UNIVERSIDAD TECNOLÓGICA NACIONAL

Tecnicatura universitaria en programación a distancia

Trabajo Práctico Integrador:
Seguridad en el Sistemas Operativos

Materia: Arquitectura y Sistemas Operativos
Profesor: Mauricio Gabriel Pasti
Fecha de Entrega: 05 de junio de 2025

Alumnos:
Barrutia Milagros – milagrosbarrutia1995@gmail.com
Caballero Julieta – julieta.carolina.caballero.00@gmail.com

Índice

1. Introducción

La seguridad en sistemas operativos Linux es importante para garantizar la integridad y confidencialidad de los datos informáticos. Las amenazas cibernéticas y la exposición de servidores y equipos conectados a redes públicas o privadas pueden volverlo vulnerable. La vulnerabilidad en un sistema puede ser explotada y comprometer la confidencialidad de la información de la máquina.
Este trabajo aborda cómo detectar puertos que pueden comprometer la seguridad en sistemas Linux, aplicando configuraciones seguras e implementando herramientas clave para mitigar amenazas.
Con un ejemplo se muestra la importancia de mantener el sistema seguro y no propenso a sufrir amenazas.

2. Marco Teórico

Linux se basa en un modelo de permisos de usuarios e incorpora funciones de privilegios que facilitan y permiten el monitoreo del sistema.
Aunque Linux cuenta con una arquitectura abierta y su frecuente uso en servidores no está exento de ser objeto de ataques, aunque generalmente se considera más seguro que otros sistemas operativos. Pero esta seguridad depende en gran parte de una correcta configuración y mantenimiento continuo del sistema. Su seguridad depende de la configuración, actualizaciones y mantenimiento preventivo constante.

En un entorno digital interconectado, los sistemas operativos representan el núcleo de las infraestructuras tecnológicas. En particular, Linux es ampliamente utilizado por su estabilidad, eficiencia y naturaleza de código abierto.
Entre las herramientas analizadas se encuentra Nmap, una aplicación de código abierto que ayuda a la detección, descubrimiento, escaneo, identificación de puertos maliciosos o sospechosos que están abiertos o activos. Con este código es posible anticipar las vulnerabilidades que puede tener Linux.
Se estudia el uso de iptables que son una herramienta de firewall (sistema de seguridad que regula y filtra el tráfico de red entrante y saliente) cuya función es filtrar el tráfico de entrada como de salida se reduce el ataque y fortalece la defensa contra amenazas externas.

El tipo de vulnerabilidad trabajado corresponde al tipo de conexiones no autorizadas, ya que implementa la utilización de “rootkits”, programas similares a los troyanos, que se instalan en un equipo reemplazando a una herramienta o servicio legítimo del sistema operativo. Los “rootkits”, además de cumplir con las funciones de la herramienta o servicio que reemplazan en el equipo para no despertar sospechas, incorporan otras funciones ocultas que facilitan, entre otras cosas, el control remoto del equipo comprometido.

A través de un enfoque teórico-práctico, se busca comprender e implementar medidas efectivas de protección y mantenimiento preventivo.

3. Caso Práctico

Para el caso práctico se utilizó la máquina virtual Google Cloud Shell, que es un entorno de línea de comandos basado en la nube que corre sobre una máquina virtual con Linux. Simulando el entorno de un servidor expuesto a posibles amenazas externas. El objetivo es identificar puertos abiertos, evaluar posibles vulnerabilidades y aplicar los comandos para evitar riesgos.
Se mostrará a través de comandos en Shell como escanear puertos abiertos y particularmente el puerto 31337/tcp que es un puerto usuario-servidor malicioso diseñado en los años ´90 por un grupo de hackers para demostrar cómo el sistema está vulnerable a sufrir ataques, transferir archivos, robar información confidencial y tener control total sobre la máquina.

4. Metodología

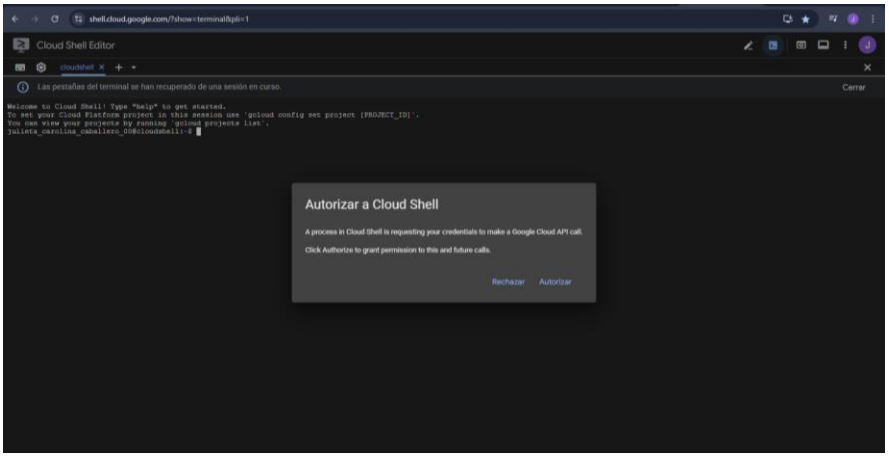
Se simula el escaneo de puertos del servidor de Nmap para analizarlos y detectar algún puerto que pueda ser inusual. Nmap nos brinda la posibilidad de escanear los puertos a los distintos hosts y ver en qué estados se encuentran esos puertos (abiertos, cerrados o

filtrados). Para probar esta herramienta se utiliza la VM de Google Cloud Shell que simula un entorno Linux. A continuación, se presentan los comandos ejecutados y su explicación. Se enumeran los pasos, para luego visualizar las imágenes en la sección de resultados.

PASO 1. Ingresamos en la consola de Google Cloud Shell, un entorno Linux temporal brindado por Google. En este espacio escribiremos los comandos para realizar la práctica.

1.1 https://shell.cloud.google.com/?show=terminal&pli=1

1.2 Se autoriza a Google para ingresar con la cuenta con la que quiera ingresar el usuario.



1.3 Se abre una terminal en un entorno Linux virtual.

1.4 Para utilizar Nmap en la consola de Cloud Shell, primero hay que instalarla, con el comando:

```
sudo apt install nmap
```

PASO 2. Luego se ejecuta el comando:

```
nmap scanme.nmap.org
```

Este comando es una posibilidad que brinda Nmap para probar la herramienta de escaneo de puertos. Escanea los puertos del servidor de la herramienta.[1]

Este comando escanea los puertos del servidor Nmap y arroja los resultados volcados en la tabla:

[1] https://nmap.org/man/es/man-examples.html

Port(Puerto)	STATE(Estado)	SERVICE (Servicio
22/tcp	open	ssh
80/tcp	open	http
9929/tcp	open	nping-echo
31337/tcp	open	Elite

PASO 3. DETECCIÓN DE PUERTOS INUSUALES.
Analizando los puertos encontrados se encontró que el puerto 31337 se ha asociado con herramientas de hacking como Back Orífice[2]. Este programa permite controlar toda la computadora del usuario desde un lugar remoto, ya sea modificar archivos, leerlos o controlar programas

Para que el atacante pueda controlar la computadora, la víctima debe instalar de forma silenciosa un archivo que contiene el servidor de Back Orífice, ya sea por correos electrónicos falsos, o abrir archivos de fuentes no seguras[3]. De esta forma el servidor de Back Orífice queda a la espera de las órdenes que ejecute el atacante. Para recibir las órdenes abre un puerto de red, el puerto número **31337**.

PASO 4. BLOQUEO DE PUERTO INUSUAL.
Detectar este puerto abierto puede ser indicador de algo inusual o peligroso. Por esta razón, se ejecutará iptables**,** una herramienta de filtrado de paquetes en Linux que permite controlar el tráfico de red. Usaremos su función de restringir puertos no deseados.

El comando ejecutado es:

```
sudo iptables -A INPUT- -p tcp —-deport 31337 -j DROP
```

Este comando descarta cualquier paquete que sea enviado desde el puerto 31337

Se quisieron guardar estos cambios usando el comando estudiado en la materia pero como Cloud Shell es un entorno temporal no es posible esta función.

[2] . Fue desarrollada en 1998 por un grupo de hackers llamado **Cult of the Dead Cow**, con el objetivo de demostrar vulnerabilidades de seguridad en sistemas Windows, especialmente Windows 95 y 98.

[3] obtenido de https://learn.microsoft.com/en-us/security-updates/securitybulletins/1998/ms98-010

```
sudo iptables-save > /etc/iptables/rules.v4
```

PASO 5. ESCANEO DEL PUERTO INUSUAL PARA VERIFICAR BLOQUEO. Una vez ejecutado el iptables, volvemos a escanear específicamente el puerto 31337 con el comando.

```
nmap -p 31337 local host
```

Se arroja específicamente el estado del puerto y se visualiza la siguiente tabla

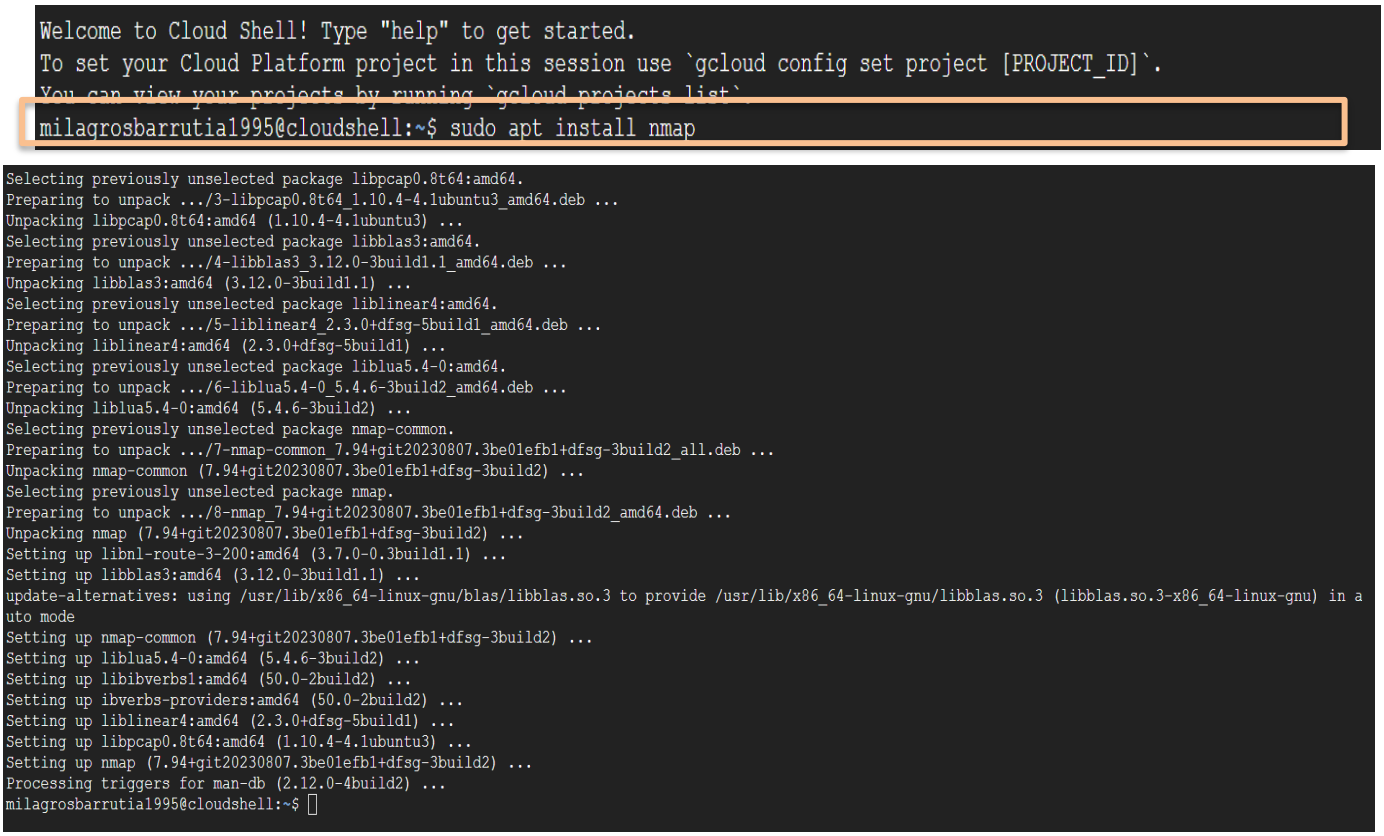
Puerto	State	Service
31337/tcp	Filtered	Elite

Esto significa que el puerto está bloqueado por el firewall que ejecutamos anteriormente.

5. Resultados Obtenidos

PASO 1. INSTALACION DE LA HERRAMIENTA NMAP

Se visualizan los resultados de una instalación exitosa.



PASO 2. EJECUCIÓN DEL COMANDO DE ESCANEO DE PUERTOS.

```
milagrosbarrutia1995@cloudshell:~$ nmap scanme.nmap.org
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-05-31 19:11 UTC
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.073s latency).
Other addresses for scanme.nmap.org (not scanned): 2600:3c01::f03c:91ff:fe18:bb2f
Not shown: 996 closed tcp ports (conn-refused)
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
9929/tcp  open  nping-echo
31337/tcp open  Elite

Nmap done: 1 IP address (1 host up) scanned in 1.30 seconds
milagrosbarrutia1995@cloudshell:~$
```

CONFIGURACIÓN.

```
Not shown: 996 closed tcp ports (conn-refused)
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
9929/tcp  open  nping-echo
31337/tcp open  Elite

Nmap done: 1 IP address (1 host up) scanned in 1.30 seconds
milagrosbarrutia1995@cloudshell:~$ sudo iptables -A INPUT- -p tcp --deport 31337 -j DROP
Bad argument `--deport'
Try `iptables -h' or 'iptables --help' for more information.
milagrosbarrutia1995@cloudshell:~$ sudo iptables -A INPUT -p tcp --dport 31337 -j DROP
milagrosbarrutia1995@cloudshell:~$ sudo iptables-save > /etc/iptables/rules.v4
-bash: /etc/iptables/rules.v4: No such file or directory
milagrosbarrutia1995@cloudshell:~$ sudo iptables-save > /etc/iptables/rules.v4
-bash: /etc/iptables/rules.v4: No such file or directory
```

PASO 5. ESCANEO DE PUERTO INUSUAL PARA VERIFICAR BLOQUEO

```
milagrosbarrutia1995@cloudshell:~$ nmap -p 31337 localhost
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-05-31 19:21 UTC
Nmap scan report for localhost (127.0.0.1)
Host is up (0.00011s latency).
Other addresses for localhost (not scanned): ::1

PORT      STATE    SERVICE
31337/tcp filtered Elite

Nmap done: 1 IP address (1 host up) scanned in 0.24 seconds
```

6. Conclusiones

- La instalación de Nmap en Google Cloud Shell es sencilla y rápida con el comando **sudo apt install nmap**, lo que permite disponer de esta herramienta para el escaneo y análisis de puertos en el sistema operativo Linux.
- El escaneo de Nmap con script **scan .nmap .org** se identificaron varios puertos abiertos y sus servicios, lo que demuestra la efectividad de esta herramienta.
- La detección del puerto **37337/tcp** conocido por estar asociado al hacking, da cuenta de la importancia de analizar los puertos abiertos ya que algunos pueden presentar riesgos de seguridad.

- El uso de iptables para bloquear el puerto identificado verifica ser una medida eficiente para bloquear el tráfico, fortaleciendo la seguridad del sistema e impedir accesos no autorizados.
- El comando **iptables-save** permite guardar temporalmente las pruebas de bloqueo con Nmap, confirmando que el puerto detectado queda filtrado y protegido.

Este ejercicio de práctica permitió comprender la importancia de las herramientas de escaneo y filtrado para identificar y atajar las vulnerabilidades en Linux, dando a conocer la necesidad del monitoreo y configuración para mantener la seguridad.

7. Bibliografía

- https://repository.unipiloto.edu.co/bitstream/handle/20.500.12277/11573/Seguridad_Sistemas_operativos_Linux.pdf?sequence=1
- https://www.redeszone.net/tutoriales/configuracion-puertos/nmap-escanear-puertos-comandos
- https://en.wikipedia.org/wiki/Cult_of_the_Dead_Cow?utm_source=chatgpt.com
- https://learn.microsoft.com/en-us/security-updates/securitybulletins/1998/ms98-010
- .https://www.cbtnuggets.com/common-ports/what-is-port-31337
- https://derouter.es/puertos/puerto-31337-tcp-descubre-como-funciona-back-orifice-la-herramienta-de-administracion-remota/
- https://www.sciencedirect.com/topics/computer-science/back-orifice
- https://tup.sied.utn.edu.ar/pluginfile.php/9656/mod_label/intro/actividad2apuntes.pdf?time=1739999312459
- https://tup.sied.utn.edu.ar/pluginfile.php/9659/mod_label/intro/nmap-pdf.pdf