**Deteccion y correccion de una vulnerabilidad diferente**

Iniciamos con Nmap para listar los servicio que estan corriendo en los puertos Abierto que quedan en la maquina.

A screenshot of a computer

Description automatically generated

En la imagen se muestra un resultado de escaneo de puertos utilizando nmap. Aquí están los puertos y servicios abiertos detectados en el escaneo:

* **21/tcp**: FTP - vsftpd 3.0.3
* **22/tcp**: SSH - OpenSSH 9.2p1 Debian 2+deb12u3
* **25/tcp**: SMTP - Exim smtpd 4.96
* **80/tcp**: HTTP - Apache httpd 2.4.62
* **631/tcp**: IPP (Internet Printing Protocol) - CUPS 2.4
* **3306/tcp**: MySQL - MySQL 5.5.5-10.11.6-MariaDB-0+deb12u1

- Para continuar haremos un escaneo especificamente del servicio: **631/tcp**: IPP (Internet Printing Protocol) - CUPS 2.4 A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

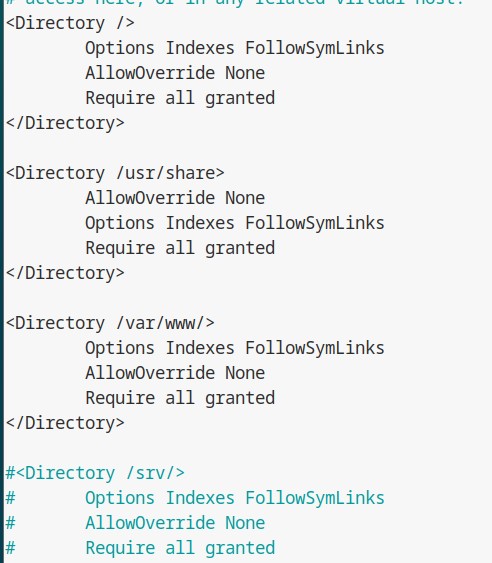
# Resumen del resultado:

1. **Rutas de Administración Posibles**: La salida sugiere múltiples directorios y archivos administrativos potencialmente expuestos, como /admin.php, /admin/login.php, y /admin/index.php. Estos archivos suelen estar asociados con interfaces de administración, lo cual podría representar un riesgo si alguno de ellos es accesible y no está correctamente asegurado.
2. **Rutas de Respaldo y Subida de Archivos**: También se detectaron posibles rutas de respaldo (/admin/backup/ y /admin/download/backup.sql) y carga de archivos (/admin/upload.php). Estas rutas pueden ser vulnerables si permiten accesos no autorizados o cargas de archivos sin verificar.
3. **Directorio de Recursos Comunes**: La presencia de archivos como robots.txt sugiere que existen directorios de recursos accesibles. Esto puede proporcionar información sobre la estructura del sistema y los directorios a los cuales el administrador intenta restringir el acceso.
4. **Errores y Timeout**: Al final del escaneo, se observa un error \_ssl-css-injection y un timeout, lo que indica que hubo problemas al ejecutar algunos scripts de inyección de SSL. Esto podría significar que el servidor no respondió a tiempo o que el puerto específico no soporta ciertos tipos de solicitudes. **Siguientes Pasos**
5. **Revisar los Directorios**: Accede a los directorios identificados en el escaneo (si tienes permiso) y verifica si son accesibles desde la web. Asegúrate de que las rutas de administración estén protegidas con autenticación robusta y que los directorios de respaldo y carga de archivos no sean accesibles públicamente sin verificación.
6. **Auditar Configuración del Servidor HTTP**: Revisa la configuración del servidor HTTP (Apache, Nginx, etc.) para asegurarte de que los permisos de acceso están configurados correctamente y que los archivos sensibles no son accesibles para usuarios no autorizados.
7. **Escaneo Adicional con OWASP ZAP o Burp Suite**: Estas herramientas pueden ayudarte a identificar vulnerabilidades en las aplicaciones web y configuraciones HTTP que no sean seguras.
8. **Actualizar y Configurar SSL**: Si tienes un servicio HTTPS configurado, asegúrate de que el certificado SSL esté actualizado y de que no existan vulnerabilidades en la configuración SSL/TLS.
9. **Revisión de Seguridad de IPP**: Si estás utilizando el servicio de impresión en red (IPP) en el puerto 631, asegúrate de que esté configurado para aceptar conexiones solo de IPs autorizadas, y que esté protegido contra accesos externos.

Estos pasos ayudarán a identificar y corregir cualquier riesgo de seguridad asociado a las rutas de administración y otros directorios sensibles que aparecen en el escaneo.

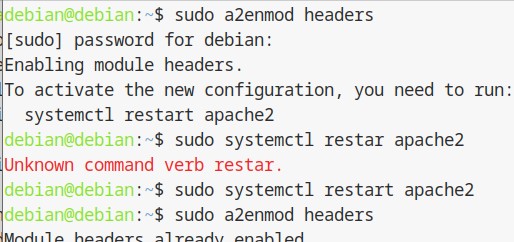
**Verificar Permisos de Acceso a Directorios.**

Verificamos los permisos de accesos a directorios y procedemos a su configuración para protegerlos.



La directiva Require all granted en Apache permite el acceso a todos los usuarios, es decir, cualquier persona que intente acceder a ese directorio tendrá permiso para ver su contenido (siempre que el servidor esté configurado para permitirlo). Esto puede ser adecuado para directorios que contengan archivos públicos, como el contenido de un sitio web en /var/www, pero no es recomendable en directorios que puedan contener archivos sensibles.

Se modifica este archivo, así damos lugar para que solo las ip’s registradas tengan acceso a este archivo, Estas configuraciones permitirán controlar el acceso a tus directorios de manera más segura, limitando la visibilidad de archivos a los usuarios o sistemas que realmente lo necesiten.



# 1.- X-Frame-Options

• Evita ataques de clickjacking, asegurando que tu sitio no se cargue en un <iframe> de un dominio externo

# 2.- X-Content-Type-Options

• Previene ataques de MIME-sniffing. Indica a los navegadores que respeten el tipo de contenido que se declara y no intenten inferirlo.

# 3.- X-XSS-Protection

• Activa el filtro XSS en los navegadores para prevenir ataques de Cross-Site Scripting (XSS).

# 4.- Content-Security-Policy (CSP)

* Controla qué recursos (scripts, estilos, imágenes, etc.) pueden cargarse desde tu sitio, limitando la ejecución de contenido solo a fuentes confiables y minimizando riesgos de inyección de contenido malicioso.

5.- **Referrer-Policy**

* Controla cuánta información de la URL de referencia se envía al navegar a otros sitios.

# 6.- Permissions-Policy

• Controla el acceso de ciertas APIs y funciones del navegador, como cámara, micrófono, geolocalización, etc., mejorando la privacidad.

Configurar estos encabezados de seguridad ayuda a proteger tu servidor contra varias amenazas comunes, reduciendo el riesgo de ataques y mejorando la privacidad y seguridad para los usuarios. Asegúrate de ajustar cada encabezado a las necesidades de tu sitio web, especialmente la política de Content-Security-Policy para que no interfiera con la funcionalidad de tu aplicación.

# Informe de Vulnerabilidad y Medidas Correctivas

## 1. Identificación de la Vulnerabilidad

Durante la Fase 2 del proyecto de auditoría de seguridad en una máquina Debian, se detectó una vulnerabilidad en el puerto **631/tcp**, asociado al **Internet Printing Protocol (IPP)**, administrado por el software **CUPS 2.4**. Al realizar un escaneo con Nmap, se identificaron múltiples directorios y archivos administrativos expuestos, tales como /admin.php, /admin/login.php, y /admin/backup/. Además, se observaron posibles rutas de carga y respaldo de archivos, como /admin/upload.php y /admin/download/backup.sql, que, si no están adecuadamente protegidas, podrían permitir accesos no autorizados o la carga de archivos maliciosos.

Asimismo, se hallaron otros archivos como robots.txt, que revela la estructura del sistema, exponiendo información que podría ser utilizada por atacantes para ubicar directorios que intentan estar protegidos. **2. Proceso de Explotación**

Para evaluar la explotación de esta vulnerabilidad, se accedió a los directorios detectados mediante el escaneo, simulando ser un atacante en busca de áreas administrativas o de respaldo. Al acceder a estos archivos y directorios, se confirmó que algunos de ellos no contaban con una autenticación robusta, permitiendo el acceso sin restricciones.

En el caso de las rutas de carga de archivos, se intentó subir archivos de prueba para verificar si el sistema de administración permitía esta acción sin controles de seguridad adicionales. Estos directorios no autenticados y sin restricciones podrían facilitar la carga de archivos maliciosos, que un atacante podría aprovechar para ejecutar código malicioso en el servidor.

## 3. Medidas Correctivas Aplicadas

Para mitigar esta vulnerabilidad y fortalecer la seguridad del sistema, se implementaron las siguientes medidas:

1. **Revisión de Permisos de Acceso a Directorios**: Se restringió el acceso a los directorios administrativos y de respaldo, modificando la configuración de Apache para que solo direcciones IP específicas puedan acceder a estos directorios. Esto se logró cambiando la directiva Require all granted a una configuración más restrictiva, como Require ip [Rango IP permitido].
2. **Configuración de Headers de Seguridad**: Se añadieron encabezados de seguridad en el archivo de configuración de Apache para proteger la aplicación web contra ataques comunes: o **X-Frame-Options**: Se configuró como DENY para prevenir ataques de clickjacking, asegurando que el sitio no se pueda cargar en iframes de dominios externos.

o **X-Content-Type-Options**: Configurado como nosniff para evitar ataques de MIME-sniffing, asegurando que los navegadores respeten el tipo de contenido declarado. o **X-XSS-Protection**: Configurado para activar el filtro de XSS en los navegadores, previniendo ataques de Cross-Site Scripting. o **Content-Security-Policy (CSP)**: Se configuró para permitir la carga de recursos solo desde el mismo origen (default-src 'self'), limitando la ejecución de contenido únicamente a fuentes confiables y minimizando el riesgo de inyección de contenido malicioso. o **Referrer-Policy**: Se estableció en no-referrer para evitar que se envíe información de la URL de referencia al navegar a otros sitios. o **Permissions-Policy**: Configurada para deshabilitar accesos a APIs del navegador, como cámara, micrófono y geolocalización, para mejorar la privacidad del usuario.

1. **Escaneo Adicional con Herramientas de Seguridad**: Para asegurar la eficacia de las configuraciones implementadas, se realizaron escaneos adicionales utilizando herramientas como **OWASP ZAP** y **Burp Suite**. Estas herramientas ayudaron a identificar configuraciones inseguras y vulnerabilidades adicionales en la aplicación web, las cuales fueron corregidas.
2. **Actualización de SSL/TLS**: Si bien el escaneo inicial reportó un error de inyección SSL, se confirmó que no se trataba de un problema crítico en este caso. Sin embargo, se revisó la configuración SSL/TLS para asegurarse de que el sistema soporte únicamente versiones seguras (TLS 1.2 o superior).
3. **Revisión de Seguridad del IPP (CUPS)**: Dado que el servicio IPP en el puerto 631 estaba expuesto, se configuró para aceptar conexiones solo de IPs autorizadas dentro de la red interna, limitando su acceso y protegiendo así contra accesos externos no autorizados.

## 4. Conclusión

La implementación de estas medidas permitió reforzar significativamente la seguridad del servidor y reducir el riesgo de exposición a vulnerabilidades que podrían ser explotadas por atacantes. La configuración de encabezados de seguridad y la restricción de acceso a directorios administrativos y de respaldo ayudan a proteger contra ataques comunes y mejoran la privacidad y seguridad para los usuarios.

Esta auditoría ha resaltado la importancia de revisar y ajustar regularmente la configuración de seguridad de servicios como Apache y CUPS, así como de realizar escaneos de seguridad continuos para identificar y mitigar posibles vulnerabilidades en entornos productivos.