INFORME DE INCIDENTE DE SEGURIDAD Y ANÁLISIS FORENSE

Respuesta ante Incidentes - Fase de Contención
Hospital General de Madrid (ficticio)

Activo Afectado: Servidor Crítico (192.168.122.10)

Tipo de Incidente: Compromiso de Sistema y Borrado de Logs

Analista Forense: Arturo Martín-Vegue

Fecha del Informe: 6 de Enero de 2026

ESTADO: TRIAJE FINALIZADO / CONFIDENCIAL

Índice

1. Introducción y Contexto del Incidente

El presente informe técnico documenta las actividades de análisis forense, contención y recuperación realizadas sobre un activo crítico perteneciente a la infraestructura tecnológica del **Hospital General de Madrid (ficticia)**.

Este servidor, identificado como parte del alcance del **Sistema de Gestión de Seguridad de la Información (SGSI)** de la organización, ha sufrido un compromiso de seguridad que pone en riesgo la confidencialidad de los datos y la disponibilidad de servicios administrativos.

En consonancia con las políticas de seguridad del Hospital y la normativa vigente, el objetivo de esta intervención ha sido:

1. **Identificar el vector de intrusión:** Determinar cómo el atacante vulneró el perímetro de seguridad del centro hospitalario.
2. **Contención y Erradicación:** Eliminar cualquier acceso no autorizado o persistencia para garantizar la integridad de la red sanitaria.
3. **Restauración Segura:** Recuperar la operatividad del activo aplicando medidas de endurecimiento (*hardening*) que cumplan con los estándares de seguridad definidos en el SGSI.

A continuación, se detallan las evidencias técnicas recolectadas, las acciones correctivas y el estado final del sistema.

2. Identificación y Alcance

2.1 Metodología de Acceso y Autenticación

Para dar comienzo a la fase de identificación, se procedió al acceso físico/consola del activo. Ante la ausencia de credenciales proporcionadas por la administración del Hospital, se aplicó una metodología de **auditoria de caja gris**, realizando pruebas de acceso mediante diccionarios de credenciales comunes.

Se logró el acceso exitoso a la consola del sistema utilizando el usuario `debian` y la contraseña `123456`. Este hecho se documenta como el primer hallazgo crítico del análisis, evidenciando una política de contraseñas de alta predictibilidad. ## 2.2 Ficha Técnica del Activo

Una vez obtenido el acceso al sistema, se procedió a la identificación técnica del sistema operativo y direccionamiento IP mediante la ejecución de comandos nativos:

- **Comando ejecutado:** `ip a && uname -a`
- **Dirección IP Identificada:** `192.168.122.10` (Red Interna Hospitalaria)
- **Kernel:** Linux 6.1.0-25-amd64

Evidencia de Identificación:

```
debian@debian:~$ ip a && uname -a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: enp1s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 52:54:00:c7:b1:bc brd ff:ff:ff:ff:ff:ff
    inet 192.168.122.10/24 brd 192.168.122.255 scope global dynamic noprefixroute enp1s0
       valid_lft 2126sec preferred_lft 2126sec
    inet6 fe80::5054:ff:fec7:b1bc/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
Linux debian 6.1.0-25-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.106-3 (2024-08-26) x86_64 GNU/Linux
```

2.3 Análisis de Superficie de Ataque (Puertos)

Se realizó una enumeración de puertos en escucha para identificar servicios expuestos.

- **Comando ejecutado:** `netstat -tulpen`

Tabla de Servicios Críticos Detectados:

Puerto	Protocolo	Servicio	Estado	Dirección de Escucha	Observaciones
21	TCP	vsftpd	LISTEN	`:::*`	**CRÍTICO.** FTP Expuesto.
80	TCP	apache2	LISTEN	`:::*`	**CRÍTICO.** Web Pública.
3306	TCP	mariadbd	LISTEN	`127.0.0.1`	**SEGURO.** Solo local.

Evidencia de Puertos:

```
debian@debian:~$ netstat -tulpen
(Not all processes could be identified, non-owned process info
 will not be shown, you would have to be root to see it all.)
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       User       Inode      PID/Program name
tcp        0      0 127.0.0.1:3306          0.0.0.0:*               LISTEN      111        16358      -
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      0          16879      -
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN      0          18466      -
tcp6       0      0 ::1:631                 :::*                    LISTEN      0          18465      -
tcp6       0      0 :::22                   :::*                    LISTEN      0          16881      -
tcp6       0      0 :::21                   :::*                    LISTEN      0          18218      -
tcp6       0      0 :::80                   :::*                    LISTEN      0          15351      -
udp        0      0 0.0.0.0:5353            0.0.0.0:*                           104        18084      -
udp        0      0 0.0.0.0:50996           0.0.0.0:*                           104        18086      -
udp6       0      0 :::5353                 :::*                                104        18085      -
udp6       0      0 :::53030                :::*                                104        18087      -
```

Interpretación de la Superficie de Ataque: El análisis de puertos evidencia una **exposición crítica** en el perímetro del servidor. Se observa que tanto el servicio de transferencia de ficheros (**FTP**, puerto 21) como el servidor web (**HTTP**, puerto 80) están configurados para aceptar conexiones desde cualquier origen (`:::*` o `0.0.0.0`), careciendo de restricciones de red.

Por el contrario, el servicio de base de datos (**MariaDB**, puerto 3306) se encuentra correctamente configurado, escuchando únicamente en la interfaz de bucle local (`127.0.0.1`), lo que mitiga el riesgo de ataques directos contra el motor de base de datos desde el exterior. # 3. Análisis Forense

3.1 Detección de Técnicas Anti-Forense

Durante la revisión preliminar de los registros del sistema para identificar accesos no autorizados, se detectó una anomalía crítica en la integridad de los logs.

- **Comando ejecutado:** `ls -la /var/log/`
- **Hallazgo:** Se confirma la **eliminación intencionada** de los archivos `/var/log/auth.log` y `/var/log/syslog`.
- **Interpretación** Esta acción evidencia un intento de ocultación de huellas por parte del atacante tras obtener privilegios elevados.

Evidencia de Eliminación de Logs:

```
debian@debian:~$ ls -la /var/log/
total 1040
drwxr-xr-x  11 root              root              4096 Jan  5 14:29 .
drwxr-xr-x  12 root              root              4096 Sep 30  2024 ..
-rw-r--r--   1 root              root                 0 Jan  5 14:29 alternatives.log
-rw-r--r--   1 root              root             48068 Sep 30  2024 alternatives.log.1
drwxr-x---   2 root              adm               4096 Sep 30  2024 apache2
drwxr-xr-x   2 root              root              4096 Jan  5 14:29 apt
-rw-------   1 root              root                 0 Jan  5 14:29 boot.log
-rw-------   1 root              root             80083 Jan  5 14:29 boot.log.1
-rw-rw----   1 root              utmp                 0 Jan  5 14:29 btmp
-rw-rw----   1 root              utmp              2688 Oct  8  2024 btmp.1
drwxr-xr-x   2 root              root              4096 Jan  5 14:29 cups
-rw-r--r--   1 root              root                 0 Jan  5 14:29 dpkg.log
-rw-r--r--   1 root              root            765626 Oct  8  2024 dpkg.log.1
-rw-r--r--   1 root              root                 0 Jul 31  2024 faillog
-rw-r--r--   1 root              root              5602 Sep 30  2024 fontconfig.log
drwxr-xr-x   3 root              root              4096 Jul 31  2024 installer
drwxr-sr-x+  3 root              systemd-journal   4096 Jul 31  2024 journal
-rw-rw-r--   1 root              utmp                 0 Jul 31  2024 lastlog
drwx--x--x   2 root              root              4096 Jan  5 14:29 lightdm
drwx------   2 root              root              4096 Jul 31  2024 private
lrwxrwxrwx   1 root              root                39 Jul 31  2024 README -> ../../usr/share/doc/systemd/README.logs
drwxr-xr-x   3 root              root              4096 Sep 30  2024 runit
drwx------   2 speech-dispatcher root              4096 Nov 25  2022 speech-dispatcher
-rw-rw-r--   1 root              utmp             23808 Jan  5 16:17 wtmp
-rw-r--r--   1 root              root             44050 Jan  5 16:18 Xorg.0.log
-rw-r--r--   1 root              root             26934 Oct  8  2024 Xorg.0.log.old
```

3.2 Análisis de Logs Web (Apache)

Se procedió a inspeccionar el archivo `access.log` en busca de direcciones IP externas o vectores de ataque web (inyecciones SQL, XSS, subida de shells).

- **Comando ejecutado:** `sudo grep -v "127.0.0.1" /var/log/apache2/access.log`
- **Hallazgo:** El registro **carece de direcciones IP externas**. Se ha verificado que la totalidad del tráfico registrado corresponde exclusivamente a procesos internos del propio servidor (`localhost` en IPv4 e IPv6).
- **Conclusión Forense:** La ausencia total de peticiones externas en un servidor público comprometido confirma que el atacante realizó un **borrado selectivo** o purgado de logs tras la intrusión para eliminar cualquier evidencia de su dirección IP de origen.

Evidencias de Direcciones IP:

```
debian@debian:~$ sudo grep -v "127.0.0.1" /var/log/apache2/access.log
[sudo] password for debian:
::1 - - [30/Sep/2024:12:23:51 -0400] "OPTIONS * HTTP/1.0" 200 126 "-" "Apache/2.4.62 (Debian) (internal dummy connection)"
::1 - - [30/Sep/2024:12:23:52 -0400] "OPTIONS * HTTP/1.0" 200 126 "-" "Apache/2.4.62 (Debian) (internal dummy connection)"
::1 - - [30/Sep/2024:12:23:53 -0400] "OPTIONS * HTTP/1.0" 200 126 "-" "Apache/2.4.62 (Debian) (internal dummy connection)"
::1 - - [30/Sep/2024:12:23:57 -0400] "OPTIONS * HTTP/1.0" 200 126 "-" "Apache/2.4.62 (Debian) (internal dummy connection)"
::1 - - [08/Oct/2024:16:49:52 -0400] "OPTIONS * HTTP/1.0" 200 126 "-" "Apache/2.4.62 (Debian) (internal dummy connection)"
::1 - - [08/Oct/2024:16:49:53 -0400] "OPTIONS * HTTP/1.0" 200 126 "-" "Apache/2.4.62 (Debian) (internal dummy connection)"
::1 - - [08/Oct/2024:16:49:54 -0400] "OPTIONS * HTTP/1.0" 200 126 "-" "Apache/2.4.62 (Debian) (internal dummy connection)"
```

3.3 Auditoría de Permisos Web

Se realizó una inspección de los permisos en el directorio raíz del servidor web (`/var/www/html`) para verificar la integridad de los ficheros de configuración.

- **Comando ejecutado:** `ls -la /var/www/html/`
- **Hallazgo CRÍTICO:** Se detectó una configuración de permisos insegura (**777** o `-rwxrwxrwx`) aplicada de forma recursiva a todo el directorio web.
- **Impacto:** El archivo sensible `wp-config.php`, que contiene las credenciales de la base de datos, es legible y modificable por cualquier usuario del sistema. Esto incumple el principio de mínimo privilegio y expone la base de datos del Hospital a un compromiso total.

Evidencia de Permisos Inseguros:

```
debian@debian:~$ ls -la /var/www/html/
total 256
drwxrwxrwx  5 www-data www-data  4096 Oct  8  2024 .
drwxr-xr-x  3 root     root      4096 Sep 30  2024 ..
-rwxrwxrwx  1 www-data www-data   523 Sep 30  2024 .htaccess
-rwxrwxrwx  1 www-data www-data 10701 Sep 30  2024 index.html
-rwxrwxrwx  1 www-data www-data   405 Feb  6  2020 index.php
-rwxrwxrwx  1 www-data www-data 19915 Dec 31  2023 license.txt
-rwxrwxrwx  1 www-data www-data  7409 Jun 18  2024 readme.html
-rwxrwxrwx  1 www-data www-data  7387 Feb 13  2024 wp-activate.php
drwxrwxrwx  9 www-data www-data  4096 Sep 10  2024 wp-admin
-rwxrwxrwx  1 www-data www-data   351 Feb  6  2020 wp-blog-header.php
-rwxrwxrwx  1 www-data www-data  2323 Jun 14  2023 wp-comments-post.php
-rwxrwxrwx  1 www-data www-data  3017 Sep 30  2024 wp-config.php
drwxrwxrwx  5 www-data www-data  4096 Oct  8  2024 wp-content
-rwxrwxrwx  1 www-data www-data  5638 May 30  2023 wp-cron.php
drwxrwxrwx 30 www-data www-data 12288 Sep 10  2024 wp-includes
-rwxrwxrwx  1 www-data www-data  2502 Nov 26  2022 wp-links-opml.php
-rwxrwxrwx  1 www-data www-data  3937 Mar 11  2024 wp-load.php
-rwxrwxrwx  1 www-data www-data 51238 May 28  2024 wp-login.php
-rwxrwxrwx  1 www-data www-data  8525 Sep 16  2023 wp-mail.php
-rwxrwxrwx  1 www-data www-data 28774 Jul  9  2024 wp-settings.php
-rwxrwxrwx  1 www-data www-data 34385 Jun 19  2023 wp-signup.php
-rwxrwxrwx  1 www-data www-data  4885 Jun 22  2023 wp-trackback.php
-rwxrwxrwx  1 www-data www-data  3246 Mar  2  2024 xmlrpc.php
```

3.4 Exposición de Credenciales Críticas

Tras identificar los permisos inseguros, se auditó el archivo de configuración `wp-config.php` para evaluar la robustez de las credenciales de conexión a la base de datos.

- **Comando ejecutado:** `cat /var/www/html/wp-config.php`
- **Hallazgo CRÍTICO:** Se identificó el uso de credenciales por defecto/débiles (`DB_PASSWORD: '123456'`) almacenadas en texto claro.

- **Impacto en el Hospital:** Esta contraseña trivial permite a cualquier atacante que haya leído el archivo (gracias a los permisos 777) conectarse a la base de datos, exfiltrar historias clínicas de pacientes (Confidencialidad) o borrarlas (Disponibilidad e Integridad).

Evidencia de Credenciales Débiles:

```
// ** Database settings - You can get this info from your web host ** //
/** The name of the database for WordPress */
define( 'DB_NAME', 'wordpress' );

/** Database username */
define( 'DB_USER', 'wordpressuser' );

/** Database password */
define( 'DB_PASSWORD', '123456' );
```

3.5 Auditoría de Acceso Remoto (SSH)

Finalmente, se revisó la configuración del servicio SSH para verificar las políticas de acceso administrativo.

- **Comando ejecutado:** `grep "PermitRootLogin" /etc/ssh/sshd_config`
- **Hallazgo:** La directiva `PermitRootLogin` está configurada en **yes**.
- **Impacto:** Permite a un atacante realizar ataques de fuerza bruta directamente contra la cuenta `root`. Si se compromete esta contraseña, el atacante obtiene control total inmediato del servidor sin necesidad de escalar privilegios desde un usuario estándar.

Evidencia de Configuración SSH Insegura:

```
debian@debian:~$ grep "PermitRootLogin" /etc/ssh/sshd_config
PermitRootLogin yes
# the setting of "PermitRootLogin prohibit-password".
```

---

4. Contención y Erradicación

Tras finalizar la fase de análisis forense y confirmar la alteración de la integridad del sistema, se procedió a la ejecución inmediata de medidas de contención. El objetivo prioritario de esta fase es aislar el vector de ataque, revocar accesos no autorizados y restablecer la configuración segura de los servicios críticos para garantizar la continuidad operativa del Hospital.

4.1 Restablecimiento de Permisos Seguros (Webroot)

Se detectó que la estructura de permisos del directorio web (`/var/www/html`) había sido modificada maliciosamente a **777** (lectura/escritura/ejecución global), lo que facilitaba la persistencia del atacante. Se procedió a revertir estos cambios aplicando el principio de mínimo privilegio.

- **Acción Correctiva:** Normalización recursiva de permisos en directorios y ficheros, y blindaje específico del archivo de configuración crítico.

Desglose Técnico de la Remediación:

Para revertir los permisos inseguros de forma masiva sin romper la funcionalidad del servidor web, se utilizaron comandos `find` con la siguiente lógica:

- **Comando:** `sudo find /var/www/html -type d -exec chmod 755 {} \;`
- `find /var/www/html`: Indica al sistema que busque dentro del directorio raíz de la web.
- `-type d` / `-type f`: Filtra la búsqueda.
  - `d`: Solo aplica cambios a **directorios** (carpetas).

- `f`: Solo aplica cambios a **ficheros** (archivos).

- `-exec ... {} \;`: Ejecuta un comando sobre cada elemento encontrado. `{}` es el marcador de posición del archivo y `\;` indica el fin del comando.
- **`chmod 755` (Para Directorios):**
  - Asigna permisos `rwxr-xr-x`.
  - **Justificación:** El propietario (root/www-data) tiene control total. El resto solo puede listar y entrar en la carpeta. Es necesario el permiso de ejecución (`x`) en carpetas para poder acceder a ellas.
- **`chmod 644` (Para Ficheros):**
  - Asigna permisos `rw-r--r--`.
  - **Justificación:** Elimina el permiso de ejecución (`x`) para evitar que scripts maliciosos se ejecuten como programas del sistema, y elimina el permiso de escritura (`w`) para usuarios no privilegiados, evitando la modificación del código web.

Evidencia de Corrección (Permisos Restaurados):

```
debian@debian:~$ sudo find /var/www/html -type d -exec chmod 755 {} \;
```

```
debian@debian:~$ sudo find /var/www/html -type f -exec chmod 644 {} \;
```

```
debian@debian:~$ ls -la /var/www/html
total 256
drwxr-xr-x  5 www-data www-data  4096 Oct  8  2024 .
drwxr-xr-x  3 root     root      4096 Sep 30  2024 ..
-rw-r--r--  1 www-data www-data   523 Sep 30  2024 .htaccess
-rw-r--r--  1 www-data www-data 10701 Sep 30  2024 index.html
-rw-r--r--  1 www-data www-data   405 Feb  6  2020 index.php
-rw-r--r--  1 www-data www-data 19915 Dec 31  2023 license.txt
-rw-r--r--  1 www-data www-data  7409 Jun 18  2024 readme.html
-rw-r--r--  1 www-data www-data  7387 Feb 13  2024 wp-activate.php
drwxr-xr-x  9 www-data www-data  4096 Sep 10  2024 wp-admin
-rw-r--r--  1 www-data www-data   351 Feb  6  2020 wp-blog-header.php
-rw-r--r--  1 www-data www-data  2323 Jun 14  2023 wp-comments-post.php
-rw-r--r--  1 www-data www-data  3017 Sep 30  2024 wp-config.php
drwxr-xr-x  5 www-data www-data  4096 Oct  8  2024 wp-content
-rw-r--r--  1 www-data www-data  5638 May 30  2023 wp-cron.php
drwxr-xr-x 30 www-data www-data 12288 Sep 10  2024 wp-includes
-rw-r--r--  1 www-data www-data  2502 Nov 26  2022 wp-links-opml.php
-rw-r--r--  1 www-data www-data  3937 Mar 11  2024 wp-load.php
-rw-r--r--  1 www-data www-data 51238 May 28  2024 wp-login.php
-rw-r--r--  1 www-data www-data  8525 Sep 16  2023 wp-mail.php
-rw-r--r--  1 www-data www-data 28774 Jul  9  2024 wp-settings.php
-rw-r--r--  1 www-data www-data 34385 Jun 19  2023 wp-signup.php
-rw-r--r--  1 www-data www-data  4885 Jun 22  2023 wp-trackback.php
-rw-r--r--  1 www-data www-data  3246 Mar  2  2024 xmlrpc.php
```

4.2 Hardening del Servicio SSH

Para mitigar el riesgo de ataques de fuerza bruta exitosos contra la cuenta de *superusuario*, se modificó la configuración del servicio SSH.

- **Acción:** Deshabilitar el inicio de sesión directo para `root`.
- **Comando de remediación:** `sudo sed -i 's/PermitRootLogin yes/PermitRootLogin no/' /etc/ssh/sshd_config`
- **Validación:** `grep "PermitRootLogin" /etc/ssh/sshd_config`

Evidencia de Corrección:

```
debian@debian:~$ sudo sed -i 's/PermitRootLogin yes/PermitRootLogin no/' /etc/ssh/sshd_config
debian@debian:~$ grep "PermitRootLogin" /etc/ssh/sshd_config
PermitRootLogin no
# the setting of "PermitRootLogin prohibit-password".
```

4.3 Rotación de Credenciales de Base de Datos

Se mitigó la vulnerabilidad crítica de uso de contraseñas por defecto (`123456`) para el usuario de base de datos `wordpressuser`.

- **Acción:** Cambio de contraseña en el motor de base de datos y actualización sincronizada en el fichero de configuración de WordPress.
- **Nueva Política:** Se estableció una contraseña robusta (`HospGenMad_Secure!2025`) que cumple con los requisitos de complejidad del SGSI del Hospital.
- **Comandos ejecutados:**
  1. Actualización en MySQL:
     - `sudo mysql -u root -e "ALTER USER 'wordpressuser'@'localhost' IDENTIFIED BY 'HospGenMad_Secure\!2025';"`
  2. Actualización en wp-config:
     - `sudo sed -i "s/123456/HospGenMad_Secure!2025/" /var/www/html/wp-config.php`
  3. Verificar los cambios en las credenciales de Wordpress:
     - `sudo grep "DB_PASSWORD" /var/www/html/wp-config.php`
  4. Verificar el acceso a la base de datos:
     - `mysql -u wordpressuser -p'HospGenMad_Secure!2025' -e "status"`

Evidencia de Cambio de Credenciales:

- Cambiar la contraseña de la base datos de MySQL.

- Cambiar la contraseña de acceso en Wordpress.

```
debian@debian:~$ sudo sed -i "s/123456/HospGenMad_Secure\!2025/" /var/www/html/wp-config.php
```

- Verificación de los cambios hechos en el archivo `wp-config.php` que almacena las credenciales de acceso de Wordpress.

```
debian@debian:~$ sudo grep "DB_PASSWORD" /var/www/html/wp-config.php
define( 'DB_PASSWORD', 'HospGenMad_Secure!2025' );
```

- Verificación de acceso con la nueva contraseña en MySQL haciendo que muestre en pantalla los datos del servidor.

```
debian@debian:~$ mysql -u wordpressuser -p'HospGenMad_Secure!2025' -e "status"
--------------
mysql  Ver 15.1 Distrib 10.11.6-MariaDB, for debian-linux-gnu (x86_64) using  EditLine wrapper

Connection id:          33
Current database:
Current user:           wordpressuser@localhost
SSL:                    Not in use
Current pager:          stdout
Using outfile:          ''
Using delimiter:        ;
Server:                 MariaDB
Server version:         10.11.6-MariaDB-0+deb12u1 Debian 12
Protocol version:       10
Connection:             Localhost via UNIX socket
Server characterset:    utf8mb4
Db     characterset:    utf8mb4
Client characterset:    utf8mb3
Conn.  characterset:    utf8mb3
UNIX socket:            /run/mysqld/mysqld.sock
Uptime:                 2 hours 52 min 20 sec

Threads: 1  Questions: 63  Slow queries: 0  Opens: 33  Open tables: 26  Queries per second avg: 0.006
--------------
```

4.4 Prevención de Fuga de Información (Directory Listing)

Se detectó que la configuración del servidor web permitía el **listado de directorios** (`Options Indexes`), lo que supone un riesgo de divulgación de información al permitir a un atacante ver la estructura de ficheros si no existe un archivo índice.

Permite que, si un usuario entra en una carpeta de la web y no existe un archivo `index.php` o `index.html`, el servidor muestre automáticamente una **lista de todos los archivos** que hay dentro.

- **Comando de verificación:** `cat /etc/apache2/apache2.conf | grep Indexes`
- **Hallazgo:** La directiva `Indexes` estaba habilitada.
- **Acción Correctiva:** Se creó un fichero de control de acceso (`.htaccess`) en la raíz del sitio para forzar la deshabilitación del listado de ficheros.
- **Comando de remediación:** `echo "Options -Indexes" | sudo tee /var/www/html/.htaccess`

Evidencia de Hardening:

```
debian@debian:~$ cat /etc/apache2/apache2.conf | grep Indexes
        Options Indexes FollowSymLinks
        Options Indexes FollowSymLinks
        Options Indexes FollowSymLinks
#       Options Indexes FollowSymLinks
debian@debian:~$ echo "Options -Indexes" | sudo tee /var/www/html/.htaccess
Options -Indexes
```

5. Conclusiones y Recomendaciones (SGSI)

5.1 Resumen del Incidente

La investigación forense confirma que el activo crítico fue comprometido debido a una combinación de **negligencias en la configuración base** (permisos 777 y contraseñas por defecto) y falta de controles de acceso (SSH Root habilitado). El atacante logró ocultar su rastro eliminando los logs de sistema, dificultando la atribución de la autoría.

Basado en los permisos 777 encontrados y la contraseña débil, la hipótesis más probable es que el atacante automatizado explotó la exposición pública del servicio web o FTP para subir una webshell y, desde ahí, escalar privilegios aprovechando la configuración SSH insegura.

5.2 Estado Final de Seguridad

Tras las medidas de contención aplicadas, el servidor se encuentra actualmente **operativo y asegurado**. Se han cerrado los vectores de ataque identificados y se ha restablecido la integridad de los permisos.

5.3 Recomendaciones Estratégicas (Alineación ISO 27001)

Para evitar la recurrencia de este incidente en el Hospital General de Madrid, se proponen las siguientes mejoras al SGSI:

1. **Centralización de Logs (SIEM):** Configurar el envío de logs a un servidor remoto seguro para evitar que un atacante pueda borrarlos localmente (Anti-Forense).
2. **Política de Contraseñas Robusta:** Implementar controles técnicos que impidan configurar claves débiles como “123456”.
3. **Habilitación de WAF:** Desplegar un Web Application Firewall (como ModSecurity) para detectar intentos de escaneo y ataques web en tiempo real.