SEGUNDA ENTREGA - CIBERSEGURIDAD

Bachillerato Tecnológico 3 | Scuola Italiana di Montevideo | Kore

Punto a) Configuración segura del servicio SSH (deshabilitar login como root, cambiar puerto, uso de llaves públicas/privadas)

Procedimiento realizado:

Se implementó una configuración completa de seguridad para el servicio SSH del servidor, incluyendo la deshabilitación del acceso root, configuración de puerto único, y sistema de autenticación basado en llaves públicas/privadas para todos los usuarios técnicos.

1. Verificación y corrección de configuración de puerto

Verificación inicial del estado SSH:

sudo ss -tlnp | grep sshd

```
sigie-admin@sigie-server:~$ sudo nano /etc/ssh/sshd_config
[sudo] password for sigie-admin:
sigie-admin@sigie-server:~$ sudo ss -tlnp | grep sshd
LISTEN 0      4096         0.0.0.0:22        0.0.0.0:*    users:(("sshd",pid=15931,fd=3),("systemd",pid=1,fd=116))

LISTEN 0      4096            [::]:22           [::]:*    users:(("sshd",pid=15931,fd=4),("systemd",pid=1,fd=117))
```

Configuración aplicada manualmente: Se editó el archivo /etc/ssh/sshd_config para:

- Dejar únicamente Port 22 activo
- Mantener PermitRootLogin no
- Verificar AllowUsers sigie-admin sysadmin dbadmin webadmin netadmin

Verificación final de configuración:

sudo grep -E "^(Port|PermitRootLogin|AllowUsers)" /etc/ssh/sshd_config
sudo sshd -t && echo "Sintaxis OK"
sudo systemctl restart ssh
sudo ss -tlnp | grep sshd

```
sigie-admin@sigie-server:~$ sudo grep -E "^(Port|PermitRootLogin|AllowUsers)" /etc/ssh/sshd_config
Port 22
PermitRootLogin no
AllowUsers sigie-admin sysadmin dbadmin webadmin netadmin
```

```
sigie-admin@sigie-server:~$ sudo ss -tlnp | grep sshd
LISTEN 0      4096         0.0.0.0:22        0.0.0.0:*    users:(("sshd",pid=15931,fd=3),("systemd",pid=1,fd=116))

LISTEN 0      4096            [::]:22           [::]:*    users:(("sshd",pid=15931,fd=4),("systemd",pid=1,fd=117))
```

2. Implementación de autenticación por llaves SSH

Generación de llaves para sigie-admin: Generación de llaves para sigie-admin:

ssh-keygen -t ed25519 -C "sysadmin@sigie"

```
sigie-admin@sigie-server:~$ ls -la ~/.ssh/
   cat ~/.ssh/id_ed25519.pub
total 24
drwx------ 2 sigie-admin sigie-admin 4096 sep 19 14:36 .
drwxr-x--- 5 sigie-admin sigie-admin 4096 sep 19 17:09 ..
-rw------- 1 sigie-admin sigie-admin    0 sep 16 23:43 authorized_keys
-rw------- 1 sigie-admin sigie-admin  411 sep 19 14:34 id_ed25519
-rw-r--r-- 1 sigie-admin sigie-admin   96 sep 19 14:34 id_ed25519.pub
-rw------- 1 sigie-admin sigie-admin  978 sep 19 14:36 known_hosts
-rw------- 1 sigie-admin sigie-admin  142 sep 19 14:36 known_hosts.old
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIA4Zvoc3YhA1iffdNw760bv0CO0T5i+WaBdHNyY3vC0z sysadmin@sigie
```

Configuración para usuarios técnicos:

Para cada usuario técnico se generaron llaves desde sus respectivas máquinas cliente usando el mismo proceso.

Verificación de llaves instaladas:

ls -la /home/sysadmin/.ssh/
cat /home/sysadmin/.ssh/authorized_keys

```
sigie-admin@sigie-server:~$ sudo ls -la /home/sysadmin/.ssh/
[sudo] password for sigie-admin:
total 28
drwx------ 2 sysadmin sistema 4096 sep 19 16:31 .
drwxr-x--- 5 sysadmin sistema 4096 sep 19 14:36 ..
-rw------- 1 sysadmin sistema  192 sep 19 16:31 authorized_keys
-rw------- 1 sysadmin sistema  411 sep 19 16:30 id_ed25519
-rw-r--r-- 1 sysadmin sistema   96 sep 19 16:30 id_ed25519.pub
-rw------- 1 sysadmin sistema  978 sep 19 15:34 known_hosts
-rw-r--r-- 1 sysadmin sistema  142 sep 19 15:34 known_hosts.old
sigie-admin@sigie-server:~$ sudo cat /home/sysadmin/.ssh/authorized_keys
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIA4Zvoc3YhA1iffdNw760bv0CO0T5i+WaBdHNyY3vC0z sysadmin@sigie
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIESj6/B+CbzlyC0aeruZX7c7YXV6BOby4tIgCPevaUMD sysadmin@sigie
sigie-admin@sigie-server:~$
```

Distribución de llaves para administración:

```
PUB="$HOME/.ssh/id_ed25519.pub"
USUARIOS=("dbadmin" "webadmin" "netadmin")

for USUARIO in "${USUARIOS[@]}"; do
    echo "--- Configurando acceso sin contraseña para $USUARIO ---"

    sudo mkdir -p /home/$USUARIO/.ssh
    sudo chown $USUARIO:$USUARIO /home/$USUARIO/.ssh
    sudo chmod 700 /home/$USUARIO/.ssh

    sudo cp $PUBLIC_KEY_FILE /home/$USUARIO/.ssh/authorized_keys
    sudo chown $USUARIO:$USUARIO /home/$USUARIO/.ssh/authorized_keys
    sudo chmod 600 /home/$USUARIO/.ssh/authorized_keys

    echo "✓ Configuración completada para $USUARIO"
```

done

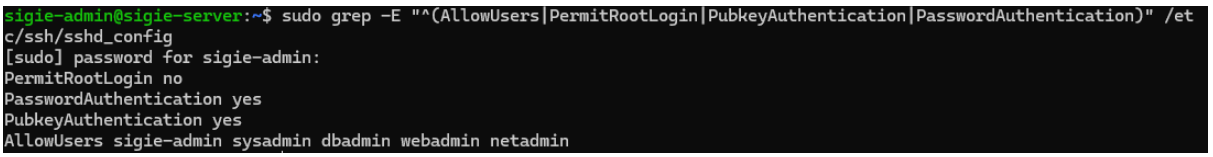
3. Verificación de acceso sin contraseña

Pruebas de acceso sin contraseña:

ssh sysadmin@192.168.2.50
ssh dbadmin@192.168.2.50
ssh webadmin@192.168.2.50
ssh netadmin@192.168.2.50

Verificación final de configuración SSH:

sudo grep -E "^(AllowUsers|PermitRootLogin|PubkeyAuthentication|PasswordAuthentication)" /etc/ssh/sshd_config



Punto b) Comprobación de firewall activo (UFW) y creación de reglas mínimas

Procedimiento realizado:

Se implementó y configuró el firewall UFW (Uncomplicated Firewall) con políticas restrictivas, permitiendo únicamente el tráfico esencial para el funcionamiento del servidor.

1. Instalación y configuración inicial

Comandos ejecutados:

1. Instalación y configuración inicial

Comandos ejecutados:

sudo apt update && sudo apt install ufw -y
sudo ufw default deny incoming
sudo ufw default allow outgoing

2. Creación de reglas específicas

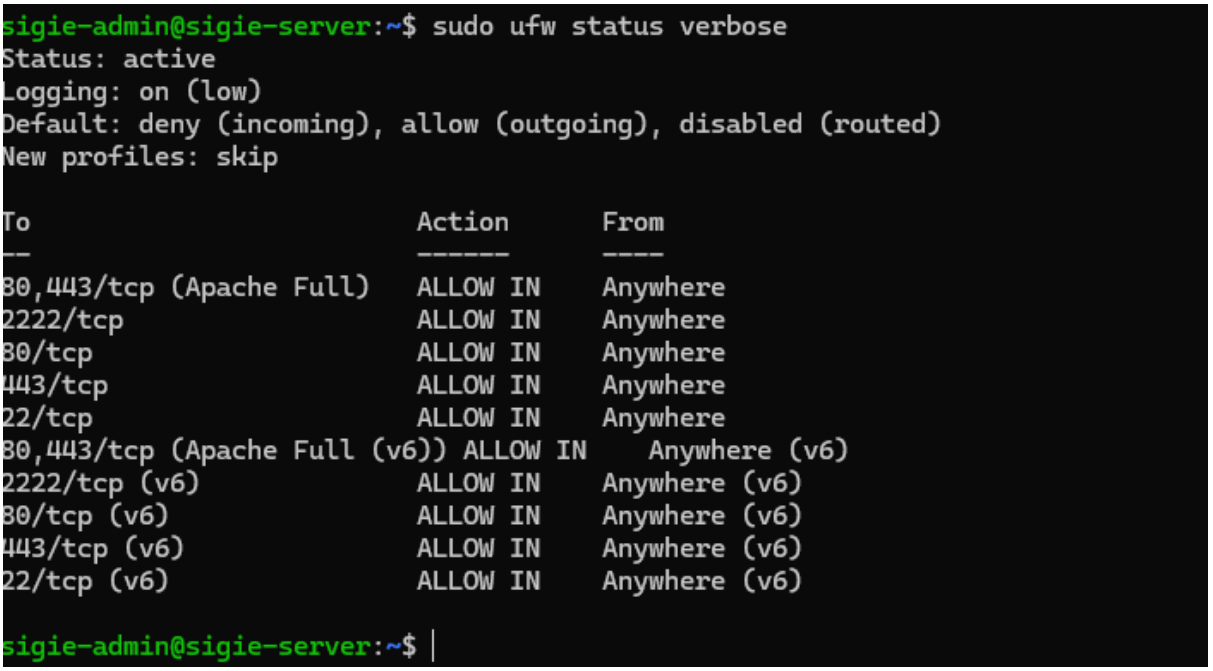
Reglas implementadas:

sudo ufw allow 22/tcp
sudo ufw allow 80/tcp
sudo ufw allow 443/tcp
sudo ufw enable

3. Verificación de funcionamiento

Estado del firewall:

sudo ufw status verbose
sudo systemctl status ufw --no-pager



Prueba de conectividad post-firewall:

ssh sysadmin@192.168.2.50

conecto sin problemas.

Punto c) Implementación de control básico de acceso a los scripts mediante permisos (chmod, chown)

Procedimiento realizado:

Se aplicaron permisos restrictivos a todos los scripts de administración del sistema, asegurando que solo el usuario propietario tenga acceso de lectura, escritura y ejecución.

1. Identificación de scripts existentes

ls -l /home/sigie-admin/*.sh

```
sigie-admin@sigie-server:~$ ls -l /home/sigie-admin/*.sh
-rwx------ 1 sigie-admin sigie-admin 1848 sep 17 02:27 /home/sigie-admin/admin_menu.sh
-rwx------ 1 sigie-admin sigie-admin  626 sep 18 12:18 /home/sigie-admin/backup_full.sh
-rwx------ 1 sigie-admin sigie-admin  843 sep 17 02:13 /home/sigie-admin/create_roles.sh
-rwx------ 1 sigie-admin sigie-admin  845 sep 17 01:49 /home/sigie-admin/create_users_groups.sh
-rwx------ 1 sigie-admin sigie-admin  746 sep 19 17:17 /home/sigie-admin/db_test.sh
-rwx------ 1 sigie-admin sigie-admin 1029 sep 17 16:27 /home/sigie-admin/menu_principal.sh
-rwx------ 1 sigie-admin sigie-admin 2320 sep 17 02:15 /home/sigie-admin/menu.sh
-rwx------ 1 sigie-admin sigie-admin  707 sep 17 01:53 /home/sigie-admin/menu_user.sh
-rwx------ 1 sigie-admin sigie-admin 1054 sep 18 12:22 /home/sigie-admin/restore_full.sh
```

2. Aplicación de permisos restrictivos

Se aplicaron permisos restrictivos usando comandos en bloque para todos los scripts de administración:

sudo chown sigie-admin:sigie-admin /home/sigie-admin/*.sh
sudo chmod 700 /home/sigie-admin/*.sh

3. Verificación de permisos aplicados

ls -l /home/sigie-admin/*.sh

```
sigie-admin@sigie-server:~$ ls -l /home/sigie-admin/*.sh
-rwx------ 1 sigie-admin sigie-admin 1848 sep 17 02:27 /home/sigie-admin/admin_menu.sh
-rwx------ 1 sigie-admin sigie-admin  626 sep 18 12:18 /home/sigie-admin/backup_full.sh
-rwx------ 1 sigie-admin sigie-admin  843 sep 17 02:13 /home/sigie-admin/create_roles.sh
-rwx------ 1 sigie-admin sigie-admin  845 sep 17 01:49 /home/sigie-admin/create_users_groups.sh
-rwx------ 1 sigie-admin sigie-admin  746 sep 19 17:17 /home/sigie-admin/db_test.sh
-rwx------ 1 sigie-admin sigie-admin 1029 sep 17 16:27 /home/sigie-admin/menu_principal.sh
-rwx------ 1 sigie-admin sigie-admin 2320 sep 17 02:15 /home/sigie-admin/menu.sh
-rwx------ 1 sigie-admin sigie-admin  707 sep 17 01:53 /home/sigie-admin/menu_user.sh
-rwx------ 1 sigie-admin sigie-admin 1054 sep 18 12:22 /home/sigie-admin/restore_full.sh
```

Significado de los permisos aplicados:

- -rwx------: Solo el propietario (sigie-admin) tiene permisos completos
- r (read): Permiso de lectura
- w (write): Permiso de escritura
- x (execute): Permiso de ejecución
- Los 6 guiones restantes indican que grupo y otros usuarios no tienen ningún permiso

Punto d) Verificación de integridad básica de scripts usando SHA256SUM

Procedimiento realizado:

Se implementó un sistema de verificación de integridad utilizando hashes SHA256 para detectar modificaciones no autorizadas en los scripts críticos del sistema.

1. Generación de hashes de integridad

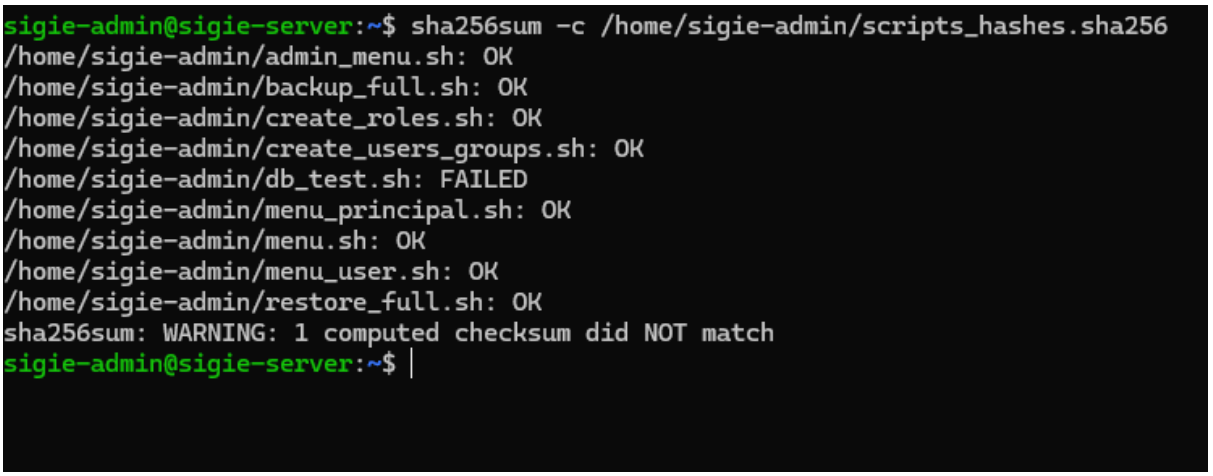
sha256sum /home/sigie-admin/*.sh > /home/sigie-admin/scripts_hashes.sha256
sudo chmod 600 /home/sigie-admin/scripts_hashes.sha256

2. Verificación del archivo de hashes generado

ls -l /home/sigie-admin/scripts_hashes.sha256
cat /home/sigie-admin/scripts_hashes.sha256

3. Verificación de integridad

sha256sum -c /home/sigie-admin/scripts_hashes.sha256



Punto e) Introducción al uso de fail2ban para protección de servicios como SSH

Procedimiento realizado:

Se instaló y configuró fail2ban como sistema de prevención de intrusiones, específicamente para proteger el servicio SSH contra ataques de fuerza bruta.

1. Instalación de fail2ban

sudo apt update && sudo apt install fail2ban -y

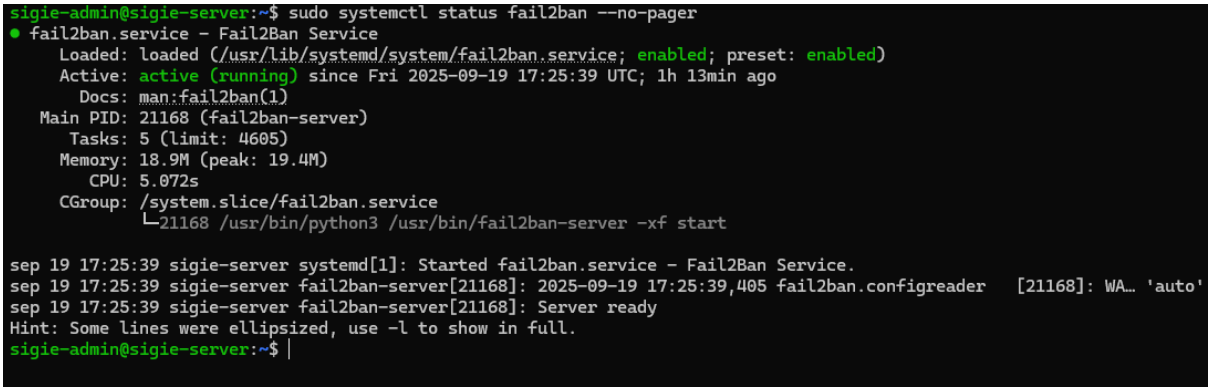
2. Configuración personalizada

Se creó una configuración optimizada eliminando la configuración problemática por defecto y estableciendo parámetros específicos para el entorno:

- **Configuración aplicada manualmente:**
  - bantime = 3600 (1 hora de bloqueo)
  - findtime = 600 (10 minutos de ventana)
  - maxretry = 3 (máximo 3 intentos)
  - enabled = true para jail sshd
  - port = 22 y logpath = /var/log/auth.log

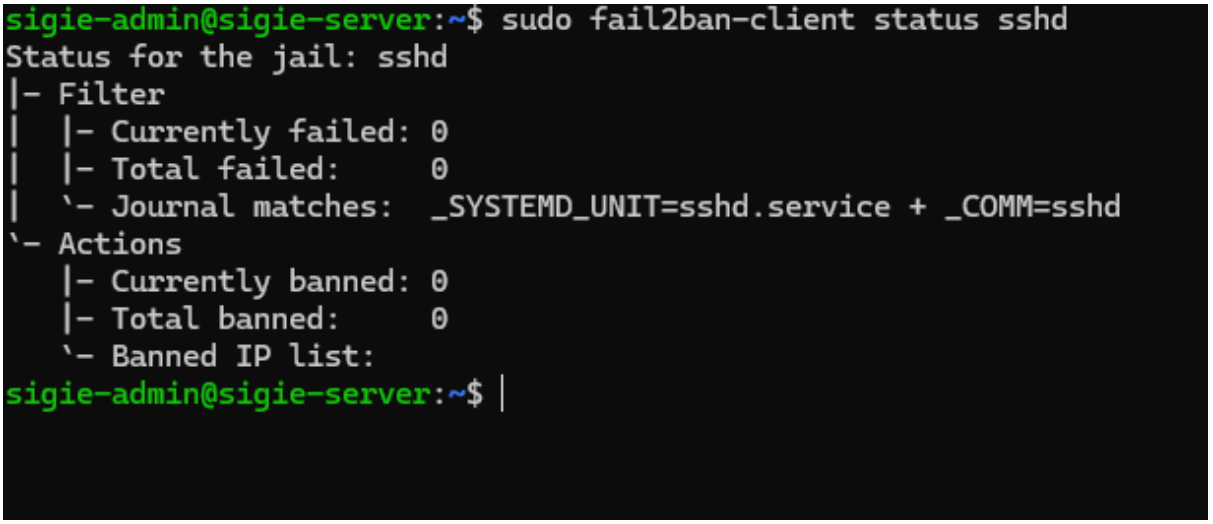
3. Activación y verificación del servicio

sudo systemctl restart fail2ban
sudo systemctl status fail2ban --no-pager



4. Verificación de protección SSH

sudo fail2ban-client status sshd



Parámetros de protección configurados:

- **maxretry = 3**: Máximo 3 intentos fallidos antes del bloqueo
- **bantime = 3600**: Bloqueo por 1 hora (3600 segundos)
- **findtime = 600**: Ventana de 10 minutos para contar intentos
- **logpath = /var/log/auth.log**: Archivo de log monitorizado

5. Comandos administrativos útiles

Para desbanear una IP específica:

sudo fail2ban-client set sshd unbanip <IP-ADDRESS>

Para ver todas las IPs baneadas:

sudo fail2ban-client status sshd

Para ver logs de fail2ban:

sudo tail -f /var/log/fail2ban.log

Verificación General de Seguridad

Checklist de verificación completa:

Se implementó un comando de verificación integral para comprobar el estado de todos los puntos de seguridad implementados:

echo "=== a) SSH ===" && sudo grep -E "^(AllowUsers|PermitRootLogin)" /etc/ssh/sshd_config; echo "=== b) UFW ===" && sudo ufw status | head -n 2; echo "=== c) Permisos ===" && ls -l /home/sigie-admin/*.sh | head -n 3; echo "=== d) Hashes ===" && sha256sum -c /home/sigie-admin/scripts_hashes.sha256 2>/dev/null | head -n 2; echo "=== e) Fail2ban ===" && sudo systemctl is-active fail2ban && sudo fail2ban-client status sshd | grep -E "(Status|banned)"

Resultado de la verificación:

```
sigie-admin@sigie-server:~$ echo "=== a) SSH ===" && sudo grep -E "^(AllowUsers|PermitRootLogin)" /etc/ssh/sshd_config;
echo "=== b) UFW ===" && sudo ufw status | head -n 2; echo "=== c) Permisos ===" && ls -l /home/sigie-admin/*.sh | head
-n 3; echo "=== d) Hashes ===" && sha256sum -c /home/sigie-admin/scripts_hashes.sha256 2>/dev/null | head -n 2; echo "==
= e) Fail2ban ===" && sudo systemctl is-active fail2ban && sudo fail2ban-client status sshd | grep -E "(Status|banned)"
=== a) SSH ===
PermitRootLogin no
AllowUsers sigie-admin sysadmin dbadmin webadmin netadmin
=== b) UFW ===
Status: active

=== c) Permisos ===
-rwx------ 1 sigie-admin sigie-admin 1848 sep 17 02:27 /home/sigie-admin/admin_menu.sh
-rwx------ 1 sigie-admin sigie-admin  626 sep 18 12:18 /home/sigie-admin/backup_full.sh
-rwx------ 1 sigie-admin sigie-admin  843 sep 17 02:13 /home/sigie-admin/create_roles.sh
=== d) Hashes ===
/home/sigie-admin/admin_menu.sh: OK
/home/sigie-admin/backup_full.sh: OK
=== e) Fail2ban ===
active
Status for the jail: sshd
   |- Currently banned: 0
   |- Total banned:     0
sigie-admin@sigie-server:~$
```

Resumen de la Segunda Entrega

Estado final del sistema de ciberseguridad:

Medidas de seguridad implementadas:

1. **SSH Seguro**:
    - Acceso root completamente deshabilitado
    - Autenticación basada en llaves públicas/privadas para todos los usuarios técnicos
    - Lista restrictiva de usuarios autorizados
    - Puerto único (22) configurado
2. **Firewall UFW**:
    - Política por defecto: denegar todo tráfico entrante

    - Reglas específicas solo para servicios esenciales (SSH:22, HTTP:80, HTTPS:443)
    - Estado activo y funcional verificado
3. **Control de acceso a scripts**:
    - Permisos 700 (-rwx------) aplicados a todos los scripts críticos
    - Solo el administrador (sigie-admin) puede acceder a los archivos
    - Propietario y grupo correctamente asignados
4. **Verificación de integridad**:
    - Sistema de hashes SHA256 implementado
    - Archivo único de verificación protegido con permisos 600
    - Capacidad de detección de modificaciones no autorizadas
5. **Protección contra ataques de fuerza bruta**:
    - Fail2ban configurado y activo
    - Monitoreo automático del servicio SSH
    - Bloqueo automático después de 3 intentos fallidos por 1 hora

Configuraciones de red y acceso:

- **Servidor**: 192.168.2.50/24
- **Acceso SSH**: Solo mediante llaves privadas
- **Usuarios autorizados**: sigie-admin, sysadmin, dbadmin, webadmin, netadmin
- **Servicios protegidos**: SSH (22), HTTP (80), HTTPS (443)

Automatización y monitoreo:

- **Verificación de integridad**: Disponible mediante comando sha256sum -c
- **Monitoreo de intentos de acceso**: fail2ban registra todos los intentos
- **Logs de seguridad**: Almacenados en /var/log/auth.log y /var/log/fail2ban.log

La segunda entrega proporciona una capa robusta de seguridad para el servidor SIGIE, implementando las mejores prácticas de ciberseguridad en un entorno educativo, con múltiples líneas de defensa contra amenazas comunes como ataques de fuerza bruta, acceso no autorizado y modificación maliciosa de archivos críticos.