Seguridad en Sistemas Operativos

* Alumnos:
  + Nahuel Urciuoli Zabala – [Nahuel\_zabala@live.com](mailto:Nahuel_zabala@live.com)
  + Martin Rotolo – martinrotolo97@gmail.com
* Profesor:
  + Osvaldo Falabella
* Tutora:
  + Ana Valeria Celerier
* Fecha de entrega:
  + - * + 05/06/2025

**Contenido**

[Introducción 2](#_Toc199254658)

[Marco Teórico 2](#_Toc199254659)

[Caso Práctico 2](#_Toc199254660)

[Metodología Utilizada 2](#_Toc199254661)

[Resultados Obtenidos 2](#_Toc199254662)

[Conclusiones 2](#_Toc199254663)

[Bibliografía 2](#_Toc199254664)

[Anexos 2](#_Toc199254665)

# 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

**📘 Este apartado contiene la fundamentación conceptual del tema tratado. Debe incluir**

**definiciones, clasificaciones, jerarquías, estructuras y sintaxis si corresponde.  Incluir gráficos, esquemas o tablas si son útiles para la comprensión.  Mencionar autores o documentación oficial si se toman definiciones o explicaciones.**

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

**Aquí se debe presentar un problema o situación concreta que haya sido desarrollada o**

**simulada para aplicar el contenido del trabajo.**

**Incluye:  Breve descripción del problema a resolver.  Capturas de pantalla si corresponde.  Validación del funcionamiento.**

**raudes, engaños y extorsiones.**

**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



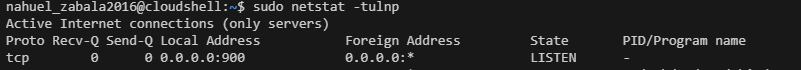
Asignar un usuario a un grupo especifico incluyendo sudo(ejemplo testuser a admin)



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

# 

Listar Puertos abiertos:



Permitir trafico SSH desde IP especifica:



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





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**

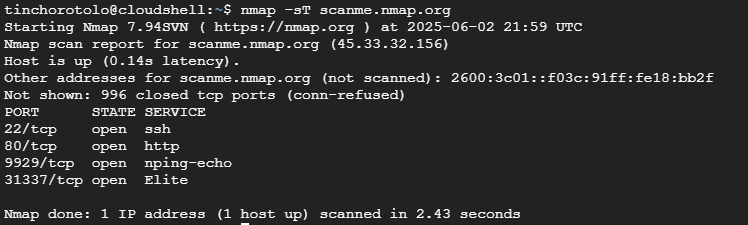
La empresa ficticia "WebSecure S.A." solicita una auditoría básica de seguridad sobre uno de sus servidores públicos. Para ello, se realiza un escaneo con la herramienta Nmap desde un entorno controlado (Google Cloud Shell), utilizando como objetivo el dominio scanme.nmap.org, que permite este tipo de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

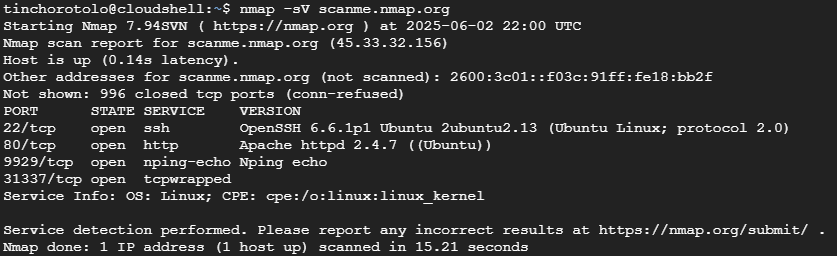
nmap -sT scanme.nmap.org



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.

1. Detección de versiones de servicios:

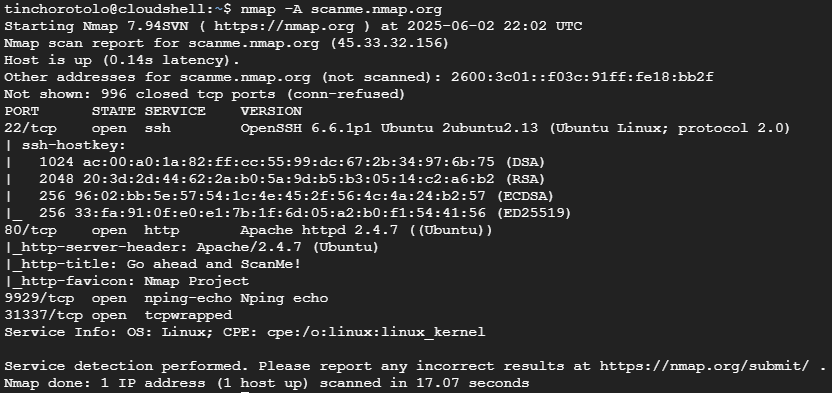
nmap -sV scanme.nmap.org



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).

1. Escaneo completo con información de sistema operativo y scripts:

nmap -A scanme.nmap.org



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.

1. Detección de vulnerabilidades comunes:

nmap --script vuln scanme.nmap.org



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

Describe los pasos seguidos durante el desarrollo del trabajo.

Se pueden incluir:  Investigación previa (fuentes utilizadas).  Etapas de diseño y prueba del código.  Herramientas y recursos utilizados (IDE, librerías, control de versiones, etc.).  Trabajo colaborativo (reparto de tareas en el grupo de trabajo).

**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

🎯 Reflexión final del grupo de trabajo. Aquí se recomienda incluir:  Qué se aprendió al hacer el trabajo.  Posibles mejoras o extensiones futuras.  Dificultades que surgieron si las hubo y cómo se resolvieron.

La seguridad en sistemas operativos es una responsabilidad clave. La comprensión de los riesgos,

junto con la aplicación de estrategias efectivas, permite mantener un entorno seguro y confiable

para la operación de los sistemas informáticos. Implementar buenas prácticas, utilizar

herramientas adecuadas y estar en constante actualización son medidas esenciales para

minimizar amenazas y garantizar la estabilidad de los sistemas

Para concluir el hecho de llevar a cabo un ataque informático, indica que los intrusos deben

disponer de los medios técnicos, los conocimientos y las herramientas adecuadas, deben contar con

una determinada motivación o finalidad, y se tiene que dar además una determinada oportunidad

que facilite el desarrollo del ataque (como podría ser el caso de un fallo en la seguridad del sistema

informático elegido). Estos tres factores constituyen lo que podríamos denominar como el

“Triángulo de la Intrusión”, concepto que se presenta de forma gráfica en la siguiente figura:

No repetir contraseñas e incluir en ellas letras mayúsculas, minúsculas, números y

caracteres especiales con más de 8 caracteres.

2. Tener actualizado el antivirus y escanear el equipo con frecuencia para revisar que

todo está en orden.

3. No ignorar las actualizaciones del sistema operativo y aplicaciones que ayudan a

mantener un sistema con las menores vulnerabilidades.

4. No dejar los dispositivos desbloqueados y con la sesión abierta si no se está

trabajando en ellos.

5. Desconfiar de los dispositivos externos.

6. Habilitar el spam, no abrir ningún mail de origen desconocido ni enlaces de correos

no deseados.

7. Navegación segura, desconfía de la publicidad engañosa que ofrecen premios a

cambio de click, comprobar que en la barra del navegador el protocolo https.

8. Las descargas de programas sólo realizarlas a través de páginas web oficiales.

Conclusión

La combinación de firewalls, antivirus y monitoreo de actividad es esencial para mantener la

seguridad de sistemas y redes. Mientras que el firewall actúa como una barrera de protección

contra accesos no autorizados, el antivirus evita infecciones por malware y el monitoreo de

actividad permite la detección temprana de amenazas. Implementar estas soluciones de manera

conjunta es clave para una estrategia de ciberseguridad efectiva y proactiva.

Una adecuada configuración de seguridad en Linux es un pilar fundamental para la protección

de los sistemas informáticos. La combinación de firewalls bien configurados, permisos estrictos

en archivos y directorios, y autenticación robusta contribuye significativamente a reducir las

superficies de ataque.

El uso de iptables o firewalld permite establecer reglas de tráfico que impiden accesos no

autorizados. La correcta gestión de usuarios, asignando permisos con chmod y chown, junto con

políticas de roles mediante sudo, refuerza el control sobre los privilegios del sistema.

Finalmente, la autenticación de dos factores y la restricción del acceso SSH a través de claves

públicas garantizan que solo usuarios legítimos puedan interactuar con los recursos críticos del

sistema.

Implementar estas medidas de manera conjunta ayuda a prevenir incidentes de seguridad y

mejora la resiliencia de los sistemas frente a amenazas cibernéticas.

Una adecuada configuración de seguridad en Linux es un pilar fundamental para la protección

de los sistemas informáticos. Implementar firewalls bien configurados, permisos estrictos,

autenticación robusta y optimización de recursos contribuye significativamente a reducir las

superficies de ataque y mejorar la estabilidad del sistema.

La implementación de herramientas y configuraciones adecuadas es crucial para mantener un

sistema seguro y estable. A medida que las amenazas evolucionan, es vital actualizar y mejorar

constantemente las medidas de seguridad

La seguridad avanzada y el mantenimiento preventivo son dos pilares fundamentales en la

protección de sistemas operativos. Implementar medidas como MFA, cifrado de datos y

sistemas de detección, combinados con una gestión proactiva de parches y auditorías

periódicas, permite crear un entorno resiliente frente a amenazas. Además, realizar pruebas de

penetración y analizar de forma continua los registros del sistema garantiza que cualquier

vulnerabilidad se detecte y corrija antes de que pueda ser explotada.

Este enfoque integral no solo mejora la seguridad inmediata, sino que también establece una

base sólida para el mantenimiento a largo plazo, adaptándose a nuevos desafíos y amenazas en

el entorno digital.

**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.

📥 Recomendaciones para la presentación  Formato del archivo: .pdf 

Tipografía: Arial o Calibri, tamaño 11 o 12

 Interlineado: 1,5 

Márgenes estándar (2.5 cm)

 Portada opcional con el logo de la institución (si se requiere)

 Entrega digital mediante plataforma institucional o por correo