LXC + SFTP chroot para hosting web

Objetivo

Entorno seguro para hosting web con:

- Acceso SFTP confinado (chroot)
- Separación clara entre sistema, chroot y contenido
- Buenas prácticas de permisos
- Verificaciones integradas

1. Instalación de paquetes esenciales

```
apt update
apt install openssh-server nginx acl -y
```

Nota: ACL es necesario para permisos granulares.

2. Creación de usuario y estructura de directorios

```
# Usuario sin shell, solo SFTP
useradd -m -s /usr/sbin/nologin -G www-data paco
echo "paco:pacoPass" | chpasswd

# Estructura de directorios
mkdir -p /var/www/vb.red.lan/public_html
mkdir -p /var/www/vb.red.lan/uploads
```

3. Configuración de permisos

```
# 1. Chroot: propiedad root, solo lectura para otros
chown root:root /var/www
chmod 755 /var/www

# 2. Directorio del sitio: propiedad www-data, accesible para grupo
chown -R www-data:www-data /var/www/vb.red.lan
chmod 755 /var/www/vb.red.lan

# 3. Contenido público: permisos estándar
chmod 755 /var/www/vb.red.lan/public_html

# 4. Verificar soporte ACL (Mejora #1)
touch /var/www/vb.red.lan/test_acl
setfacl -m u:paco:rwx /var/www/vb.red.lan/test_acl 2>/dev/null
if [ $? -eq 0 ]; then
    echo "ACL soportado - configurando permisos granulares"
    # Directorio de uploads con permisos especiales via ACL
    setfacl -R -m u:paco:rwx /var/www/vb.red.lan/uploads
    setfacl -R -d -m u:paco:rwx /var/www/vb.red.lan/uploads
    rm -f /var/www/vb.red.lan/test_acl
else
    echo "ACL NO soportado - usando permisos tradicionales"
    chmod 775 /var/www/vb.red.lan/uploads
fi
```

```
# 5. Consideraciones para CMS y aplicaciones web (añadir según necesidades)
echo ""
echo "=== PERMISOS PARA CMS Y APLICACIONES WEB ==="
echo "Para CMS como WordPress, Laravel u otros que requieren escritura en
directorios específicos,"
echo "debes aplicar permisos similares a los siguientes directorios:"
echo ""
echo "Ejemplos comunes:"
echo "# WordPress: wp-content/uploads, wp-content/cache"
echo "# Laravel: storage, bootstrap/cache"
echo "# Symfony: var/cache, var/log"
echo ""
echo "Puedes configurarlos con:"
echo "# Opción A: Usando ACL (si está soportado)"
echo "setfacl -R -m u:paco:rwx
/var/www/vb.red.lan/public_html/wp-content/uploads"
echo "setfacl -R -d -m u:paco:rwx
/var/www/vb.red.lan/public_html/wp-content/uploads"
echo ""
echo "# Opción B: Permisos tradicionales"
echo "chmod -R 775 /var/www/vb.red.lan/public_html/wp-content/uploads"
echo "chown -R www-data:www-data
/var/www/vb.red.lan/public_html/wp-content/uploads"
```

4. Configuración SSH para SFTP chroot

Editar `/etc/ssh/sshd_config`:

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

Configuración exacta:

```
# COMENTAR línea antigua (crítico)
#Subsystem sftp /usr/lib/openssh/sftp-server

# Usar internal-sftp
Subsystem sftp internal-sftp

Match User paco
    ChrootDirectory /var/www
    ForceCommand internal-sftp
    AllowTcpForwarding no
    AllowAgentForwarding no
    X11Forwarding no
    PermitTunnel no
    PermitTTY no
```

5. Directorio de separación de privilegios

```
mkdir -p /run/sshd
chmod 0755 /run/sshd
echo "d /run/sshd 0755 root root" > /etc/tmpfiles.d/sshd.conf
```

6. Validación y reinicio SSH

```
sshd -t
systemctl restart ssh
```

7. Verificación de conectividad SFTP

Prueba exhaustiva desde otra terminal:

```
# 1. Conexión básica
echo "Probar conexión SFTP (contraseña: pacoPass):"
sftp -P 22 paco@localhost

# 2. Comandos a ejecutar DENTRO de SFTP:
#    ls                    # Debe ver solo 'vb.red.lan'
#    cd vb.red.lan         # Debe funcionar
#    ls                    # Debe ver 'public_html' y 'uploads'
#    mkdir test_sftp       # Debe FALLAR (solo uploads es escribible)
#    cd uploads
#    mkdir test_ok         # Debe FUNCIONAR (si ACL configurado)
#    put /etc/hostname .   # Probar subida de archivo
#    exit

# 3. Verificación desde el sistema:
sudo -u www-data ls -la /var/www/vb.red.lan/uploads/
```

Resultados esperados:

- Usuario ve solo `/var/www` como raíz
- Puede navegar a `vb.red.lan/public_html` (solo lectura)
- Puede escribir SOLO en `uploads/` (si ACL funciona)
- No puede salir del chroot

8. Configuración de virtual host Nginx

```
server {
    listen 80;
    server_name vb.red.lan;
    root /var/www/vb.red.lan/public_html;
    index index.html index.php;

    location / {
        try_files $uri $uri/ =404;
    }

    # Opcional: PHP (descomentar si se instala php-fpm)
    # location ~ \.php$ {
    #     include snippets/fastcgi-php.conf;
    #     fastcgi_pass unix:/run/php/php-fpm.sock;
    # }
}
```

Habilitar sitio:

```
ln -s /etc/nginx/sites-available/vb.red.lan /etc/nginx/sites-enabled/
nginx -t
systemctl reload nginx
```

9. Otras mejoras profesionales

9.1. Autenticación por clave SSH (recomendada)

```
# Crear directorio .ssh dentro del chroot
mkdir -p /var/www/.ssh
echo "ssh-ed25519 AAAAC3... tu-clave" > /var/www/.ssh/authorized_keys

# Permisos estrictos
chown -R paco:www-data /var/www/.ssh
chmod 700 /var/www/.ssh
chmod 600 /var/www/.ssh/authorized_keys
```

Nota: En `sshd_config` no se necesita `AuthorizedKeysFile`, el SSH reconoce la ruta dentro del chroot.

9.2. Fail2ban para protección web

```
apt install fail2ban -y
cat > /etc/fail2ban/jail.d/nginx.conf << EOF
[nginx-http-auth]
enabled = true
port    = http,https
filter  = nginx-http-auth
logpath = /var/log/nginx/error.log
maxretry = 3
EOF
systemctl restart fail2ban
```

9.3. Configuración para PHP/Laravel

```
apt install php-fpm php-mysql php-curl php-gd -y
chown -R www-data:www-data /var/www/vb.red.lan/public_html
find /var/www/vb.red.lan/public_html -type d -exec chmod 755 {} \;
find /var/www/vb.red.lan/public_html -type f -exec chmod 644 {} \;

# Para Laravel:
# chmod -R 775 /var/www/vb.red.lan/public_html/storage
# chmod -R 775 /var/www/vb.red.lan/public_html/bootstrap/cache
```

9.4. Monitoreo básico

```
# Monitoreo de intentos de acceso SFTP y SSH
echo "=== MONITOREO BÁSICO ==="
echo "Los logs de acceso SFTP/SSH se registran en:"
echo "1. /var/log/auth.log (intentos de autenticación)"
echo "2. /var/log/syslog (eventos generales del sistema)"
echo ""
echo "Comandos útiles para monitoreo:"
echo "# Intentos fallidos de autenticación:"
echo "grep 'Failed password\|Invalid user' /var/log/auth.log | tail -10"
echo ""
echo "# Conexiones SFTP exitosas:"
echo "grep 'session opened for user paco' /var/log/auth.log | tail -5"
echo ""
echo "# Intentos de acceso no autorizados:"
echo "grep 'Connection closed by authenticating user' /var/log/auth.log"
```

10. Comprobación final completa

```
echo "=== VERIFICACIÓN FINAL ==="
echo "1. Usuario y grupos:"
id paco
echo -e "\n2. Permisos del chroot:"
ls -ld /var/www/
echo -e "\n3. Contenido del sitio:"
ls -la /var/www/vb.red.lan/
echo -e "\n4. Estado SSH:"
systemctl status ssh --no-pager -l
echo -e "\n5. Prueba de escritura web:"
sudo -u www-data touch /var/www/vb.red.lan/public_html/test.html && echo "✓
Escritura web OK"
```

11. Puntos críticos que suelen fallar

- **Chroot no funciona / acceso denegado**
  - Verifica que `/var/www` sea propiedad `root:root` con `chmod 755`.
  - Asegúrate de que la línea antigua `Subsystem sftp /usr/lib/openssh/sftp-server` esté comentada.
- **Autenticación por clave falla**
  - La carpeta `.ssh` debe existir **dentro del chroot** (`/var/www/.ssh`) y con permisos `700`.
  - El archivo `authorized_keys` debe tener permisos `600` y ser propiedad de `paco:www-data`.
- **Usuario no puede escribir en uploads**
  - Si usas ACL, verifica con `getfacl /var/www/vb.red.lan/uploads`.
  - Si no hay ACL, comprueba permisos tradicionales (`chmod 775`) y pertenencia a grupo `www-data`.
- **SSH se niega a iniciar después de cambios**
  - Ejecuta siempre `sshd -t` para validar la configuración antes de reiniciar el servicio.
  - Revisa `journalctl -u ssh -n 20` para errores inmediatos.
- **Problemas con PHP o Nginx**
  - Revisa que el usuario `www-data` tenga propiedad y permisos correctos en los directorios del sitio.
  - Verifica el socket de PHP-FPM y los logs de Nginx.