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. Ingredar 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.

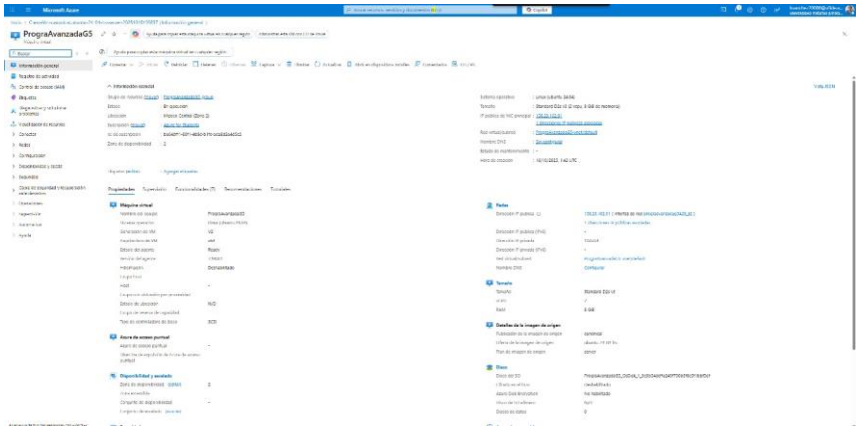
Del todo no exponer puertos innecesarios.

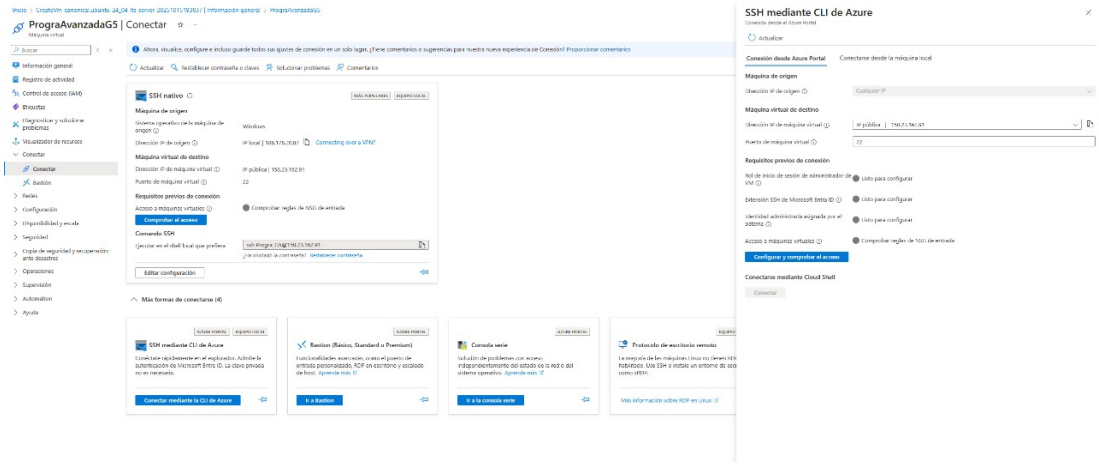
Siempre mantener el firewall activo y sobre todo revisado.

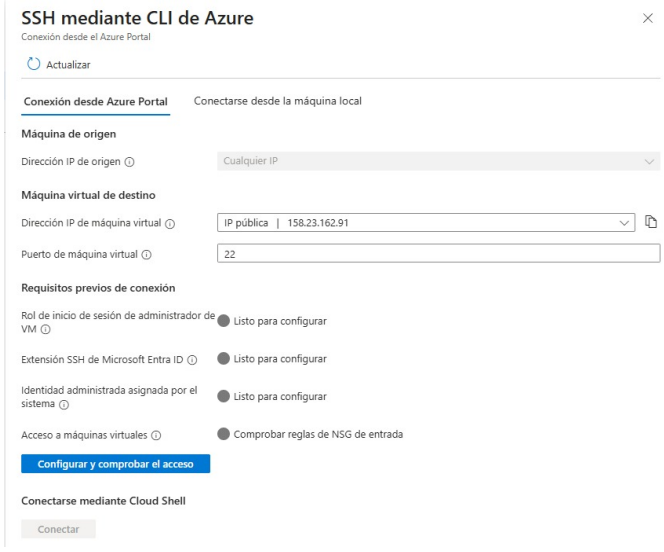
Solamente subir solo los archivos requeridos al repositorio.

Siempre documentar todo con capturas claras y evidencias.

5. Evidencias del avance







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:

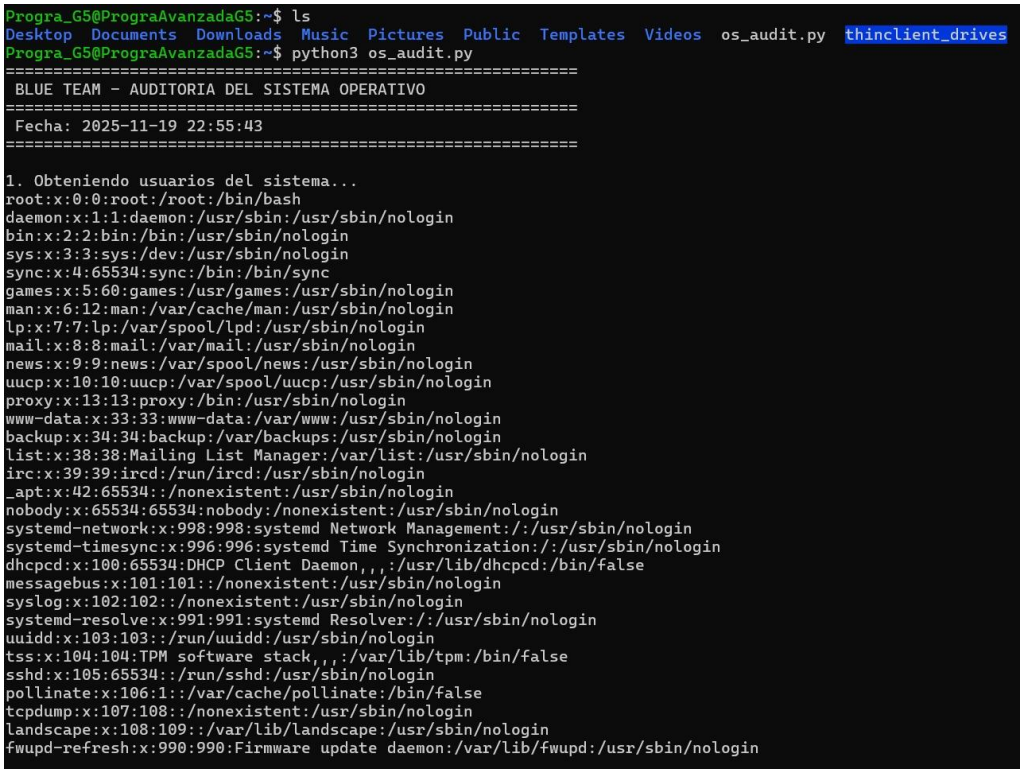
python3 os_audit.py

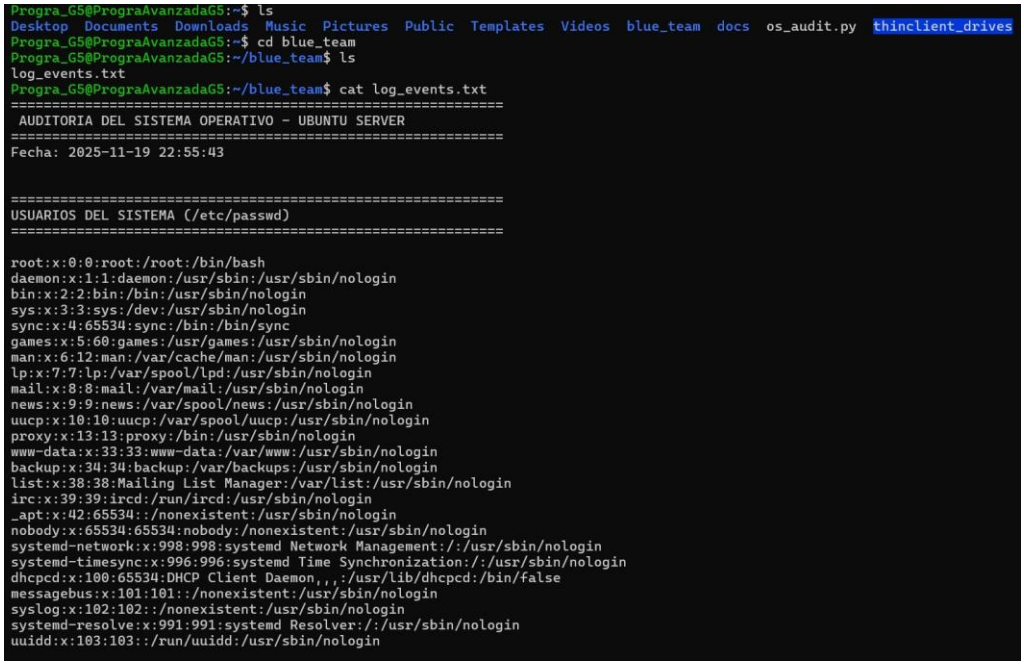
4.Ejecutar en la terminal de Linux lo siguiente:

cd RUTA/AL/PROYECTO/Proyecto_progra_avanzada

python3 blue_team/os_audit.py

5. Debe de generar archivos automaticamente.





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:

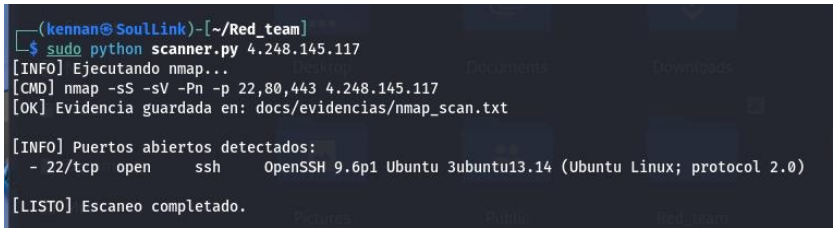
Sudo python3 red_team/packet_attack.py.

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:



Packet:

Sniffer:

Este script llamado sniffer_defense.py pertenece al Blue Team, es una herramienta diseñada para logar 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, ademas 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.