

UNIVERSIDAD ANDINA DEL CUSCO
FACULTAD DE INGENIERÍA Y ARQUITECTURA
ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS



TEMA
“LINUX WEB SERVER HARDENING”

DOCENTE

Mgt. Ing. Ediwn Carrasco Poblete

ALUMNOS

Almanza Cuno, Gonzalo Jair

Condori Contretas, Mike Maraco

Ninantay Halanoca, Jesus Sergio

Villena Rojas, Oscar Phool

CUSCO – PERÚ

2023

Presentación

El presente trabajo titulado: “Linux Web Server Hardening” busca desplegar un servidor web Apache en Linux aplicando las recomendaciones básicas de los documentos NIST SP 800-123 y NIST SP 800-44 para su fortalecimiento. Donde se abordaron temas como el aseguramiento del sistema operativo del servidor, aseguramiento del servidor web, aseguramiento del contenido web, gestión de bitácoras, procedimientos de copias de respaldo (backup) del servidor. Todo ello se llevará a cabo haciendo uso de diferentes herramientas para cubrir con todos los temas antes descritos las cuales serán explicadas de forma detallada a lo largo del documento.

Introducción

Hoy en día la seguridad y protección de los datos dentro de las organizaciones es un aspecto fundamental ya que toda esta información en la mayoría de los casos se encuentra expuesta en la red y en varias ocasiones son objetivos de ataques por parte de ciberdelincuentes. Motivo por el cual invertir en una buena línea de defensa ayuda a evitar situaciones que comprometan a toda la organización.

Los servidores web Linux son ampliamente usados debido a su estabilidad, flexibilidad y seguridad, mas no son del todo seguros por ello es importante tomar medidas adicionales para fortalecerlos y evitar caer en vulnerabilidades.

El fortalecimiento de la seguridad tanto en el sistema operativo como en los servidores web implica el uso de medidas y prácticas que ayuden a proteger la información almacenada en el servidor con el objetivo de prevenir el acceso de intrusos y lograr garantizar la continuidad de los servicios que suelen ofrecer estos servidores. Para lograr esto se suele hacer uso de configuración de cortafuegos, actualización regular del sistema operativo y aplicaciones, uso de certificados SSL/TLS buscando la comunicación segura entre el servidor y los usuarios.

A lo largo de este trabajo se mostrarán las recomendaciones propuestas en los documentos NIST 800-123 y NIST 800-44 donde se aconseja la revisión de distintos procesos que suelen quedar en el aire al momento de instalar el sistema operativo del servidor o desplegar el servidor web. Por ello se hará uso de algunas herramientas para validar la seguridad de lo implementado a lo largo del documento.

Índice General

Presentación	2
Introducción	3
Índice General	4
Índice de Figuras	6
Índice de Tablas	7
I. Marco teórico	8
A. Sistema operativo Linux	8
B. Servidor web Apache	8
C. Contenido web	8
D. Gestión de bitácoras en el S.O. Linux y el servidor web Apache	9
E. Gestión de backups en el S.O. Linux y el servidor Apache	9
F. NIST 800-123.	10
G. NIST 800-44.	10
II. Implementación de recomendaciones NIST 800-123 y NIST 800-44	11
A. Aseguramiento del sistema operativo del servidor	11
1. Recomendaciones NIST	11
2. Implementación de recomendaciones	12
3. Lista de comprobación	20
B. Aseguramiento del servidor web	20
1. Recomendaciones NIST	20
2. Implementación de recomendaciones	21
3. Lista de comprobación	24
C. Aseguramiento del contenido Web	24
1. Recomendaciones NIST	24
2. Implementación de recomendaciones	24
3. Lista de comprobación	28
D. Gestión de bitácoras	29
1. Recomendaciones NIST	29
2. Implementación de recomendaciones	29
3. Lista de comprobación	33
E. Procedimientos de copia de respaldo del servidor	33
1. Recomendaciones NIST	33
2. Implementación de recomendaciones	34
3. Lista de comprobación	35
III. Conclusiones	36
IV. Referencias bibliográficas	37

Índice de Figuras

Figura 1: “Instalación mínima del Sistema Operativo”	12
Figura 2: “Verificación de repositorios actuales”	12
Figura 3: “Listado de paquetes disponibles”	13
Figura 4: “Listado de paquetes actualizados”	13
Figura 5: “Procesos en funcionamiento”	13
Figura 6: “Paquetes instalados, versión y descripción”	14
Figura 7: “Eliminación de paquetes innecesarios”	14
Figura 8: “Creación de usuario con permisos de administrador”	15
Figura 9: “Configuración de permiso para logueo por SSH del usuario root”	17
Figura 10: “Instalación del servidor web Apache”	17
Figura 11: “Aplicación de parches y actualizaciones”	18
Figura 12: “Instalación del servidor web Apache”	18
Figura 13: “Configuración del firewall para proteger el servidor web de posibles ataques usando UFW (Uncomplicated Firewall) para habilitar el tráfico HTTP y HTTPS”	19
Figura 14: “Instalación de SSL para habilitar la conexión HTTPS en Apache”	19
Figura 15: “Generación de certificado autofirmado”	20
Figura 16: “Creación de un archivo de configuración SSL”	20
Figura 17: “Reinicio del servidor, para aplicar los cambios”	21
Figura 18: “Instalación de nikto”	23
Figura 19: “Instalación de log4j2”	25
Figura 20: “Configuración de registro de actividad para el escritorio”	26
Figura 21: “Ruta de acceso al log”	29
Figura 22: “Logs registrados en el SO”	29
Figura 23: “Archivo de configuración de registro en el SO”	29
Figura 24: “Configuración de la rotación de registros”	30
Figura 25: “Eventos de autenticación fallidos en el SO”	31
Figura 26: “Eventos registrados en el SO”	31
Figura 27: “Archivo de configuración de registro de eventos en el SO”	32
Figura 28: “Configuración de la generación de registros en el servidor”	33
Figura 29: “Configuración de los módulos de registro en el servidor”	33
Figura 30: “Configuración de la rotación de registros”	34
Figura 31: “Registro de rotación de registros”	35
Figura 32: “Últimos 10 eventos registrados en el servidor”	36

Índice de Tablas

Tabla 1: “Lista de comprobación recomendaciones NIST 800-123 y NIST 800-44 v2”	20
Tabla 2: “Lista de comprobación recomendaciones NIST 800-123 y NIST 800-44 v2”	20
Tabla 3: “Lista de comprobación recomendaciones NIST 800-123 y NIST 800-44 v2”	26
Tabla 4: “Lista de comprobación recomendaciones NIST 800-123 y NIST 800-44 v2”	35
Tabla 5: “Lista de comprobación recomendaciones NIST 800-123 y NIST 800-44 v2”	38

I. Marco teórico

A. Sistema operativo Linux

Tanenbaum describe el sistema operativo Linux como "un sistema operativo completo y autónomo que se ha desarrollado en la línea de la tradición de Unix" (página 129). Linux comparte muchas de las características de Unix, como la compatibilidad con múltiples usuarios, la capacidad de ejecutar varios programas simultáneamente y el uso de una interfaz de línea de comandos para interactuar con el sistema.

En resumen, Linux es un sistema operativo basado en Unix que ha sido ampliamente adoptado debido a su naturaleza de código abierto y su capacidad para ejecutar en una amplia variedad de sistemas.

B. Servidor web Apache

El servidor web Apache es un software libre y de código abierto que permite a los usuarios publicar sitios web en Internet. Según la documentación oficial de Apache, este servidor web es "el servidor HTTP más popular del mundo" y se utiliza en una amplia variedad de sistemas operativos, incluyendo Linux (Apache, n.d.).

El servidor Apache es un software que se ejecuta en el sistema operativo y espera peticiones entrantes de los clientes a través de Internet. Cuando un cliente envía una solicitud a través de un navegador web, Apache procesa la solicitud y devuelve una respuesta al cliente. Según Wagner y otros (2012), "Apache maneja todas las solicitudes de los clientes y las envía al proceso apropiado dentro del sistema, ya sea para un archivo estático o un script dinámico" (p. 10).

C. Contenido web

El contenido web se refiere a cualquier información o material ya sea en forma de texto, imágenes, videos, audio, presentaciones, infografías, entre otros formatos, en palabras de Bill Gates, "el contenido es el rey", lo que significa que el contenido de calidad es esencial para el éxito en línea (Gates, 1996).

El contenido web puede ser creado por individuos, empresas o instituciones, y se publica en sitios web, blogs, redes sociales y otras plataformas en línea. El objetivo del contenido web es atraer y retener a los visitantes del sitio web o plataforma, y puede ser optimizado para los motores de búsqueda con el fin de mejorar su visibilidad y posicionamiento en los resultados de búsqueda.

Según Halvorson (2012), una estrategia de contenido sólida es esencial para asegurar que el contenido se alinee con los objetivos de negocio y las necesidades de los usuarios.

D. Gestión de bitácoras en el S.O. Linux y el servidor web Apache

Los sistemas operativos junto con sus servidores deben de ser seguros y gestionables, teniendo eso en cuenta las bitácoras son un gran aliado ya que son una forma de registrar los eventos que están ocurriendo en el sistema o el servidor, incluyendo eventos que hasta el administrador no se daría cuenta de lo que está sucediendo.

Para lograr una gestión de bitácoras correctamente se debe conocer el propósito de la bitácora, reconocer el formato en el que se está enviando, conocer los ataques propios de cada servicio.

En el caso de linux se puede encontrar dentro del directorio `/var/log/` y para otros sistemas Unix se encuentran en `/var/adm/`, algunas bitácoras encontradas comúnmente como herramientas tanto de línea de comandos o gráficos son: `syslog`, `rsyslog`, `logwatch`, `logrotate`.

En el caso de gestión de bitácoras en el servidor web apache puede registrar la bitacoras mediante el registro y diagnóstico de errores(Error Log) que encuentre al momento de procesar peticiones también puede procesar los accesos con un registro (Access Log) que tiene el cual es modificable según lo requiera el administrador.

E. Gestión de backups en el S.O. Linux y el servidor Apache

Los Backups son duplicados de cierta información, sin ningún tipo de procesamiento sobre la misma (compresión, etc). Dicho duplicado debe almacenarse en un medio distinto al que se encuentra la información original, siendo este utilizado al momento de tener problemas como es la eliminación de información relevante o problemas al momento de ejecutar la información entre otras más, los aspectos de un backups son completo, incremental y diferencial

En Linux y el servidor apache hay varias formas de realizar backups donde se pueden utilizar herramientas que estén especializadas como lo son TAR, Rsync, Amanda, Bacula, Duplicity, también se puede utilizar comandos pero requiere más conocimientos.

Comenzando con que archivos vamos a querer tener un respaldo pudiendo ser configuraciones, documentos o base de datos.

Para realizar la copia a un disco extraíble de un forma manual siendo la copia manual de los archivos con el siguiente comando `cp /etc/httpd/conf/httpd.conf /backup//`

Para programar algun backup de forma periódica se puede utilizando el comando 'cron' creando un archivo script como puede ser un ejemplo `0 3 * * *` `/ruta/al/script.sh` donde se programa que se ejecute el script a las 3am.

También se debe de establecer una política de retención para saber cuánto tiempo se deberá estar en el servidor los backups tomando en cuenta la cantidad de almacenamiento y la recuperación de datos.

F. NIST 800-123.

NIST Special Publication 800-123, "Guide to General Server Security", es una publicación del Instituto Nacional de Estándares y Tecnología (NIST) que brinda orientación sobre cómo proteger los sistemas de información que funcionan como plataformas de servidor. Cubre una amplia gama de temas de seguridad, incluida la seguridad del hardware y el software del servidor, la gestión segura de la configuración, los controles de acceso, el acceso remoto seguro y la respuesta a incidentes.

El documento está destinado a ayudar a las organizaciones a desarrollar e implementar políticas y procedimientos de seguridad efectivos para sus entornos de servidor. Brinda recomendaciones para protegerse contra una variedad de amenazas, incluido el malware, los ataques de denegación de servicio y el acceso no autorizado.

Algunas de las recomendaciones clave descritas en la guía incluyen:

- Garantizar que los servidores estén configurados de forma segura y actualizados con los últimos parches y actualizaciones de seguridad.
- Implementar controles de acceso para restringir el acceso a servidores y datos basados en un modelo de privilegios mínimos
- Uso del cifrado para proteger los datos en tránsito y en reposo
- Implementar mecanismos de autenticación sólidos para evitar el acceso no autorizado
- Monitoreo de servidores y sistemas en busca de signos de compromiso o actividad maliciosa
- Desarrollar e implementar un plan de respuesta a incidentes para detectar y responder rápidamente a incidentes de seguridad.

En general, NIST 800-123 proporciona una guía valiosa para las organizaciones que buscan mejorar la seguridad de sus entornos de servidores.

G. NIST 800-44.

La publicación especial NIST 800-44, "Pautas para proteger los servidores web públicos", es una publicación del Instituto Nacional de Estándares y Tecnología (NIST) que brinda orientación sobre cómo proteger los servidores web públicos. Está destinado a administradores de sistemas, gerentes de seguridad y otros profesionales de TI responsables de proteger los servidores web públicos.

El documento proporciona una descripción general de los riesgos de seguridad asociados con los servidores web públicos y describe una serie de controles de seguridad que las organizaciones pueden implementar para mitigar esos riesgos. Algunos de los temas clave cubiertos en el documento incluyen:

- Arquitectura de seguridad y consideraciones de diseño para servidores web públicos
- Prácticas seguras de gestión de la configuración del servidor
- Mecanismos de autenticación y control de acceso para restringir el acceso a datos confidenciales
- Protocolos de comunicación seguros, incluidos Transport Layer Security (TLS) y Secure Sockets Layer (SSL)
- Prácticas de monitoreo y respuesta a incidentes para detectar y responder a incidentes de seguridad
- Las mejores prácticas para proteger las aplicaciones web, incluida la validación de entrada y el manejo de errores.

En general, NIST 800-44 proporciona una guía valiosa para las organizaciones que buscan proteger los servidores web públicos y proteger los datos confidenciales del acceso no autorizado y otras amenazas de seguridad.

II. Implementación de recomendaciones NIST 800-123 y NIST 800-44ver2

A. Aseguramiento del sistema operativo del servidor

1. Recomendaciones NIST

Dentro de las recomendaciones tenemos:

- **Actualizar regularmente el sistema operativo y aplicaciones:** Es importante mantener el sistema operativo y las aplicaciones actualizadas con los últimos parches de seguridad para evitar vulnerabilidades.
- **Limitar el acceso a recursos del sistema:** Es necesario limitar el acceso a los recursos del sistema, como el registro del sistema y los archivos de configuración, solo a los usuarios autorizados.
- **Monitorear el servidor:** Es importante monitorear el servidor para detectar posibles intentos de acceso no autorizado, actividades maliciosas y otros eventos que puedan afectar la seguridad del servidor.
- **Configurar adecuadamente los permisos y roles:** Es necesario asignar roles y permisos adecuados a los usuarios y aplicaciones que acceden al servidor, con el fin de garantizar que solo tengan acceso a los recursos que necesitan y evitar la exposición de información confidencial.
- **Deshabilitar servicios innecesarios:** Es importante deshabilitar los servicios que no son necesarios en el servidor para reducir la superficie de ataque y minimizar. Es necesario configurar adecuadamente el firewall del sistema operativo para bloquear el tráfico no deseado y permitir solo el tráfico necesario para el funcionamiento del servidor.
- **Configurar adecuadamente el firewall:**
 - **Utilizar software de antivirus:** Es importante utilizar software de antivirus actualizado para proteger el servidor contra virus y otros tipos de malware.

2. Implementación de recomendaciones

Se implementarán las recomendaciones necesarias

1. Instalación mínima del sistema operativo

Se usará el sistema operativo Linux y la distribución Debian 11 en su versión 11.6 Bullseye, siguiendo los siguientes pasos.

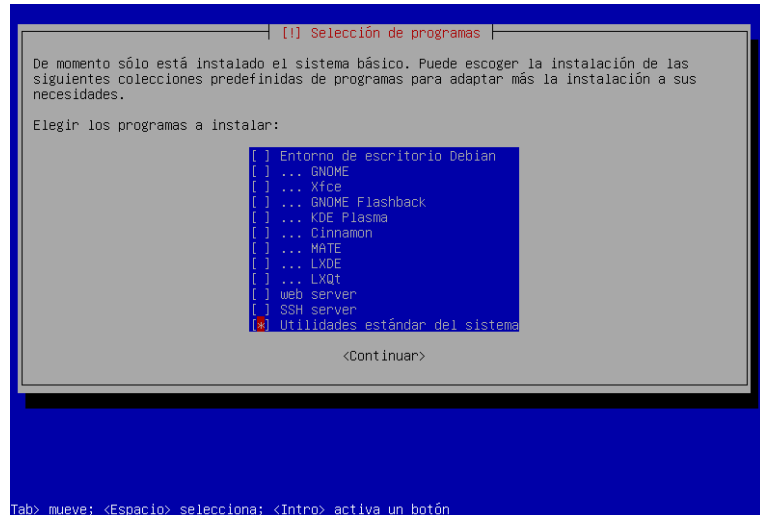


Figura 1: “Instalación mínima del Sistema Operativo” Fuente: “Elaboración propia”

Parches y actualizaciones del sistema operativo

```
cat /etc/apt/sources.list
```

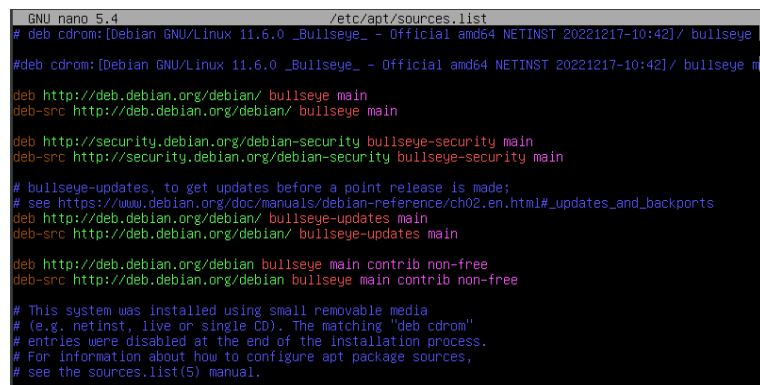


Figura 2: “Verificación de repositorios actuales” Fuente: “Elaboración propia”

Verificamos la lista de paquetes disponibles y actualizamos el sistema operativo.

```
apt-get update
```

```
root@Jair-Servidor:~# apt-get update
Obj:1 http://security.debian.org/debian-security bullseye-security InRelease
Obj:2 http://deb.debian.org/debian bullseye InRelease
Des:3 http://deb.debian.org/debian bullseye-updates InRelease [44,1 kB]
Leyendo lista de paquetes... Hecho
```

Figura 3: “Listado de paquetes disponibles” Fuente: “Elaboración propia”

```
apt-get upgrade
```

```
root@Jair-Servidor:~# apt-get upgrade
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Calculando la actualización... Hecho
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
root@Jair-Servidor:~#
```

Figura 4: “Listado de paquetes actualizados” Fuente: “Elaboración propia”

Actualizamos y reforzamos el SO, siguiendo las recomendaciones: Eliminamos los servicios, protocolos de red y aplicaciones que no necesitemos.

```
ss -tulpn
```

```
root@Jair-Servidor:~# ss -tulpn
Netid  State  Recv-Q  Send-Q  Local Address:Port  Peer Address:Port
Process
udp    UNCONN  0        0        0.0.0.0:68          0.0.0.0:*
users:(("dhclient",pid=487,fd=9))
tcp    LISTEN  0        5        0.0.0.0:873        0.0.0.0:*
users:(("rsync",pid=20173,fd=5))
tcp    LISTEN  0        511        *:443             *:443
users:(("apache2",pid=19809,fd=6),("apache2",pid=19808,fd=6),("apache2",pid=19807,fd=6))
tcp    LISTEN  0        5        [::]:873          [::]:*
users:(("rsync",pid=20173,fd=6))
tcp    LISTEN  0        511        *:80              *:80
users:(("apache2",pid=19809,fd=4),("apache2",pid=19808,fd=4),("apache2",pid=19807,fd=4))
root@Jair-Servidor:~# _
```

Figura 5: “Procesos en funcionamiento” Fuente: “Elaboración propia”

Los procesos que se encuentran funcionando son los necesarios, verificamos los paquetes instalados y en caso de tener paquetes que no sean necesarios los desinstalamos.

```
dpkg -list
```

Nombre	Versión	Arquitectura	Descripción
adduser	3.118	all	add and remove users
alsa-topology-conf	1.2.4-1	all	ALSA topology conf
alsa-ucm-conf	1.2.4-2	all	ALSA Use Case Manag
anacron	2.3-30	amd64	cron-like program
apache2	2.4.56-1~deb11u1	amd64	Apache HTTP Server
apache2-bin	2.4.56-1~deb11u1	amd64	Apache HTTP Server
apache2-data	2.4.56-1~deb11u1	all	Apache HTTP Server
apache2-dev	2.4.56-1~deb11u1	amd64	Apache HTTP Server
apache2-utils	2.4.56-1~deb11u1	amd64	Apache HTTP Server
apparmor	2.13.6-10	amd64	user-space parser
apt	2.2.4	amd64	commandline package
apt-listchanges	3.24	all	package change his
apt-utils	2.2.4	amd64	package management
aspell	0.60.8-3	amd64	GNU Aspell spell-c
aspell-es	1.11-16	all	Spanish dictionary
autoconf	2.69-14	all	automatic configur
automake	1:1.16.3-2	all	Tool for generating
autopoint	0.21-4	all	tool for setting up
autotools-dev	20180224.1+nm1	all	Update infrastruc
avahi-autoipd	0.8-5+deb11u1	amd64	Avahi IPv4LL netwo
base-files	11.1+deb11u6	amd64	Debian base system
base-passwd	3.5.51	amd64	Debian base system
bash	5.1-2+deb11u1	amd64	GNU Bourne Again S
bash-completion	1:2.11-2	all	programmable comp
bind9-dnsutils	1:9.16.37-1~deb11u1	amd64	Clients provided w
bind9-host	1:9.16.37-1~deb11u1	amd64	DNS Lookup Utility
bind9-libs:amd64	1:9.16.37-1~deb11u1	amd64	Shared Libraries
binutils	2.35.2-2	amd64	GNU assembler, lin
binutils-common:amd64	2.35.2-2	amd64	Common files for t
binutils-x86-64-linux-gnu	2.35.2-2	amd64	GNU binary utiliti
bluetooth	5.55-3.1	all	Bluetooth support

Figura 6: “Paquetes instalados, versión y descripción” Fuente: “Elaboración propia”

Eliminamos los paquetes que consideramos innecesarios.

```
apt-get purge bluetooth
```

```
root@Jair-Servidor:~# apt-get purge bluetooth
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.
  alsa-topology-conf alsa-ucm-conf bluez libasound2 libasound2-data libbdw1 libgl12.0-0
  libgl12.0-data shared-mime-info xdg-user-dirs
Utilice «apt autoremove» para eliminarlos.
Los siguientes paquetes se ELIMINARÁN:
  bluetooth*
0 actualizados, 0 nuevos se instalarán, 1 para eliminar y 0 no actualizados.
Se liberarán 72,7 kB después de esta operación.
¿Desea continuar? [S/n] s
(Leyendo la base de datos ... 44733 ficheros o directorios instalados actualmente.)
Desinstalando bluetooth (5.55-3.1) ...
root@Jair-Servidor:~# _
```

Figura 7: “Eliminación de paquetes innecesarios” Fuente: “Elaboración propia”

Configuramos un usuario con acceso de administrador, para el ingreso al sistema.

```
adduser usuario_seguridad

apt-get install sudo
```

```
root@Jair-Servidor:~# adduser usuario_seguridad
Añadiendo el usuario 'usuario_seguridad' ...
Añadiendo el nuevo grupo 'usuario_seguridad' (1001) ...
Añadiendo el nuevo usuario 'usuario_seguridad' (1001) con grupo 'usuario_seguridad' ...
Creando el directorio personal '/home/usuario_seguridad' ...
Copiando los ficheros desde '/etc/skel' ...
Nueva contraseña:
Vuelva a escribir la nueva contraseña:
passwd: contraseña actualizada correctamente
Cambiando la información de usuario para usuario_seguridad
Introduzca el nuevo valor, o pulse INTRO para usar el valor predeterminado
Nombre completo []:
Número de habitación []:
Teléfono del trabajo []:
Teléfono de casa []:
Otro []:
¿Es correcta la información? [S/n] s
root@Jair-Servidor:~# usermod -aG sudo usuario_seguridad
root@Jair-Servidor:~#
```

Figura 8: “Creación de usuario con permisos de administrador”
Fuente: “Elaboración propia”

Quitamos los privilegios para loguearse por SSH al usuario root, ya que al ser un nombre conocido es bastante inseguro.

```
nano /etc/ssh/sshd_config
```

```
#LoginGraceTime 2m
PermitRootLogin no
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

#PubkeyAuthentication yes

# Expect .ssh/authorized_keys2 to be disregarded by default in future.
#AuthorizedKeysFile .ssh/authorized_keys .ssh/authorized_keys2

#AuthorizedPrincipalsFile none

#AuthorizedKeysCommand none
#AuthorizedKeysCommandUser nobody

# For this to work you will also need host keys in /etc/ssh/ssh_known_hosts
#HostbasedAuthentication no

root@Jair-Servidor:~# systemctl restart ssh.service
root@Jair-Servidor:~#
```

Figura 9: “Configuración de permiso para logueo por SSH del usuario root” Fuente: “Elaboración propia”

3. Lista de comprobación

Tabla 1: “Lista de comprobación recomendaciones NIST 800-123 y NIST 800-44 v2”

RECOMENDACIONES NIST 800-123 Y NIST 800-44 v2	Check
Planificación de la instalación y la implementación del sistema operativo host y otros componentes para el servidor web	✓
Aplicación de parches y actualización del sistema operativo host según sea necesario	✓
Fortalezca y configure el sistema operativo para abordar la seguridad adecuadamente	✓
Instale y configure controles de seguridad adicionales, si es necesario.	
Probar el sistema operativo host para garantizar que los cuatro pasos anteriores abordaron adecuadamente todos los problemas de seguridad	

B. Aseguramiento del servidor web

1. Recomendaciones NIST

Las recomendaciones que encontramos en el NIST 800-44 y NIST 800-123 para el aseguramiento del servidor web son:

- **Configurar adecuadamente el servidor web:** Es importante configurar adecuadamente el servidor web para minimizar las vulnerabilidades y asegurar que los servicios se ejecuten de manera segura.
- **Actualizar regularmente el servidor web y aplicaciones:** Es necesario mantener el servidor web y las aplicaciones actualizadas con los últimos parches de seguridad para evitar vulnerabilidades.

- **Deshabilitar servicios y módulos innecesarios:** Es importante deshabilitar los servicios y módulos que no son necesarios en el servidor web para reducir la superficie de ataque y minimizar la exposición a vulnerabilidades.
- **Utilizar certificados SSL/TLS:** Es importante utilizar certificados SSL/TLS para proteger la comunicación entre el servidor web y los clientes y evitar que la información se envíe en texto plano.
- **Utilizar software de seguridad para servidor web:** Es necesario utilizar software de seguridad para servidor web, como firewalls de aplicaciones web y sistemas de detección de intrusiones, para proteger el servidor contra ataques y vulnerabilidades.
- **Realizar copias de seguridad y restauración:** Es necesario realizar copias de seguridad regularmente y establecer procedimientos para la restauración de datos en caso de una interrupción o desastre.

2. Implementación de recomendaciones

Instalamos el servidor web Apache

```

root@Jair-Servidor:~# apt-get install apache2
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  apache2-bin apache2-data apache2-utils libapr1 libaprutil1 libaprutil1-dbd-sqlite3
  libaprutil1-ldap libcurl4 liblua5.3-0 ssl-cert
Paquetes sugeridos:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom www-browser
Se instalarán los siguientes paquetes NUEVOS:
  apache2 apache2-bin apache2-data apache2-utils libapr1 libaprutil1 libaprutil1-dbd-sqlite3
  libaprutil1-ldap libcurl4 liblua5.3-0 ssl-cert
0 actualizados, 11 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
Se necesita descargar 2,843 kB de archivos.
Se utilizarán 9,231 kB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] s
Des:1 http://deb.debian.org/debian bullseye/main amd64 liblua5.3-0 amd64 5.3.3-1.1+b1 [120 kB]
Des:2 http://security.debian.org/debian-security bullseye-security/main amd64 libapr1 amd64 1.7.0-6+
  deb11u2 [106 kB]
Des:3 http://security.debian.org/debian-security bullseye-security/main amd64 libaprutil1 amd64 1.6.
  1-5+deb11u1 [92,3 kB]
Des:4 http://security.debian.org/debian-security bullseye-security/main amd64 libaprutil1-dbd-sqlite
  3 amd64 1.6.1-5+deb11u1 [19,0 kB]
Des:5 http://security.debian.org/debian-security bullseye-security/main amd64 libaprutil1-ldap amd64
  1.6.1-5+deb11u1 [17,2 kB]
Des:6 http://deb.debian.org/debian bullseye/main amd64 apache2-bin amd64 2.4.54-1~deb11u1 [1.425 kB]
Des:7 http://security.debian.org/debian-security bullseye-security/main amd64 libcurl4 amd64 7.74.0-
  1.3+deb11u7 [346 kB]
52% [6 apache2-bin 760 kB/1.425 kB 53%]_

```

*Figura 10: “Instalación del servidor web Apache” Fuente:
“Elaboración propia”*

Instalamos cualquier parche o actualización para corregir las vulnerabilidades del servidor.

```
root@Jair-Servidor:~# apt upgrade apache2
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
apache2 ya está en su versión más reciente (2.4.56-1~deb11u1).
Calculando la actualización... Hecho
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.
  alsa-topology-conf alsa-ucm-conf bluez libasound2 libasound2-data libdw1 libglib2.0-0
  libglib2.0-data shared-mime-info xdg-user-dirs
Utilice «apt autoremove» para eliminarlos.
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
root@Jair-Servidor:~# _
```

Figura 11: “Aplicación de parches y actualizaciones” Fuente: “Elaboración propia”

Verificamos los servicios instalados por el servidor Apache, en caso de tener instalados servicios innecesarios estos se eliminan.

```
root@Jair-Servidor:~# systemctl list-unit-files --type=service | grep apache
apache-htcacheclean.service          disabled      enabled
apache-htcacheclean@.service         disabled      enabled
apache2.service                      enabled       enabled
apache2@.service                     disabled      enabled
root@Jair-Servidor:~# _
```

Figura 12: “Instalación del servidor web Apache” Fuente: “Elaboración propia”

Eliminamos o deshabilitamos las cuentas predeterminadas generadas por el servidor.

Configuramos el firewall para proteger el servidor web, habilitando el tráfico HTTP y HTTPS en los puertos 80 y 443.

```
ufw allow "WWW"
```

```
ufw allow "WWW Full"
```

```
root@Jair-Servidor:/etc/ufw/applications.d# ufw allow 'WWW'
Rule added
Rule added (v6)
root@Jair-Servidor:/etc/ufw/applications.d# ufw app info 'WWW'
Profile: WWW
Title: Web Server
Description: Web server

Port:
  80/tcp
root@Jair-Servidor:/etc/ufw/applications.d# ufw app info 'WWW Full'
Profile: WWW Full
Title: Web Server (HTTP,HTTPS)
Description: Web Server (HTTP,HTTPS)

Ports:
  80,443/tcp
root@Jair-Servidor:/etc/ufw/applications.d# ufw allow 'WWW Full'
Rule added
Rule added (v6)
root@Jair-Servidor:/etc/ufw/applications.d#
```

Figura 13: “Configuración del firewall para proteger el servidor web de posibles ataques usando UFW (Uncomplicated Firewall) para habilitar el tráfico HTTP y HTTPS” Fuente: “Elaboración propia”

Instalamos un generador de certificados con la herramienta SSL-CERT para habilitar la conexión https en el servidor.

```
root@Jair-Servidor:/etc/ufw/applications.d# apt-get install ssl-cert
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
ssl-cert ya está en su versión más reciente (1.1.0+nmu1).
fijado ssl-cert como instalado manualmente.
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
root@Jair-Servidor:/etc/ufw/applications.d#
```

Figura 14: “Instalación de SSL para habilitar la conexión HTTPS en Apache” Fuente: “Elaboración propia”

Generamos un certificado autofirmado que dure 365 días.

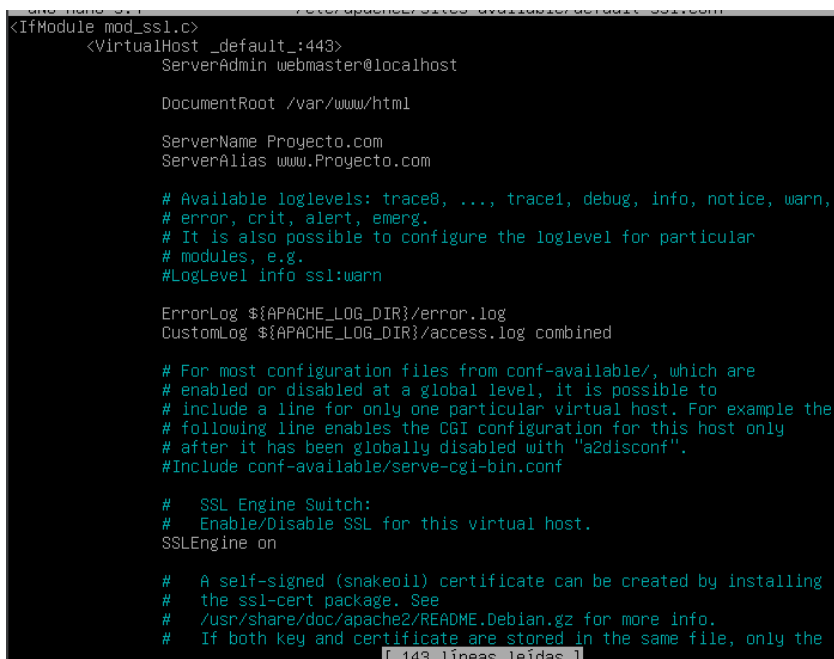
```
sudo openssl req -x509 -nodes -days 365
-newkey rsa:2048 -keyout
/etc/ssl/private/apache-selfsigned.key -out
/etc/ssl/certs/apache-selfsigned.crt
```

```
root@Jair-Servidor:~# openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/ssl/private/a
pache-selfsigned.key -out /etc/ssl/certs/apache-selfsigned.crt
Generating a RSA private key
.....+++++
.....+++++
writing new private key to '/etc/ssl/private/apache-selfsigned.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:PE
State or Province Name (full name) [Some-State]:Cusco
Locality Name (eg, city) []:Cusco
Organization Name (eg, company) [Internet Widgits Pty Ltd]:UAC
Organizational Unit Name (eg, section) []:
Common Name (e.g. server FQDN or YOUR name) []:
Email Address []:
root@Jair-Servidor:~#
```

*Figura 15: “Generación de certificado autofirmado” Fuente:
“Elaboración propia”*

Modificamos el archivo de configuración SSL para establecer conexiones seguras y privadas con los diferentes clientes en el servidor apache que se implementó.

```
nano
/etc/apache2/sites-available/default-ssl.conf
```



```
<!-- nano SSL --> /etc/apache2/sites-available/default-ssl.conf
<IfModule mod_ssl.c>
<VirtualHost _default_:443>
    ServerAdmin webmaster@localhost

    DocumentRoot /var/www/html

    ServerName Proyecto.com
    ServerAlias www.Proyecto.com

    # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
    # error, crit, alert, emerg.
    # It is also possible to configure the loglevel for particular
    # modules, e.g.
    #LogLevel info ssl:warn

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined

    # For most configuration files from conf-available/, which are
    # enabled or disabled at a global level, it is possible to
    # include a line for only one particular virtual host. For example the
    # following line enables the CGI configuration for this host only
    # after it has been globally disabled with "a2disconf".
    #Include conf-available/serve-cgi-bin.conf

    #
    # SSL Engine Switch:
    # Enable/Disable SSL for this virtual host.
    SSLEngine on

    #
    # A self-signed (snakeoil) certificate can be created by installing
    # the ssl-cert package. See
    # /usr/share/doc/apache2/README.Debian.gz for more info.
    # If both key and certificate are stored in the same file, only the
```

*Figura 16: “Creación de un archivo de configuración SSL”
Fuente: “Elaboración propia”*

Finalmente reiniciamos el servidor y los cambios realizados serán actualizados.

```
root@Jair-Servidor:~# systemctl restart apache2.service
root@Jair-Servidor:~# _
```

Figura 17: “Reinicio del servidor, para aplicar los cambios” Fuente: “Elaboración propia”

3. Lista de comprobación

Tabla 2: “Lista de comprobación recomendaciones NIST 800-123 y NIST 800-44 v2”

RECOMENDACIONES NIST 800-123 Y NIST 800-44 v2	Check
Instale el software del servidor en un host dedicado o en un sistema operativo invitado dedicado si se emplea la virtualización.	✓
Aplique cualquier parche o actualización para corregir vulnerabilidades conocidas en el software del servidor.	✓
Cree un disco físico dedicado o una partición lógica (separado del sistema operativo y la aplicación del servidor) para los datos del servidor, si corresponde.	
Elimine o deshabilite todos los servicios instalados por la aplicación del servidor, pero no requeridos.	✓
Utilizar certificados SSL/TLS para la seguridad del servidor	✓
Elimine o deshabilite todas las cuentas de usuario predeterminadas innecesarias creadas por la instalación del servidor.	
Eliminar toda la documentación de los fabricantes del servidor.	

C. Aseguramiento del contenido Web

1. Recomendaciones NIST

Entre las recomendaciones más importantes se encuentran las siguientes:

- **Análisis de riesgos y evaluación de vulnerabilidad:** Es importante realizar una evaluación de los riesgos y vulnerabilidades de seguridad del sitio web, y tomar medidas para mitigar y reducir los riesgos identificados.
- **Autenticación y control de acceso:** Se recomienda implementar medidas de autenticación sólidas, como la autenticación multifactor, para garantizar que solo los usuarios autorizados puedan acceder al contenido del sitio web.
- **Cifrado y protección de datos:** Se recomienda utilizar protocolos de cifrado y herramientas de protección de datos para garantizar la confidencialidad, integridad y disponibilidad del contenido web.
- **Monitoreo y registro de actividad:** Es esencial monitorear y registrar la actividad en el sitio web para detectar y responder rápidamente a cualquier actividad sospechosa.
- **Actualizaciones y parches de seguridad:** Es importante mantener el software del sitio web actualizado y aplicar los parches de seguridad de forma regular para evitar vulnerabilidades conocidas.
- **Gestión de incidentes:** Es importante tener un plan de gestión de incidentes en su lugar para responder rápidamente a cualquier incidente de seguridad en el sitio web y minimizar el impacto.

2. Implementación de recomendaciones

- **Análisis de riesgos y evaluación de la vulnerabilidad:**

Se pueden utilizar herramientas como Nikto, es una herramienta de evaluación de vulnerabilidad para escanear el sitio web en busca de posibles vulnerabilidades.

Para instalar Nikto en Linux, se pueden utilizar los siguientes comandos:

```
sudo apt update
```

```
sudo apt install nikto
```

```

Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  libnet-ssleay-perl libwhisker2-perl perl-openssl-defaults
Paquetes sugeridos:
  nmap
Se instalarán los siguientes paquetes NUEVOS:
  libnet-ssleay-perl libwhisker2-perl nikto perl-openssl-defaults
0 actualizados, 4 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
Se necesita descargar 701 kB de archivos.
Se utilizarán 3.567 kB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] s
Des:1 http://deb.debian.org/debian bullseye/main amd64 perl-openssl-defaults amd64 5 [7.360 B]
Des:2 http://deb.debian.org/debian bullseye/main amd64 libnet-ssleay-perl amd64 1.88-3+b1 [321 kB]
Des:3 http://deb.debian.org/debian bullseye/main amd64 libwhisker2-perl all 2.5-1.1 [108 kB]
Des:4 http://deb.debian.org/debian bullseye/non-free amd64 nikto all 1:2.1.5-3.1 [264 kB]
Descargados 701 kB en 1s (622 kB/s)
Seleccionando el paquete perl-openssl-defaults:amd64 previamente no seleccionado.
(Leyendo la base de datos ... 34882 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar .../perl-openssl-defaults_5.amd64.deb ...
Desempaquetando perl-openssl-defaults:amd64 (5) ...
Seleccionando el paquete libnet-ssleay-perl previamente no seleccionado.
Preparando para desempaquetar .../libnet-ssleay-perl_1.88-3+b1.amd64.deb ...
Desempaquetando libnet-ssleay-perl (1.88-3+b1) ...
Seleccionando el paquete libwhisker2-perl previamente no seleccionado.
Preparando para desempaquetar .../libwhisker2-perl_2.5-1.1_all.deb ...
Desempaquetando libwhisker2-perl (2.5-1.1) ...
Seleccionando el paquete nikto previamente no seleccionado.
Preparando para desempaquetar .../nikto_1:2.1.5-3.1_all.deb ...
Desempaquetando nikto (1:2.1.5-3.1) ...
Configurando perl-openssl-defaults:amd64 (5) ...
Configurando libwhisker2-perl (2.5-1.1) ...
Configurando libnet-ssleay-perl (1.88-3+b1) ...
Configurando nikto (1:2.1.5-3.1) ...
Procesando disparadores para man-db (2.9.4-2) ...
root@Jair-Servidor:~#

```

Figura 18: “Instalación de nikto” Fuente: “Elaboración propia”

Después de instalar esta herramienta, se ejecuta el siguiente comando para escanear el sitio web en busca de vulnerabilidades:

```
sudo nikto -h http://192.168.1.200:80
```

El resultado de este escaneo proporcionará una lista de posibles vulnerabilidades que deben ser abordadas para asegurar el contenido web.

- **Autenticación y control de acceso:**

El control de acceso y la autenticación son esenciales para garantizar que solo las personas autorizadas tengan acceso al contenido web. El NIST 800-63 proporciona directrices sobre la autenticación y el control de acceso.

Para ello solicitaremos autenticación para acceder al contenido web que se encontrará alojado en el servidor apache

```

sudo apt install apache2 apache2-utils

sudo a2enmod auth_basic

sudo a2enmod authz_core

```

Se configura la autenticación y el control de acceso utilizando el siguiente código en el archivo de configuración de Apache:

```

<Directory /var/www/html>

    AuthType Basic

    AuthName "Restricted Content"

```

```
AuthUserFile /etc/apache2/.htpasswd

Require valid-user

</Directory>
```

Este código configura el control de acceso para el directorio `/var/www/html` y utiliza el método de autenticación básico. También crea un archivo de contraseñas en `/etc/apache2/.htpasswd` para almacenar las credenciales de los usuarios autorizados.

- **Cifrado y protección de datos:**

El cifrado y la protección de datos son críticos para garantizar la confidencialidad y la integridad del contenido web. El NIST 800-52 proporciona directrices sobre el cifrado y la protección de datos.

Para implementar el cifrado y la protección de datos, se pueden utilizar herramientas como OpenSSL, que es una biblioteca de cifrado y protocolo de seguridad de redes de código abierto.

Para instalar OpenSSL en Linux, se pueden utilizar los siguientes comandos:

```
sudo apt update

sudo apt install openssl
```

Después de instalar OpenSSL, se puede generar un certificado SSL para el sitio web utilizando el siguiente comando:

```
sudo openssl req -x509 -nodes -days 365
-newkey rsa:2048 -keyout
/etc/ssl/private/http://192.168.1.200:80.k
ey -out
/etc/ssl/certs/http://192.168.1.200:80.crt
```

Este comando generará un certificado SSL válido por 365 días y lo almacenará en `/etc/ssl/private/` y `/etc/ssl/certs/` con el nombre del sitio web.

Después de generar el certificado SSL, se puede configurar Apache HTTP Server para utilizar el cifrado SSL en el sitio web utilizando el siguiente código en el archivo de configuración de Apache:

```
<VirtualHost *:443>

    ServerName http://192.168.1.200:80

    SSLEngine on

    SSLCertificateFile/etc/ssl/certs/http://
192.168.1.200:80.crt
```



```

    SSLCertificateKeyFile
    /etc/ssl/private/http://192.168.1.200:80
    .key

</VirtualHost>

```

Este código configura el sitio web para utilizar el cifrado SSL en el puerto 443 y utiliza el certificado SSL generado previamente.

- **Monitoreo y registro de actividad:**

El monitoreo y registro de actividad son esenciales para detectar y responder rápidamente a cualquier actividad sospechosa en el sitio web. El NIST 800-92 proporciona directrices sobre el monitoreo y registro de actividad.

Para implementar el monitoreo y registro de actividad, se pueden utilizar herramientas como Apache Log4j, que es una biblioteca de registro de actividad de código abierto para aplicaciones Java.

Para instalar Apache Log4j en Linux, se pueden utilizar los siguientes comandos:

```

sudo apt update

sudo apt install log4j2

```

```

root@Jair-Servidor:~# apt-get install liblog4j2-java
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  libapache-pom-java libcommons-codec-java libcommons-logging-java libcommons-parent-java
  libgoogle-gson-java libhttpclient-java libhttpcore-java liblightcouch-java libmongodb-java
  libslf4j-java
Paquetes sugeridos:
  libavalon-framework-java libcommons-logging-java-doc libexcalibur-logkit-java liblog4j1.2-java
  libgoogle-gson-java-doc libcommons-compress-java libcommons-csv-java
  libconversant-disruptor-java libdisruptor-java libgeronimo-jms-1.1-spec-java
  libjackson2-dataformat-xml-java libjackson2-dataformat-yaml libjansi-java libjctools-java
  libjeromq-java libmail-java libwoodstox-java
Se instalarán los siguientes paquetes NUEVOS:
  libapache-pom-java libcommons-codec-java libcommons-logging-java libcommons-parent-java
  libgoogle-gson-java libhttpclient-java libhttpcore-java liblightcouch-java liblog4j2-java
  libmongodb-java libslf4j-java
0 actualizados, 11 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
Se necesita descargar 6.650 kB de archivos.
Se utilizarán 11,2 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] s
Des:1 http://deb.debian.org/debian bullseye/main amd64 libapache-pom-java all 18-1 [4,676 B]
Des:2 http://deb.debian.org/debian bullseye/main amd64 libcommons-parent-java all 43-1 [10,8 kB]
Des:3 http://deb.debian.org/debian bullseye/main amd64 libcommons-codec-java all 1.15-1 [292 kB]
Des:4 http://deb.debian.org/debian bullseye/main amd64 libcommons-logging-java all 1.2-2 [62,2 kB]
Des:5 http://deb.debian.org/debian bullseye/main amd64 libgoogle-gson-java all 2.8.6-1+deb11ui [225
kB]
Des:6 http://deb.debian.org/debian bullseye/main amd64 libhttpcore-java all 4.4.14-1 [631 kB]
Des:7 http://deb.debian.org/debian bullseye/main amd64 libhttpclient-java all 4.5.13-2 [1,233 kB]
Des:8 http://deb.debian.org/debian bullseye/main amd64 liblightcouch-java all 0.0.6-1.1 [65,2 kB]
Des:9 http://deb.debian.org/debian bullseye/main amd64 libmongodb-java all 3.6.3-2 [1,901 kB]
55% [9 libmongodb-java 857 kB/1.901 kB 45%]

```

Figura 19: “Instalación de log4j2” Fuente: “Elaboración propia”

Después de instalar Apache Log4j, se puede configurar el registro de actividad en el sitio web utilizando el siguiente código en el archivo de configuración de Apache:

```
<Directory /var/www/html>

    Options FollowSymLinks

    AllowOverride None

    Require all granted

    LogLevel INFO

    ErrorLog /var/log/apache2/error.log

    CustomLog /var/log/apache2/access.log
    combined

</Directory>
```

```
GNU nano 5.4 /etc/apache2/apache2.conf
    Require all denied
</Directory>

<Directory /usr/share>
    AllowOverride None
    Require all granted
</Directory>

<Directory /var/www/>
    Options Indexes FollowSymLinks
    AllowOverride None
    Require all granted
</Directory>

<Directory /var/www/html>
    AuthType Basic
    AuthName "Restricted Content"
    AuthUserFile /etc/apache2/.htpasswd
    Require valid-user

    Options FollowSymLinks
    AllowOverride None
    Require all granted
    LogLevel INFO
    ErrorLog /var/log/apache2/error.log
    CustomLog /var/log/apache2/access.log combined
</Directory>

#<Directory /srv/>
#     Options Indexes FollowSymLinks
#     AllowOverride None

[Ctrl] [G] Ayuda      [Ctrl] [O] Guardar  [Ctrl] [B] Buscar   [Ctrl] [X] Cortar   [Ctrl] [E] Ejecutar  [Ctrl] [U] Ubicación [Ctrl] [M] Deshacer
[Ctrl] [X] Salir    [Ctrl] [R] Leer.fich. [Ctrl] [F] Reemplazar [Ctrl] [V] Pegar     [Ctrl] [J] Justificar [Ctrl] [A] Ir a línea [Ctrl] [H] Rehacer
```

Figura 20: “Configuración de registro de actividad para el escritorio” Fuente: “Elaboración propia”

Este código configura el registro de actividad para el directorio `/var/www/html` y registra los errores en `/var/log/apache2/error.log` y los accesos en `/var/log/apache2/access.log`.

- **Actualizaciones y parches de seguridad:**
 - **APT (Advanced Package Tool):**

Es un gestor de paquetes que permite instalar, actualizar y eliminar software en sistemas basados en Debian y Ubuntu. Para actualizar todos los paquetes del sistema, se usa el siguiente comando:

```
sudo apt-get update && sudo apt-get upgrade
```

3. Lista de comprobación

Tabla 3: “Lista de comprobación recomendaciones NIST 800-123 y NIST 800-44 v2”

Implementar medidas de autenticación y autorización para controlar el acceso al contenido web.	✓
Utilizar el cifrado para proteger la comunicación entre los usuarios y el servidor web.	✓
Implementar medidas de seguridad para proteger el servidor web, como el uso de firewalls, antivirus y actualizaciones de seguridad regulares.	✓
Implementar medidas de seguridad para proteger la base de datos del contenido web, como la encriptación de datos, la realización de copias de seguridad regulares y la monitorización del acceso a la base de datos.	
Controlar y limitar los privilegios de acceso a la base de datos y al servidor web para minimizar el riesgo de comprometer la seguridad.	✓
Realizar pruebas de penetración y evaluaciones de vulnerabilidades periódicas para identificar y remediar posibles vulnerabilidades en el contenido web.	
Establecer un plan de contingencia y recuperación de desastres en caso de que se produzca una interrupción del servicio o una pérdida de datos.	

D. Gestión de bitácoras

1. Recomendaciones NIST

- Configurar el sistema de registro de eventos del kernel

Las organizaciones deben definir los eventos que deben ser registrados y asegurarse de que los registros sean suficientemente detallados para ser útiles.

- Configurar la rotación de registros

Las organizaciones deben rotar los registros según su importancia, con registros más críticos que requieren una retención más larga que los registros menos críticos.

- Monitorear los registros

Las organizaciones deben monitorear los registros de forma continua para detectar actividades inusuales o sospechosas

- Definir una política de registro de eventos

Las organizaciones deben definir una política de registro de eventos que establezca qué eventos se deben registrar y cómo se deben registrar.

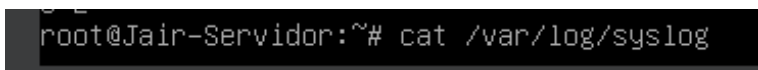
2. Implementación de recomendaciones

a) Linux

- **Configurar la generación de registros:**

La generación de registros en Linux se realiza mediante el uso del sistema syslog.

`(/var/log/syslog)`



```
root@Jair-Servidor:~# cat /var/log/syslog
```

Figura 21: “Ruta de acceso al log” Fuente: “Elaboración propia”

```

Mar 20 21:20:54 Jair-Servidor kernel: [ 2747.960791] [UFW BLOCK] IN=enp0s3 OUT= MAC=01:00:5e:00:00:
1:94:02:6b:57:7c:e1:08:00 SRC=192.168.1.1 DST=224.0.0.1 LEN=36 TOS=0x00 PREC=0x00 TTL=1 ID=0 DF PRO
0=2
Mar 20 21:21:24 Jair-Servidor kernel: [ 2778.040367] [UFW BLOCK] IN=enp0s3 OUT= MAC=01:00:5e:00:00:
1:94:02:6b:57:7c:e1:08:00 SRC=192.168.1.1 DST=224.0.0.1 LEN=36 TOS=0x00 PREC=0x00 TTL=1 ID=0 DF PRO
0=2
Mar 20 21:21:54 Jair-Servidor kernel: [ 2808.110100] [UFW BLOCK] IN=enp0s3 OUT= MAC=01:00:5e:00:00:
1:94:02:6b:57:7c:e1:08:00 SRC=192.168.1.1 DST=224.0.0.1 LEN=36 TOS=0x00 PREC=0x00 TTL=1 ID=0 DF PRO
0=2
Mar 20 21:22:24 Jair-Servidor kernel: [ 2838.179922] [UFW BLOCK] IN=enp0s3 OUT= MAC=01:00:5e:00:00:
1:94:02:6b:57:7c:e1:08:00 SRC=192.168.1.1 DST=224.0.0.1 LEN=36 TOS=0x00 PREC=0x00 TTL=1 ID=0 DF PRO
0=2
Mar 20 21:22:54 Jair-Servidor kernel: [ 2868.259766] [UFW BLOCK] IN=enp0s3 OUT= MAC=01:00:5e:00:00:
1:94:02:6b:57:7c:e1:08:00 SRC=192.168.1.1 DST=224.0.0.1 LEN=36 TOS=0x00 PREC=0x00 TTL=1 ID=0 DF PRO
0=2
Mar 20 21:23:02 Jair-Servidor systemd[1]: Starting The Apache HTTP Server...
Mar 20 21:23:02 Jair-Servidor apachectl[2960]: AH00526: Syntax error on line 185 of /etc/apache2/ap
ache2.conf:
Mar 20 21:23:02 Jair-Servidor apachectl[2960]: ErrorLog not allowed in <Directory> context
Mar 20 21:23:02 Jair-Servidor apachectl[2957]: Action 'start' failed.
Mar 20 21:23:02 Jair-Servidor apachectl[2957]: The Apache error log may have more information.
Mar 20 21:23:02 Jair-Servidor systemd[1]: apache2.service: Control process exited, code=exited, sta
tus=1/FAILURE
Mar 20 21:23:02 Jair-Servidor systemd[1]: apache2.service: Failed with result 'exit-code'.
Mar 20 21:23:02 Jair-Servidor systemd[1]: Failed to start The Apache HTTP Server.
Mar 20 21:23:25 Jair-Servidor kernel: [ 2898.340944] [UFW BLOCK] IN=enp0s3 OUT= MAC=01:00:5e:00:00:
1:94:02:6b:57:7c:e1:08:00 SRC=192.168.1.1 DST=224.0.0.1 LEN=36 TOS=0x00 PREC=0x00 TTL=1 ID=0 DF PRO
0=2
Mar 20 21:23:37 Jair-Servidor systemd[1]: Starting The Apache HTTP Server...
Mar 20 21:23:37 Jair-Servidor apachectl[2970]: AH00558: apache2: Could not reliably determine the s
erver's fully qualified domain name, using 127.0.1.1. Set the 'ServerName' directive globally to sup
press this message
Mar 20 21:23:37 Jair-Servidor systemd[1]: Started The Apache HTTP Server.
Mar 20 21:23:55 Jair-Servidor kernel: [ 2928.410853] [UFW BLOCK] IN=enp0s3 OUT= MAC=01:00:5e:00:00:
1:94:02:6b:57:7c:e1:08:00 SRC=192.168.1.1 DST=224.0.0.1 LEN=36 TOS=0x00 PREC=0x00 TTL=1 ID=0 DF PRO
0=2

```

Figura 22: “Logs registrados en el SO” Fuente: “Elaboración propia”

- **Configurar el sistema de registro de eventos del kernel:**

El kernel de Linux tiene un sistema de registro de eventos que puede ser utilizado para registrar varios eventos del sistema. Para configurar el sistema de registro de eventos del kernel, se puede utilizar el siguiente comando:

```
sudo nano /etc/rsyslog.conf
```

```

GNU nano 5.4 /etc/rsyslog.conf
#####
#### RULES ####
#####

#
# First some standard log files.  Log by facility.
#
auth,authpriv.* /var/log/auth.log
*.;auth,authpriv.none -/var/log/syslog
#cron.* /var/log/cron.log
daemon.* -/var/log/daemon.log
kern.* -/var/log/kern.log
lpr.* -/var/log/lpr.log
mail.* -/var/log/mail.log
user.* -/var/log/user.log

#
# Logging for the mail system.  Split it up so that
# it is easy to write scripts to parse these files.
#
mail.info -/var/log/mail.info
mail.warn -/var/log/mail.warn
mail.err /var/log/mail.err

#
# Some "catch-all" log files.
#
*.debug;\
auth,authpriv.none;\
mail.none -/var/log/debug
*.info;*.notice;*.warn;\
auth,authpriv.none;\
cron,daemon.none;\

```

Figura 23: “Archivo de configuración de registro en el SO” Fuente: “Elaboración propia”

Este comando abrirá el archivo de configuración de rsyslog en el editor nano. Dentro del archivo, se deben descomentar las líneas que comienzan con "module" para habilitar los módulos correspondientes, y luego agregar las

siguientes líneas al final del archivo para definir las reglas de registro de eventos

```
auth,authpriv.*    /var/log/auth.log

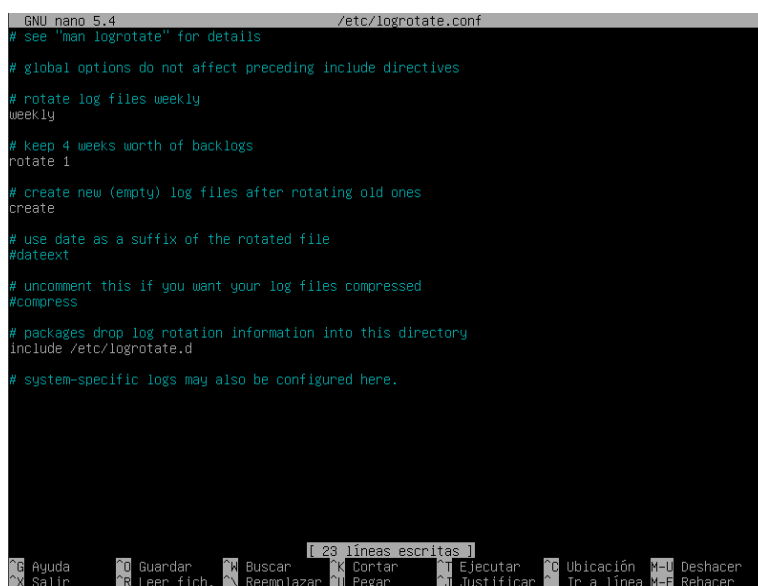
*.info;mail.none;authpriv.none;cron.none
/var/log/messages
```

Estas reglas registran eventos de autenticación y autorización en el archivo `/var/log/auth.log` y otros eventos informativos en el archivo `/var/log/messages`. Puede ajustar estas reglas según sea necesario para registrar eventos específicos.

○ **Configurar la rotación de registros:**

Es importante rotar los registros para evitar que los archivos de registro crezcan demasiado y ocupen todo el espacio en disco. Se puede configurar la rotación de registros utilizando el siguiente comando:

```
sudo nano /etc/logrotate.conf
```



```
GNU nano 5.4 /etc/logrotate.conf
# see "man logrotate" for details

# global options do not affect preceding include directives
# rotate log files weekly
weekly

# keep 4 weeks worth of backlogs
rotate 4

# create new (empty) log files after rotating old ones
create

# use date as a suffix of the rotated file
#dateext

# uncomment this if you want your log files compressed
#compress

# packages drop log rotation information into this directory
include /etc/logrotate.d

# system-specific logs may also be configured here.
```

[23 líneas escritas]

⌘ Ayuda ⌘ Guardar ⌘ Buscar ⌘ Cortar ⌘ Ejecutar ⌘ Ubicación ⌘ M-U Deshacer
⌘ Salir ⌘ Leer fich. ⌘ Reemplazar ⌘ Pegar ⌘ Justificar ⌘ Ir a línea ⌘ M-B Rehacer

*Figura 24: “Configuración de la rotación de registros”
Fuente: “Elaboración propia”*

- **Monitorear los registros:**

Es importante monitorear los registros regularmente para detectar posibles problemas de seguridad o errores del sistema. Se pueden utilizar varias herramientas de línea de comandos para monitorear los registros, como grep y tail. Por ejemplo, para buscar eventos de autenticación fallidos en el archivo /var/log/auth.log, se puede ejecutar el siguiente comando:

```
sudo grep "authentication failure" /var/log/auth.log
```

```
root@Jair-Servidor:~# grep "authentication failure" /var/log/auth.log
Mar 19 02:32:35 Jair-Servidor login[439]: pam_unix(login:auth): authentication failure; logname=LOGI
N uid=0 euid=0 tty=/dev/tty1 ruser= rhost=
root@Jair-Servidor:~#
```

*Figura 25: “Eventos de autenticación fallidos en el SO”
Fuente: “Elaboración propia”*

También se puede utilizar el comando tail para ver los registros más recientes en tiempo real. Por ejemplo, para ver los últimos 10 eventos registrados en el archivo /var/log/messages, se puede ejecutar el siguiente comando

```
sudo tail -n 10 /var/log/messages
```

```
Mar 19 02:32:16 Jair-Servidor kernel: [ 0.059940] ACPI: Reserving SSDT table memory at [mem 0xdff
f02a0-0xdfff0500]
Mar 19 02:32:16 Jair-Servidor kernel: [ 0.060087] No NUMA configuration found
Mar 19 02:32:16 Jair-Servidor kernel: [ 0.060089] Faking a node at [mem 0x0000000000000000-0x0000
0001357fffff]
Mar 19 02:32:16 Jair-Servidor kernel: [ 0.060097] NODE_DATA(0) allocated [mem 0x1357d2000-0x1357f
bfff]
Mar 19 02:32:16 Jair-Servidor kernel: [ 0.060331] Zone ranges:
Mar 19 02:32:16 Jair-Servidor kernel: [ 0.060332] DMA [mem 0x0000000000000000-0x0000000000
fffff]
Mar 19 02:32:16 Jair-Servidor kernel: [ 0.060334] DMA32 [mem 0x0000000000010000-0x00000000ff
fffff]
Mar 19 02:32:16 Jair-Servidor kernel: [ 0.060335] Normal [mem 0x0000000000000000-0x00000000135
7fffff]
Mar 19 02:32:16 Jair-Servidor kernel: [ 0.060336] Device empty
Mar 19 02:32:16 Jair-Servidor kernel: [ 0.060337] Movable zone start for each node
Mar 19 02:32:16 Jair-Servidor kernel: [ 0.060340] Early memory node ranges
Mar 19 02:32:16 Jair-Servidor kernel: [ 0.060340] node 0: [mem 0x0000000000000000-0x000000000
009efff]
Mar 19 02:32:16 Jair-Servidor kernel: [ 0.060341] node 0: [mem 0x0000000000000000-0x000000000
ffeffff]
Mar 19 02:32:16 Jair-Servidor kernel: [ 0.060342] node 0: [mem 0x0000000000000000-0x0000000013
57fffff]
Mar 19 02:32:16 Jair-Servidor kernel: [ 0.060343] Initmem setup node
Mar 19 02:32:16 Jair-Servidor kernel: [ 0.060343] Initmem setup node 0 [mem 0x0000000000000000-0x
000000001357fffff]
Mar 19 02:32:16 Jair-Servidor kernel: [ 0.060624] On node 0, zone DMA: 1 pages in unavailable ran
ges
Mar 19 02:32:16 Jair-Servidor kernel: [ 0.060647] On node 0, zone DMA: 97 pages in unavailable ra
nges
Mar 19 02:32:16 Jair-Servidor kernel: [ 0.075263] On node 0, zone Normal: 16 pages in unavailabl
e ranges
Mar 19 02:32:16 Jair-Servidor kernel: [ 0.075627] On node 0, zone Normal: 10240 pages in unavaila
ble ranges
Mar 19 02:32:16 Jair-Servidor kernel: [ 0.075923] ACPI: PM-Timer IO Port: 0x4008
Mar 19 02:32:16 Jair-Servidor kernel: [ 0.075972] IOAPIC[0]: apic_id 2, version 32, address 0xfec
00000, GSI 0-23
```

*Figura 26: “Eventos registrados en el SO” Fuente:
“Elaboración propia”*

- **Revisar los archivos de configuración del registro de eventos**

Los sistemas Linux tienen archivos de configuración que definen qué eventos se deben registrar y cómo se van a registrar

Por ejemplo, para revisar la configuración del registro de eventos del sistema en Ubuntu:

```
cat /etc/rsyslog.conf
```

```
#
# First some standard log files. Log by facility.
#
auth,authpriv.*          /var/log/auth.log
*.*;auth,authpriv.none   -/var/log/syslog
#cron.*                  /var/log/cron.log
daemon.*                 -/var/log/daemon.log
kern.*                   -/var/log/kern.log
lpr.*                    -/var/log/lpr.log
mail.*                   -/var/log/mail.log
user.*                   -/var/log/user.log

#
# Logging for the mail system. Split it up so that
# it is easy to write scripts to parse these files.
#
mail.info                -/var/log/mail.info
mail.warn                -/var/log/mail.warn
mail.err                 /var/log/mail.err

#
# Some "catch-all" log files.
#
*.*=debug;\
    auth,authpriv.none;\
    mail.none             -/var/log/debug
*.*=info;*.=notice;*.=warn;\
    auth,authpriv.none;\
    cron,daemon.none;\
    mail.none             -/var/log/messages

#
# Emergencies are sent to everybody logged in.
#
*.emerg                  :omusrmsg:*

root@Jair-Servidor: /home/backupServer#
```

Figura 27: “Archivo de configuración de registro de eventos en el SO” Fuente: “Elaboración propia”

b) Apache

- Configurar la generación de registros:

Se puede configurar Apache Server para generar registros abriendo la configuración de apache:

```
sudo nano/etc/apache2/apache2.conf
```

Se agregara las líneas de archivo que pueda configurar el registro de acceso y de errores:

```
LogFormat "%h %l %u %t \"%r\" %>s %b\n\"%(Referer) i\" \"%user{User-agent}i\" \" combined

CustomLog /var/log/apache2/access.log
combined

# Registros de error

ErrorLog /var/log/apache2/error.log
```



```
# Registros de acceso
LogFormat "%h %l %u %t \"%r\" %s %b \"%{Referer}i\" \"%{User-Agent}i\"" combined
CustomLog /var/log/apache2/access.log combined

# Registros de error
ErrorLog /var/log/apache2/error.log

# Include of directories ignores editors' and dpkg's backup files,
# see README.Debian for details.

# Include generic snippets of statements

root@Jair-Servidor:/home# systemctl restart apache2
root@Jair-Servidor:/home#
```

Figura 28: “Configuración de la generación de registros en el servidor” Fuente: “Elaboración propia”

- Configurar los módulos de registro de Apache

```
sudo nano /etc/apache2/conf/apache2.conf
```

Dentro del archivo de configuración, se pueden descomentar y ajustar las líneas relacionadas con los módulos de registro. Por ejemplo, para habilitar el registro de errores del servidor, se pueden agregar las siguientes líneas:

```
LogLevel info
```

```
# It is also possible to configure the log level for particular modules, e.g.
# "LogLevel info ssl:warn"
#
#LogLevel warn
LogLevel info

# Include module configuration:
IncludeOptional mods-enabled/*.load
IncludeOptional mods-enabled/*.conf

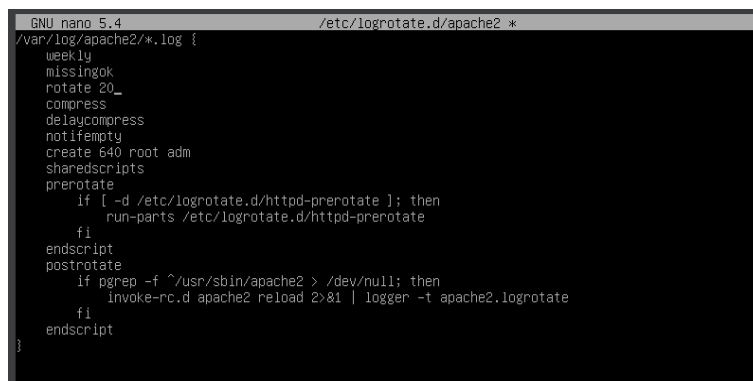
# Include list of ports to listen on
Include ports.conf

root@Jair-Servidor:/home# systemctl restart apache2
root@Jair-Servidor:/home#
```

Figura 29: “Configuración de los módulos de registro en el servidor” Fuente: “Elaboración propia”

- Configurar la rotación de registros

```
sudo nano /etc/logrotate.d/apache2
```



```
GNU nano 5.4 /etc/logrotate.d/apache2 *
/var/log/apache2/*.log {
    weekly
    missingok
    rotate 20
    compress
    delaycompress
    notifempty
    create 640 root adm
    sharedscripts
    prerotate
        if [ -d /etc/logrotate.d/httpd-prerotate ]; then
            run-parts /etc/logrotate.d/httpd-prerotate
        fi
    endscript
    postrotate
        if pgrep -f ^/usr/sbin/apache2 > /dev/null; then
            invoke-rc.d apache2 reload 2>&1 | logger -t apache2.logrotate
        fi
    endscript
}
```

*Figura 30: “Configuración de la rotación de registros”
Fuente: “Elaboración propia”*

```
/var/log/apache2/*.log {

    weekly

    missingok

    rotate 52

    compress

    delaycompress

    notifempty

    create 640 root adm

    sharedscripts

    postrotate

        etc/init.d/apache2 reload >/dev/null

    endscript

}
```

Esto indica que los archivos de registro que hay en la carpeta “/var/log/apache2” deben de ser rotados de una forma semanal “weekly” que tenga un máximo de 52 archivos “rotate 52” y que estén comprimidos “compress” después de esto, sino hay registros para rotar “missingok”, no se realizará ninguna acción, “notifempty” hará que el registro vacío no pueda rotar

Se establece que los nuevos archivos sean de propiedad del root y del grupo adm “create 640 root adm”, entonces después de la rotación, se recarga el servicio de apache “postrotate”, se terminará cerrando y guardando el archivo de configuración.

Para verificar la configuración si es válida se utiliza el comando:

```
sudo logrotate -d /etc/logrotate.d/apache2
```

Mostrando qué acciones puede realizar "logrotate", entonces si la configuración es correcta se ejecuta:

```
sudo logrotate /etc/logrotate.d/apache2
```

Donde se ejecutará la rotación de registros según la configuración que se estableció

```
rotating pattern: /var/log/apache2/*.log weekly (20 rotations)
empty log files are not rotated, old logs are removed
considering log /var/log/apache2/access.log
  Now: 2023-03-21 00:06
  Last rotated at 2023-03-21 00:00
  log does not need rotating (log has already been rotated)
considering log /var/log/apache2/error.log
  Now: 2023-03-21 00:06
  Last rotated at 2023-03-21 00:00
  log does not need rotating (log has already been rotated)
considering log /var/log/apache2/other_vhosts_access.log
  Now: 2023-03-21 00:06
  Last rotated at 2023-03-21 00:00
  log does not need rotating (log has already been rotated)
not running prerotate script, since no logs will be rotated
not running postrotate script, since no logs were rotated
root@Jair-Servidor:/home# logrotate /etc/logrotate.d/apache2
root@Jair-Servidor:/home# _
```

Figura 31: “Registro de rotación de registros” Fuente: “Elaboración propia”

- Monitorear los registros

Es importante monitorear los registros regularmente para detectar posibles problemas de seguridad o errores del servidor. Se pueden utilizar varias herramientas de línea de comandos para monitorear los registros, como grep y tail. Por ejemplo, para buscar eventos de errores en el archivo /var/log/httpd/error_log, se puede ejecutar el siguiente comando:

```
sudo grep "error" /var/log/apache2/error_log
```

Para ver los últimos 10 eventos

```
sudo tail -n 10 /var/log/apache2/error_log
```

```

considering log /var/log/apache2/error.log
Now: 2023-03-21 00:06
Last rotated at 2023-03-21 00:00
log does not need rotating (log has already been rotated)
considering log /var/log/apache2/other_vhosts_access.log
Now: 2023-03-21 00:06
Last rotated at 2023-03-21 00:00
log does not need rotating (log has already been rotated)
not running prerotate script, since no logs will be rotated
not running postrotate script, since no logs were rotated
root@Jair-Servidor:/home# logrotate /etc/logrotate.d/apache2
root@Jair-Servidor:/home# grep "error" /var/log
local/ lock/ log/
root@Jair-Servidor:/home# grep "error" /var/log/apache2/error.log
root@Jair-Servidor:/home# tail -n 10 /var/log/apache2/error.log
[Tue Mar 21 00:01:34.402720 2023] [socache_shmcb:info] [pid 19807:tid 139939761847616] AH00830: Shared memory socache initialised
[Tue Mar 21 00:01:34.402723 2023] [ssl:info] [pid 19807:tid 139939761847616] AH01887: Init: Initializing (virtual) servers for SSL
[Tue Mar 21 00:01:34.402725 2023] [ssl:info] [pid 19807:tid 139939761847616] AH01914: Configuring server Proyecto.com:443 for SSL protocol
[Tue Mar 21 00:01:34.403057 2023] [ssl:warn] [pid 19807:tid 139939761847616] AH01906: Proyecto.com:443:0 server certificate is a CA certificate (BasicConstraints: CA == TRUE !?)
[Tue Mar 21 00:01:34.403066 2023] [ssl:warn] [pid 19807:tid 139939761847616] AH01909: Proyecto.com:443:0 server certificate does NOT include an ID which matches the server name
[Tue Mar 21 00:01:34.403068 2023] [ssl:info] [pid 19807:tid 139939761847616] AH02568: Certificate and private key Proyecto.com:443:0 configured from /etc/ssl/certs/apache-selfsigned.crt and /etc/ssl/private/apache-selfsigned.key
[Tue Mar 21 00:01:34.403106 2023] [ssl:info] [pid 19807:tid 139939761847616] AH01876: mod_ssl/2.4.56 compiled against Server: Apache/2.4.56, Library: OpenSSL/1.1.1n
[Tue Mar 21 00:01:34.403665 2023] [mpm_event:notice] [pid 19807:tid 139939761847616] AH00489: Apache/2.4.56 (Debian) OpenSSL/1.1.1n configured -- resuming normal operations
[Tue Mar 21 00:01:34.403677 2023] [mpm_event:info] [pid 19807:tid 139939761847616] AH00490: Server built: 2023-03-08T03:05:04
[Tue Mar 21 00:01:34.403683 2023] [core:notice] [pid 19807:tid 139939761847616] AH00094: Command line: '/usr/sbin/apache2'
root@Jair-Servidor:/home# _

```

*Figura 32: “Últimos 10 eventos registrados en el servidor”
Fuente: “Elaboración propia”*

3. Lista de comprobación

Tabla 4: “Lista de comprobación recomendaciones NIST 800-123 y NIST 800-44 v2”

Seguimiento de las actividades que se realizaron por medio de un ataque	
Recuperación de configuración de registros en el sistema	✓
Implementar una política de registro de eventos para administrar su registro y como serán registrados las actividades	✓
Generar algún tipo de alerta de actividades sospechosas	
Configurar un registro de las actividades o elementos	✓
Monitorear los registros de las actividades que se realicen en el sistema	✓
Establecer una rotación de registros de los archivos que contengan las configuraciones para que no ocupe todo el espacio del disco	✓

E. Procedimientos de copia de respaldo del servidor

1. Recomendaciones NIST

Una de las funciones más importantes de un administrador de servidor web es mantener la integridad de los datos en el servidor web. Esto es importante porque los servidores web suelen ser algunos de los servidores más expuestos y vitales en la red de una organización.

Algunas recomendaciones que proporciona el NIST son:

- a) **Políticas y estrategias de copia de seguridad del servidor web:** El administrador del servidor web necesita realizar copias de seguridad del servidor web periódicamente por varios motivos. Un servidor web podría fallar como resultado de un acto malicioso o no intencional o una falla de hardware o software. Además, las agencias federales y muchas otras organizaciones se rigen por normas sobre la copia de seguridad y el archivo de datos del servidor web. Los datos del servidor web también deben respaldarse regularmente por razones legales y financieras.
- b) **Mantener un servidor web de prueba:** La mayoría de las organizaciones probablemente deseen mantener un servidor web de prueba o desarrollo. Idealmente, este servidor debería tener hardware y software idénticos al servidor web de producción o en vivo y estar ubicado en un segmento de red interna (intranet) donde pueda estar totalmente protegido por las defensas de la red perimetral de la organización.
- c) **Mantener una copia autorizada del contenido web de la organización:** Todas las organizaciones deben mantener una copia autorizada (es decir, verificada y de confianza) de sus sitios web públicos en un host al que no se pueda acceder desde Internet. Este es un complemento, pero no un reemplazo, de una política de copia de seguridad adecuada. Para sitios web simples y relativamente estáticos, esto podría ser tan simple como una copia del sitio web en un medio de sólo lectura (por ejemplo, disco compacto grabable [CDR]).

2. Implementación de recomendaciones

Para realizar un backup interdiario de los archivos del Servidor Apache en Debian 11, se hace uso el comando 'rsync'. 'rsync' es una herramienta de sincronización de archivos que permite hacer copias de seguridad incrementales y remotas de archivos.

Para hacer un backup interdiario de los archivos de tu servidor Apache con rsync, se debe realizar los siguientes pasos:

1. Crea un archivo de configuración para 'rsync' en la carpeta '/etc/rsyncd.conf':

```
sudo nano /etc/rsyncd.conf
```

```
GNU nano 5.4 /etc/rsyncd.conf
uid = nobody
gid = nogroup
use chroot = yes
max connections = 4
pid connections = 4
pid file = /var/run/rsyncd.pid
log file = /var/log/rsyncd.log
[apache-backup]
path = /var/www/
comment = Backup of Apache files
read only = true
list = false
auth users = backupuser
secrets file = /etc/rsyncd.secrets
exclude = /var/www/htdocs/hidden/*
exclude = /var/www/htdocs/private/*

[ 16 líneas leídas ]
Ayuda  Guardar  Buscar  Cortar  Ejecutar  Ubicación  Deshacer
Salir  Leer fich. Reemplazar  Pegar  Justificar  Ir a línea  Rehacer
```

Figura 33: “Configuración de rsync para backups”

Fuente: “Elaboración propia”

2. Agrega las siguientes líneas al archivo de configuración:

```
uid = nobody
gid = nogroup
use chroot = yes
max connections = 4
pid file = /var/run/rsyncd.pid
log file = /var/log/rsyncd.log
[apache-backup]
path = /var/www/
comment = Backup of Apache files
read only = true
list = false
auth users = backupuser
secrets file = /etc/rsyncd.secrets
exclude = /var/www/htdocs/hidden/*
exclude = /var/www/htdocs/private/*
```

Esto establece la configuración para el demonio rsync, definiendo la carpeta a respaldar ('path'), el usuario de autenticación ('auth users'), y la lista de exclusiones de archivos ('exclude').

3. Crea el archivo '/etc/rsyncd.secrets' con las credenciales de acceso:

```
sudo nano /etc/rsyncd.secrets
```

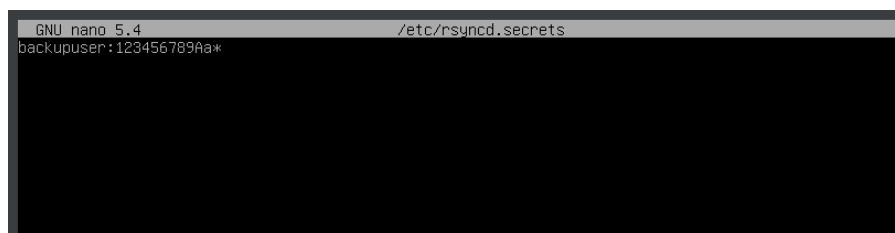


Figura 34: “Credenciales de acceso para rsync” Fuente: “Elaboración propia”

Agrega una línea con las credenciales de acceso en formato ‘usuario:contraseña’:

```
backupuser:contraseña
```

4. Establece los permisos adecuados para el archivo ‘/etc/rsyncd.secrets’:

```
sudo chmod 600 /etc/rsyncd.secrets
sudo chown root:root /etc/rsyncd.secrets
```

5. Inicia el servicio de ‘rsync’:

```
sudo systemctl start rsync
```

6. Para hacer una copia de seguridad, ejecuta el siguiente comando:

```
rsync -avz --delete
--exclude-from=/etc/rsyncd.conf /var/www/
/ruta/del/backup/
```

Esto hará una copia de seguridad de los archivos de ‘/var/www/’ en la carpeta ‘/ruta/del/backup/’. El flag ‘--exclude-from’ especifica los archivos y carpetas que se deben excluir de la copia de seguridad.

7. Para automatizar la copia de seguridad interdiaria, agrega una tarea programada en el archivo de cron. Para hacer esto, ejecuta el siguiente comando:

```
sudo crontab -e
```

Agrega la siguiente línea para ejecutar la copia de seguridad todos los días a las 2:00 AM:

```
0 2 * * * rsync -avz --delete
--exclude-from=/etc/rsyncd.conf /var/www/
/ruta/del/backup/
```

Guarda y cierra el archivo de cron.

Con estos pasos, se debería tener configurado un backup interdiario de los archivos de tu servidor Apache en Debian 11, utilizando el comando ‘rsync’.

3. Lista de comprobación

Tabla 5: “Lista de comprobación recomendaciones NIST 800-123 y NIST 800-44 v2”

Archivar copias de seguridad periódicamente	✓
Mantener una copia autorizada de los sitios web	
Crear una política de copia de seguridad del servidor web	
Realice una copia de seguridad del servidor web de forma diferencial o incremental de forma diaria o semanal	✓
Realice una copia de seguridad completa del servidor web de forma semanal o mensual	✓

III. Conclusiones

En conclusión, el uso de estos estándares puede ayudar a distintos proyectos a mejorar su postura de seguridad de la información, a través de la identificación y mitigación de riesgos, así como la implementación de controles adecuados. Es importante que las organizaciones se mantengan actualizadas en cuanto a los cambios en estos estándares y los adapten a su entorno y necesidades específicas.

También se puede decir que estas normas estandarizadas han sido creadas para la regulación en sistemas operativos y servidores donde se gestionan y administran diversos tipos de datos para así ayudar a mantener con seguridad la información frente a amenazas.

IV. Referencias bibliográficas

Tanenbaum, A. S. (2015). Sistemas operativos modernos (4ª ed.). Pearson.

Apache Software Foundation. (s.f.). Welcome to the Apache HTTP Server Project.
<https://httpd.apache.org/>

Wagner, C., et al. (2012). Professional Apache. John Wiley & Sons.

Gates, B. (1996). Content is king.
<https://www.microsoft.com/en-us/Content/IE/Articles/ContentIsKing.htm>

Halvorson, K. (2012). Content strategy for the web. Pearson Education.
<https://www.pearson.com/us/higher-education/program/Halvorson-Content-Strategy-for-the-Web-2nd-Edition/PGM119575.html>

LinuxLogFiles - Community Help Wiki. (s. f.). Ubuntu documentacion.
<https://help.ubuntu.com/community/LinuxLogFiles>