

cd RUTA/AL/PROYECTO/Proyecto\_progra\_avanzada

### 3. Ejecutar ahí el script indicando la IP objetivo:

Python3 Red\_team/scanner.py <IP\_OBJETIVO>

### 4.Para ejecutarlo en Windows:

Abres PowerShell vas a la carpeta donde esta scanner.py y ejecutas el siguiente código

py scanner.py <IP\_OBJETIVO>

Se van a generar automáticamente una carpeta y un archivo.

```

C:\Users\Isaac>cd "C:\Users\Isaac\Desktop\Proyecto_progra_avanzada\Red_team"

C:\Users\Isaac\Desktop\Proyecto_progra_avanzada\Red_team>dir
El volumen de la unidad C no tiene etiqueta.
El número de serie del volumen es: 6E45-988A

Directorio de C:\Users\Isaac\Desktop\Proyecto_progra_avanzada\Red_team

19/11/2025  11:07 a. m.    <DIR>          .
15/10/2025  06:08 p. m.    <DIR>          ..
15/10/2025  06:53 p. m.          1 .gitkeep
19/11/2025  10:59 a. m.    <DIR>          docs
19/11/2025  11:11 a. m.          1,572 README_scanner.txt
19/11/2025  10:40 a. m.          3,389 scanner.py
                3 archivos                4,962 bytes
                3 dirs 30,939,508,736 bytes libres

C:\Users\Isaac\Desktop\Proyecto_progra_avanzada\Red_team>py scanner.py 158.23.162.91
[INFO] Ejecutando nmap...
[CMD] nmap -sS -sV -Pn -p 22,80,443 158.23.162.91
[OK] Evidencia guardada en: docs\evidencias\nmap_scan.txt

[INFO] Puertos abiertos detectados:
- 22/tcp open  ssh      OpenSSH 9.6p1 Ubuntu 3ubuntu13.14 (Ubuntu Linux; protocol 2.0)

[LISTO] Escaneo completado.

```

```

└===== NMAP SCAN =====
Fecha y hora: 2025-11-19 18:28:16
IP objetivo: 158.23.162.91
Comando: nmap -sS -sV -Pn -p 22,80,443 158.23.162.91

===== STDOUT =====
Starting Nmap 7.97 ( https://nmap.org ) at 2025-11-19 18:28 -0600
Nmap scan report for 158.23.162.91
Host is up (0.10s latency).

PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 9.6p1 Ubuntu 3ubuntu13.14 (Ubuntu Linux; protocol 2.0)
80/tcp    closed http
443/tcp   closed https
Service Info: OS: Linux; CPE: cpe:/o:linux:linux\_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 3.85 seconds

===== STDERR =====

```

Para lograr correr el segundo script que es el (os\_audit.py) se debe hacer lo siguiente:

### 1.Requisitos principales:

Tener Python 3.x instalado al igual que tener los permisos para leer usuarios, servicios y puertos del sistema.

### 2.En VS Code abrir una terminal integrada e ir a la carpeta llamada:

```
cd blue_team
```

### 3:Ahí se debe ejecutar lo siguiente:

```
python3 os_audit.py
```

### 4.Ejecutar en la terminal de Linux lo siguiente:

```
cd RUTA/AL/PROYECTO/Proyecto_progra_avanzada
```

```
python3 blue_team/os_audit.py
```

### 5. Debe de generar archivos automaticamente.

```
Progra_G5@PrograAvanzadaG5:~$ ls
Desktop Documents Downloads Music Pictures Public Templates Videos os_audit.py thinclient_drives
Progra_G5@PrograAvanzadaG5:~$ python3 os_audit.py
=====
BLUE TEAM - AUDITORIA DEL SISTEMA OPERATIVO
=====
Fecha: 2025-11-19 22:55:43
=====

1. Obteniendo usuarios del sistema...
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin
_apt:x:42:65534::/nonexistent:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:998:998:systemd Network Management:/usr/sbin/nologin
systemd-timesync:x:996:996:systemd Time Synchronization:/usr/sbin/nologin
dhcpcd:x:100:65534:DHCP Client Daemon,,,:/usr/lib/dhcpcd:/bin/false
messagebus:x:101:101::/nonexistent:/usr/sbin/nologin
syslog:x:102:102::/nonexistent:/usr/sbin/nologin
systemd-resolve:x:991:991:systemd Resolver:/usr/sbin/nologin
uidd:x:103:103::/run/uidd:/usr/sbin/nologin
tss:x:104:104:TPM software stack,,,:/var/lib/tpm:/bin/false
sshd:x:105:65534::/run/sshd:/usr/sbin/nologin
pollinate:x:106:1::/var/cache/pollinate:/bin/false
tcpdump:x:107:108::/nonexistent:/usr/sbin/nologin
landscape:x:108:109::/var/lib/landscape:/usr/sbin/nologin
fwupd-refresh:x:990:990:Firmware update daemon:/var/lib/fwupd:/usr/sbin/nologin
```



```

Progra_G5@PrograAvanzadaG5:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos  blue_team  docs  os_audit.py  thinclient_drives
Progra_G5@PrograAvanzadaG5:~$ cd blue_team
Progra_G5@PrograAvanzadaG5:~/blue_team$ ls
log_events.txt
Progra_G5@PrograAvanzadaG5:~/blue_team$ cat log_events.txt
=====
AUDITORIA DEL SISTEMA OPERATIVO - UBUNTU SERVER
=====
Fecha: 2025-11-19 22:55:43

=====
USUARIOS DEL SISTEMA (/etc/passwd)
=====

root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin
_apt:x:42:65534:/nonexistent:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:998:998:systemd Network Management:/usr/sbin/nologin
systemd-timesync:x:996:996:systemd Time Synchronization:/usr/sbin/nologin
dhcpcd:x:100:65534:DHCP Client Daemon,,:/usr/lib/dhcpcd:/bin/false
messagebus:x:101:101:/nonexistent:/usr/sbin/nologin
syslog:x:102:102:/nonexistent:/usr/sbin/nologin
systemd-resolve:x:991:991:systemd Resolver:/usr/sbin/nologin
uidd:x:103:103:/run/uidd:/usr/sbin/nologin

```

## 7. Breve resumen del propósito que tiene cada script.

### 1) Script de Red Team (scanner.py).

Este script realiza un escaneo con el comando Nmap desde Python para así analizar que tal esta la seguridad de la maquina que tenemos como objetivo. El propósito de este script es lograr identificar los puertos abiertos, los servicios activos y algunas versiones que pueden presentar distintas vulnerabilidades. Básicamente el script recibe la IP como un parámetro y guarda automáticamente los resultados en un archivo de evidencia para el análisis final.

### 2) Script de Blue Team (os\_audit.py).

Este script realiza una auditoria bastante básica del sistema operativo y su objetivo es lograr registrar la información importante para la defensa como lo son los usuarios existentes en este sistema, los servicios que se ven activos y los puertos que se encuentran abiertos. La información recaudada se guarda en archivos de log que sirven como evidencia y ayudan a detectar configuraciones muy inseguras o bien actividades sospechosas.

## Packet attack:

Para que funciona:

Este script de packet attack.py nos ayuda a simular un ataque SYN 100% controlado para lograr generar trafico de red que logre ser detectado por el Blue Team sin perjudicar los servicios.

Es necesario Python 3, Scapy y también de suma importancia los permisos de administrador.

Este código se ejecuta desde la carpeta principal del proyecto utilizando:

Sudo python3 red\_team/packet\_attack.py.

Este código por defecto utiliza el puerto 80 y envía 20 paquetes SYN.

Este script genera paquetes SYN con algunos intervalos y también muestra en la consola la IP objetivo, el puerto, la fecha, la cantidad enviada y sobre todo no genera archivos. Esta es una herramienta de suma importancia para la practica de ejercicios de ciberseguridad.

Resultados del packet\_attack

Scanner:

```
(kennan@SouLink)-[~/Red_team]
$ sudo python scanner.py 4.248.145.117
[INFO] Ejecutando nmap...
[CMD] nmap -sS -sV -Pn -p 22,80,443 4.248.145.117
[OK] Evidencia guardada en: docs/evidencias/nmap_scan.txt

[INFO] Puertos abiertos detectados:
- 22/tcp open      ssh      OpenSSH 9.6p1 Ubuntu 3ubuntu13.14 (Ubuntu Linux; protocol 2.0)
[LISTO] Escaneo completado.
```

Packet:

```
(kennan@SoulLink)-[~/Red_team]
$ sudo python packet_attack.py 4.248.145.117 22 20
=====
RED TEAM - ATAQUE SYN CONTROLADO
=====
[INFO] IP objetivo : 4.248.145.117
[INFO] Puerto      : 22
[INFO] Cantidad    : 20 paquetes SYN
=====
[ENVÍO] Paquete SYN #1 -> 4.248.145.117:22
[ENVÍO] Paquete SYN #2 -> 4.248.145.117:22
[ENVÍO] Paquete SYN #3 -> 4.248.145.117:22
[ENVÍO] Paquete SYN #4 -> 4.248.145.117:22
[ENVÍO] Paquete SYN #5 -> 4.248.145.117:22
[ENVÍO] Paquete SYN #6 -> 4.248.145.117:22
[ENVÍO] Paquete SYN #7 -> 4.248.145.117:22
[ENVÍO] Paquete SYN #8 -> 4.248.145.117:22
[ENVÍO] Paquete SYN #9 -> 4.248.145.117:22
[ENVÍO] Paquete SYN #10 -> 4.248.145.117:22
[ENVÍO] Paquete SYN #11 -> 4.248.145.117:22
[ENVÍO] Paquete SYN #12 -> 4.248.145.117:22
[ENVÍO] Paquete SYN #13 -> 4.248.145.117:22
[ENVÍO] Paquete SYN #14 -> 4.248.145.117:22
[ENVÍO] Paquete SYN #15 -> 4.248.145.117:22
[ENVÍO] Paquete SYN #16 -> 4.248.145.117:22
[ENVÍO] Paquete SYN #17 -> 4.248.145.117:22
[ENVÍO] Paquete SYN #18 -> 4.248.145.117:22
[ENVÍO] Paquete SYN #19 -> 4.248.145.117:22
[ENVÍO] Paquete SYN #20 -> 4.248.145.117:22
=====
[LISTO] Ataque SYN controlado finalizado.
[LOG] Fecha/hora de la prueba: 2025-12-10 18:13:00
(kennan@SoulLink)-[~/Red_team]
$
```

Sniffer:

Este script llamado sniffer\_defense.py pertenece al Blue Team, es una herramienta diseñada para lograr detectar los paquetes SYN que sean sospechosos y además ayuda a registrar posibles IPs que sean atacantes.

Es necesario Python 3, Scapy y de suma importancia los permisos de administrador, y se debe ejecutar dentro de la máquina virtual del Blue Team.

Se inicia desde la carpeta principal del proyecto con el siguiente comando:

sudo python3 blue\_team/sniffer\_defense.py, y se puede detener con Ctrl+C.

Este programa identifica automáticamente la IP local, además registra la actividad en defense\_log.txt y guarda las IPs bloqueadas en blocked\_ips.txt. Aparte, sugiere el comando UFW para lograr bloquear manualmente una dirección:

sudo ufw deny from <IP\_ATACANTE>.

Esta es una herramienta de defensa básica para monitoreo y respuesta ante algún tráfico sospechoso.

```
Progra_G5@PrograAvanzadaG5:~/proyecto2/blue_team$ cat blocked_ips.txt
# IPS BLOQUEADAS
# Formato: IP # Fecha de bloqueo
# Creado: 2025-12-10 23:12:21

123.210.145.127 # 2025-12-10 23:12:25
186.176.20.87 # 2025-12-10 23:12:46
161.35.159.88 # 2025-12-10 23:12:47
```

```
Progra_G5@PrograAvanzadaG5:~/proyecto2/blue_team$ cat defense_log.txt
```

```
=====
NUEVA SESION - 2025-12-10 23:12:21
IP Protegida: 172.16.0.4
=====

[2025-12-10 23:12:25] SYN SOSPECHOSO
  IP: 123.210.145.127
  Puerto: 22
  Razon: Puerto fuera del rango comun

[2025-12-10 23:12:26] SYN SOSPECHOSO
  IP: 123.210.145.127
  Puerto: 22
  Razon: Puerto fuera del rango comun

[2025-12-10 23:12:46] SYN SOSPECHOSO
  IP: 186.176.20.87
  Puerto: 22
  Razon: Puerto fuera del rango comun

[2025-12-10 23:12:47] SYN SOSPECHOSO
  IP: 161.35.159.88
  Puerto: 22
```

## 8. Conclusiones

- Todo el entorno del trabajo fue configurado de la manera correcta, con la estructura del proyecto que fue establecida por el profesor y además realizamos el firewall 100% funcional.
- Se reviso que la conectividad a través de los puertos 22, 80 y 443 de manera correcta.
- Se cumplio con la organización del equipo de trabajo y se cumplieron los requerimientos de seguridad que se pedían.