Seguridad en Sistemas Operativos

- **Alumnos:**
  - **Nahuel Urciuoli Zabala – Nahuel_zabala@live.com**
  - **Martin Rotolo – martinrotolo97@gmail.com**
- **Profesor:**
  - **Osvaldo Falabella**
- **Tutora:**
  - **Ana Valeria Celerier**

- **Fecha de entrega:**
  - **05/06/2025**

Contenido

Introducción

Hoy en día debido a el creciente número de amenazas cibernéticas, es esencial comprender los riesgos y vulnerabilidades que están presentes en el día a día en la Internet por eso se considera que la seguridad en los sistemas operativos es un pilar fundamental en la protección de la información y en la estabilidad de los sistemas.

Conocer los riesgos, como detectarlos y afrontarlos son pasos importantes si se considera que son riesgos a los que uno como programador debe estar preparado.

Cada dispositivo conectado a una red es una puerta de entrada que debemos proteger.
Creo que como programador comprender cómo estas vulnerabilidades impactan en la operación y administración de sistemas nos permitirá anticiparnos a problemas críticos y diseñar soluciones.

Algunos ejemplos de estos problemas pueden ser:

- Pérdida de datos debido a accesos no autorizados
- Interrupción de servicios esenciales.
- Compromiso de la privacidad de usuarios y entidades.
- Robo de datos y falsificación de identidad entre otros

A continuación, se informará sobre algunos tipos de ataque, causas, consecuencias y como tratar de detectarlos antes de tiempo en el mejor caso

Marco Teórico

Existen dos tipos de ataques informáticos

- Ataques activos
- Ataques pasivos

Los Activos producen cambios en la información (modificación de información en un archivo o cuenta bancaria) y en la situación de los recursos del sistema (uso de recursos del sistema) Mientras que los ataques pasivos registran el uso de los recursos y trata de acceder a la información ya sea guardada o transmitida por el sistema (escuchan, acceden a información, pero no la alteran)

Algunos ejemplos de **ataques pasivos** pueden ser:

- **Sniffing**: Atacante que ve el tráfico de datos que pasa por una red, buscando contraseñas, información bancaria, etc., sin que nadie se entere.
- **Análisis de tráfico de red**: Monitorear redes para detectar y analizar fallos, robar contraseñas, interceptar correos electrónicos, espiar conversaciones de chat, etc
- **Acceso a datos**: Una vez accedido a los datos del usuario pueden suplantar su identidad para transacciones bancarias o venderlos a terceros
- **Cross-Site Scripting**: Ejecución de código “Script” en un navegador para acceder por ejemplo a las cookies del mismo

Y ejemplos de **ataques activos**:

- **Malware**: Programa, documento o mensaje susceptible de causar daños en las redes y sistemas informáticos
- **Ataques DoS**: Buscan colapsar determinados equipos o redes informáticos, para impedir que puedan ofrecer sus servicios a clientes y usuarios
- **Modificación de mensajes**: Intercepta un correo electrónico y cambia su contenido antes de que llegue al destinatario

Existen varias maneras de realizar estos ataques y generar una conexión no autorizada algunos conocidos son los “**exploits**” que vendrían a ser agujeros de seguridad, “**backdoors**”, a través de instrucciones no documentadas se toma el control del equipo salteándose controles, “**rootkits**”,

programas que se instalan en un equipo reemplazando a una herramienta o servicio legítimo del sistema operativo y entre otros el **Escáneres de puertos** , que permiten ver que puertos están abiertos y acceder a ellos,

Como mencionamos anteriormente algunas de las **consecuencias de las conexiones no autorizadas** a los sistemas son

- Acceso a información confidencial incluso a ficheros que habían sido “borrados” Los atacantes incluso podrían
- Mail relaying: Envio de correo masivos desde un servidor externo a la organización
- Utilización de la capacidad de procesamiento de los equipos para otros fines
- Creación de nuevas cuentas de usuario con privilegios administrativos
- Consumo del ancho de banda de la red de la organización para otros fines.
- Almacenamiento de contenidos ilegales en otros sistemas
- Modificación o destrucción de archivos y documentos

Por lo general un ataque informático sigue una secuencia de pasos/etapas

1. Descubrimiento y exploración del sistema informático
2. Búsqueda de vulnerabilidades en el sistema (Escáner de puertos, Dos)
3. Explotación de las vulnerabilidades detectadas (Malware, exploits, backdoors)
4. Corrupción o compromiso del sistema o datos (modificación/eliminación/restricción de programas y ficheros del sistema. Creación de cuentas con privilegios Admin y así facilitar posterior acceso)
5. Eliminación de pruebas que puedan revelar el ataque y/o comprometer al atacante (eliminación o modificación de los registros de actividad del equipo por ejemplo logs)

De la misma manera que existen diferentes tipos y modos de ataques también existen diferentes clasificaciones de Intrusos en las redes según sus intereses:

- Hackers: Lo ven como un pasatiempo o reto técnico
- Crackers: Atacan para obtener beneficios o talvez por ideologías buscan hacer daños
- Sniffers: Rastrean mensajes para descifrarlos
- Phreakers: Sabotean redes telefónicas
- Spammers: Envio masivo de mails
- Piratas Informaticos: Pirateo de programas

- Creadores de Virus: Expertos informáticos que distribuyen virus
- Lamers: Responsables de la mayoría de los ataques, son personas sin conociendo técnico que usan programas o conocimiento de terceros para realizar ataques
- Personal Interno/Externo y ex empleados

Existen diferentes motivaciones por las cuales los actores maliciosos actúan de esta manera. El FBI, por ejemplo, los clasifica bajo el acrónimo MICE, que hace referencia a (Money, Ideology, Compromise, Ego).

Por ejemplo, con la información obtenida de un Sniffer a través del sniffing, un atacante motivado por “Money” (dinero) podría obtener acceso a una cuenta bancaria y realizar transacciones no autorizadas. De manera similar, los datos de la víctima podrían ser usados para extorsiones (también por Money).

Otro ejemplo un ”cracker” que, motivado por “Money” o quizás por una “Ideology” (como el activismo), podría restringir o cifrar el acceso a datos importantes de una empresa, o amenazar con hacerlos públicos, a menos que se le entregue una cierta cantidad de dinero (ransomware).

Finalmente, un Spammer, impulsado a menudo por el “Money” o la “Ideology” de la ganancia fácil, podría realizar un envío masivo de correos electrónicos de “phishing”. En estos ataques, se hacen pasar por alguna entidad bancaria o un servicio legítimo para engañar al usuario y que este ingrese sus datos oficiales en una página web no oficial, muy similar a la original. De esta manera, obtienen las credenciales de la víctima para realizar operaciones fraudulentas o incluso restringir el acceso a la cuenta a cambio de una suma monetaria

O simplemente un Hacker que motivado por su “Ego” realiza ataque con el fin de demostrar a si mismo o a una comunidad que era capaz de hacerlos.

¿Y la gran pregunta cómo nos defendemos o detectamos de estos ataques?

Existen varias herramientas y métodos para estos casos, entre ellos:

- Firewall: registra y filtra el trafico de red permitiendo solo el legitimo o el designado por el usuario
- Antivirus: Detecta, previene y elimina software malicioso. Es importante mantener

actualizado el mismo ya que siempre salen nuevas amenazas y con las actualizaciones se las registra en el antivirus

- Monitoreo de actividad: Identificar eventos sospechosos en por ejemplo en “logs” y evitar ataques. Por ejemplo, se podría ver el aumento de trafico de red o verificar que hubo varios intentos fallidos de sesión
- MFA (Multifactor): Combinar algo que el usuario sabe (contraseña) con algo que tiene (biometría o celular)
- Actualización y Parches: Importante para corregir vulnerabilidades y tener un sistema estable
- Cifrar datos importantes por ejemplo con BitLocker
- Mantener una buena gestión de permisos y roles de usuario

Si bien la manera de afectar sistemas o acceder a información siempre evoluciona, también lo hacen los sistemas para defenderse de los mismos, representando esto una constante evolución de los sistemas de seguridad y maneras de protección

Caso Práctico

Por ejemplo, si queremos restringir el acceso de un archivo en Linux (contraseñas.txt)

Usamos el comando chmod de esta manera solo el Admin tiene acceso y solo de lectura

```
nahuel_zabala2016@cloudshell:~$ chmod 400 contraseñas.txt
nahuel_zabala2016@cloudshell:~$
```

Asignar un usuario a un grupo especifico incluyendo sudo(ejemplo testuser a admin)

```
nahuel_zabala2016@cloudshell:~$ sudo usermod -aG adm testuser
```

Ver los grupos de un usuario (agregado testuser a admin)

```
nahuel_zabala2016@cloudshell:~$ groups testuser
testuser : testuser adm
```

Listar Puertos abiertos:

```
nahuel_zabala2016@cloudshell:~$ sudo netstat -tulnp
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 0.0.0.0:900             0.0.0.0:*               LISTEN      -
```

Permitir trafico SSH desde IP especifica:

```
nahuel_zabala2016@cloudshell:~$ sudo iptables -A INPUT -p tcp --dport 22 -s 192.168.1.100 -j ACCEPT
```

Bloquear tráfico de conexiones INPUT (pasan de estado INPUT a drop) excepto de conexiones establecidas:

```
nahuel_zabala2016@cloudshell:~$ sudo iptables -P INPUT DROP
nahuel_zabala2016@cloudshell:~$ sudo iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
```

Estas son algunas medidas de seguridad en LINUX otras pueden ser :

- Luks para cifrado con clave
- Snort para identificar patrones anómalos en “logs”
- Google autentificador o PAM para MFA
- Manetener Windows update activado

Caso Práctico 2

Análisis de Seguridad en Host Público con Nmap y Hardening

La empresa ficticia "WebSecure S.A." solicita una auditoría básica de seguridad sobre uno de sus servidores públicos. Para ello, se realiza un escaneo con la herramienta Nmap desde un entorno controlado (Google Cloud Shell), utilizando como objetivo el dominio scanme.nmap.org, que permite este tipo de pruebas de forma segura y legal.

Se busca identificar los servicios expuestos, las versiones activas, y proponer una estrategia de endurecimiento (hardening) que minimice riesgos de intrusión.

Pasos ejecutados:

1. **Escaneo rápido de puertos:**

nmap -sT scanme.nmap.org

```
tinchorotolo@cloudshell:~$ nmap -sT scanme.nmap.org
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-06-02 21:59 UTC
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.14s 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 2.43 seconds
```

Se utilizó -sT en lugar de -sS debido a las restricciones de permisos en Cloud Shell. El escaneo TCP Connect detectó puertos abiertos accesibles sin necesidad de privilegios elevados. Se identificaron servicios como HTTP y SSH expuestos públicamente.

2. **Detección de versiones de servicios:**

nmap -sV scanme.nmap.org

```
tinchorotolo@cloudshell:~$ nmap -sV scanme.nmap.org
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-06-02 22:00 UTC
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.14s 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    VERSION
22/tcp    open  ssh        OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.13 (Ubuntu Linux; protocol 2.0)
80/tcp    open  http       Apache httpd 2.4.7 ((Ubuntu))
9929/tcp  open  nping-echo Nping echo
31337/tcp open  tcpwrapped
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 15.21 seconds
```

Este paso permitió identificar las versiones de los servicios que operan en los puertos abiertos. Esto es crucial, ya que muchas vulnerabilidades están asociadas a versiones específicas. La versión de Apache y SSH obtenida puede cotejarse con bases de datos de vulnerabilidades (como CVE).

3. **Escaneo completo con información de sistema operativo y scripts:**

nmap -A scanme.nmap.org

```
tinchorotolo@cloudshell:~$ nmap -A scanme.nmap.org
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-06-02 22:02 UTC
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.14s 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    VERSION
22/tcp    open  ssh        OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.13 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   1024 ac:00:a0:1a:82:ff:cc:55:99:dc:67:2b:34:97:6b:75 (DSA)
|   2048 20:3d:2d:44:62:2a:b0:5a:9d:b5:b3:05:14:c2:a6:b2 (RSA)
|   256 96:02:bb:5e:57:54:1c:4e:45:2f:56:4c:4a:24:b2:57 (ECDSA)
|_  256 33:fa:91:0f:e0:e1:7b:1f:6d:05:a2:b0:f1:54:41:56 (ED25519)
80/tcp    open  http       Apache httpd 2.4.7 ((Ubuntu))
|_http-server-header: Apache/2.4.7 (Ubuntu)
|_http-title: Go ahead and ScanMe!
|_http-favicon: Nmap Project
9929/tcp  open  nping-echo Nping echo
31337/tcp open  tcpwrapped
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 17.07 seconds
```

Esta opción permite un reconocimiento más completo: realiza detección de SO, traceroute, escaneo de puertos y servicios, e intenta identificar software y configuración del servidor. Aunque más lento, es ideal para evaluaciones iniciales de exposición.

4. Detección de vulnerabilidades comunes:

nmap --script vuln scanme.nmap.org

```
tinchorotolo@cloudshell:~$ nmap --script vuln scanme.nmap.org
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-06-02 22:04 UTC
Stats: 0:04:11 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 99.21% done; ETC: 22:08 (0:00:02 remaining)
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.14s 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
|_http-dombased-xss: Couldn't find any DOM based XSS.
| http-slowloris-check:
|   VULNERABLE:
|   Slowloris DOS attack
|     State: LIKELY VULNERABLE
|     IDs:  CVE:CVE-2007-6750
|       Slowloris tries to keep many connections to the target web server open and hold
|       them open as long as possible.  It accomplishes this by opening connections to
|       the target web server and sending a partial request. By doing so, it starves
|       the http server's resources causing Denial Of Service.
|
|     Disclosure date: 2009-09-17
|     References:
|       https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2007-6750
|_      http://ha.ckers.org/slowloris/
| http-csrf:
| Spidering limited to: maxdepth=3; maxpagecount=20; withinhost=scanme.nmap.org
|   Found the following possible CSRF vulnerabilities:
|
|     Path: http://scanme.nmap.org:80/
|     Form id: nst-head-search
|     Form action: /search/
|
|     Path: http://scanme.nmap.org:80/
|     Form id: nst-foot-search
|_    Form action: /search/
|_http-stored-xss: Couldn't find any stored XSS vulnerabilities.
| http-enum:
|_  /images/: Potentially interesting directory w/ listing on 'apache/2.4.7 (ubuntu)'
9929/tcp  open  nping-echo
31337/tcp open  Elite

Nmap done: 1 IP address (1 host up) scanned in 326.32 seconds
```

El análisis detectó tanto servicios seguros como configuraciones potencialmente vulnerables. Aunque algunos resultados indicaron que no se encontraron fallas, otros no pudieron determinarlo con certeza, lo cual es útil para justificar medidas preventivas adicionales como actualizaciones o configuraciones de refuerzo.

Los resultados revelaron varios puertos abiertos, incluyendo HTTP y SSH, así como servicios con versiones antiguas. Se recomienda aplicar medidas de hardening como:

- Actualizar software de servicios.
- Deshabilitar servicios innecesarios.
- Configurar firewall para limitar el acceso solo a IPs permitidas
- Aplicar autenticación por clave SSH y MFA.

Metodología Utilizada

Se partió de una revisión teórica basada en los documentos de la materia, incluyendo el uso de Nmap, clasificación de vulnerabilidades, fases de ataque, y medidas de mitigación.

Luego, se utilizó el entorno Google Cloud Shell para ejecutar comandos reales sobre un objetivo autorizado (scanme.nmap.org). Se utilizaron opciones básicas y avanzadas de Nmap para detectar puertos, servicios, versiones y posibles vulnerabilidades, sin comprometer el entorno ni el objetivo.

Herramientas:

- Google Cloud Shell
- Nmap (v7.80)
- Terminal Bash

Las pruebas fueron validadas con capturas de pantalla y análisis interpretativo de los resultados, proponiendo mejoras concretas con base en buenas prácticas.

Resultados Obtenidos

Se logró identificar correctamente:

- Servicios en puertos TCP comunes (22, 80).
- Versiones antiguas de servicios HTTP.
- Ausencia de políticas de hardening en cabeceras HTTP.

Nmap demostró ser eficaz como herramienta de diagnóstico en tareas de ciberseguridad.

Dificultades:

- Limitaciones del entorno Cloud Shell para escaneo de redes privadas.
- Tiempo de espera elevado para escaneos completos.

El trabajo permitió aplicar teoría a un caso práctico realista, generando una propuesta de mejora basada en los hallazgos.

Conclusiones

La seguridad en los sistemas operativos es una responsabilidad fundamental que requiere comprender los riesgos, aplicar buenas prácticas y mantenerse actualizado. Para proteger los sistemas informáticos, es clave implementar medidas como el uso de contraseñas seguras, mantener el software actualizado, evitar el uso de dispositivos externos sin verificación y realizar descargas solo desde sitios oficiales. La combinación de herramientas como firewalls, antivirus y sistemas de monitoreo permite prevenir accesos no autorizados, detectar amenazas y reducir vulnerabilidades de forma efectiva.

Por lo tanto, una configuración adecuada que incluya firewalls bien definidos, permisos estrictos, autenticación reforzada y autenticación en dos pasos son practicas muy importantes hoy en día, donde cada día hay más gente (ya sea con buenas o malas intenciones) tiene acceso a internet. Si a esto le sumamos un enfoque preventivo, basado en auditorías, pruebas de penetración y gestión de parches, fortaleceríamos aún más la resiliencia del sistema frente a amenazas emergentes.

Bibliografía:

- Documentación oficial de Nmap: https://nmap.org/book/ (consultado el 01/06/2025)
- Seguridad avanzada y mantenimiento preventivo. UTN – Programación a Distancia (2024).
- Principales vulnerabilidades y ataques. UTN – Facultad Regional Mendoza (2020).
- Herramientas básicas de seguridad. UTN – Programación a Distancia.
- Introducción a la seguridad en sistemas operativos. UTN – Programación a Distancia.