Proyecto final de Ciberseguridad

Análise forense y endurecimiento de un servidor Linux ante un incidente de seguridad | Felipe Torrado Dias

Introducción

Este trabajo analiza un incidente de seguridad detectado en un servidor Linux Debian, configurado como entorno de laboratorio.

El objetivo fue comprender **cómo se produjo el acceso**, evaluar si existió un compromiso real del sistema y aplicar medidas de endurecimiento para reducir la superficie de ataque.

Fase 1 — Reconocimiento y evidencias

¿Cómo se determinó el vector de entrada y qué se buscó en el sistema?

- Nmap para enumeración de puertos/servicios
- journalctl para revisión de logs

ANÁLISIS FORENSE DEL ACCESO SSH

Al filtrar los registros del sistema se identificó un evento clave:

journalctl --no-pager | grep -Ei "Accepted password"

Se confirmó un **inicio de sesión exitoso del usuario root**, utilizando **autenticación por contraseña**.

Esto evidenció un problema de **configuración insegura**, no una vulnerabilidad técnica avanzada.

BÚSQUEDA DE PERSISTENCIA

Se revisaron accesos fallidos, historial de sesiones y actividad del sistema

Comprobación de backdoors o persistencia

Se analizaron posibles mecanismos de persistencia revisando: tareas programadas (cron), servicios activos (systemd), usuarios y archivos sensibles

No se identificaron patrones de **fuerza bruta ni múltiples accesos** posteriores.

El acceso **fue limitado y puntual**, sin indicios de uso prolongado ni movimiento lateral.

Conclusión: **no se observaron** evidencias concluyentes de **persistencia o payload**.

MEDIDAS DE ENDURECIMIENTO APLICADAS

Corrección de configuraciones inseguras

Bloqueo del vector de entrada

Como medida principal se modificó la configuración del servicio SSH, deshabilitando el acceso remoto del usuario root:

PermitRootLogin no

Además, se reforzaron permisos, se actualizaron paquetes y se eliminaron configuraciones innecesarias, reduciendo el riesgo de futuros accesos no autorizados.

Servicio FTP

Descubrimiento: escaneo Nmap (nmap -A -p- localhost) identificó FTP (21/tcp) junto a otros servicios.

Riesgo: autenticación anónima es mala práctica; aumenta superficie de ataque y puede exponer datos si aparecen ficheros.

Validación: acceso anónimo sin permisos de escritura y directorio vacío (riesgo potencial, no explotación efectiva).

Mitigación: detener, deshabilitar y enmascarar el servicio → el puerto 21 deja de estar en escucha.

Servicio Apache

En la parte del servicio Apache, no se identificó una explotación directa, pero sí una **debilidad de configuración.**
El servidor web **exponía información innecesaria**, como el tipo y la versión del software, además de mostrar la página por defecto.

Esto supone **un riesgo** porque **facilita la fase de reconocimiento de un atacante** y le ayuda a seleccionar exploits específicos.

Como medida de endurecimiento, se **modificó la configuración de seguridad** de Apache para **minimizar la información expuesta**.
Se ocultó la versión del servidor, se eliminaron las firmas en páginas de error y se **deshabilitó el método TRACE**.

Hardening adicional: permisos y base de datos

Se detectaron **permisos excesivos en el directorio web** (/var/www/html), lo que suponía un riesgo de lectura o modificación no autorizada de archivos del sitio. Como medida correctiva, se aplicó el **principio de mínimo privilegio**, asignando **permisos restrictivos** y asegurando que solo el usuario del servicio web (www-data) pudiera acceder a los recursos necesarios.

Adicionalmente, se realizó una **auditoría de las cuentas** de MariaDB, identificando una **cuenta genérica** con **privilegios administrativos completos**. Tras verificar que la aplicación utilizaba un usuario específico (@user), se eliminó la cuenta innecesaria, reduciendo el riesgo de abuso de credenciales y accesos indebidos.

Respuesta a incidentes y SGSI

Para **gestionar el incidente** se aplicó un plan de respuesta basado en **NIST** SP 800-61, permitiendo **identificar** el problema, **contener** el acceso, **eliminar** configuraciones inseguras, **recuperar** el sistema y **extraer lecciones** aprendidas.

Como complemento, se planteó un Sistema de Gestión de Seguridad de la Información (SGSI) según **ISO 27001**, definiendo el **alcance** del sistema, **evaluando riesgos** y aplicando **controles** de seguridad de forma continua.

Entre las recomendaciones finales destacan el uso de **autenticación por claves en SSH**, la **restricción de accesos por IP** cuando sea posible, la **monitorización periódica de logs** y la **revisión continua de configuraciones** para evitar valores por defecto inseguros.

CONCLUSIONES

El análisis demostró que una **configuración insegura** es suficiente para **comprometer un sistema**, **incluso sin exploits complejos**.

La **revisión periódica, el principio de mínimo privilegio y el endurecimiento de servicios** son claves para una postura de seguridad sólida.

Gracias

Contacto:

Felipe Torrdo Dias

professorfelipemancebo@gmail.com

+34 600077482

Barcelona