

Configuración del servicio: OpenSSH

Introducción

OpenSSH es una implementación de protocolo SSH de OpenBSD. SSH reemplaza rlogin y rsh, para proporcionar comunicaciones encriptadas seguras entre dos hosts en una red insegura. Las conexiones X11 y los puertos arbitrarios TCP/IP pueden adelantarse en un canal seguro. La autenticación de la llave pública puede ser usada para el acceso a los servidores "passwordless".

Securizar OpenSSH

El objetivo de este módulo es proporcionar comunicaciones seguras via 'ssh' (secure shell) y 'sftp' (secure ftp) para los usuarios del sistema.

Modificaciones en el fichero /etc/ssh/sshd_config

Es el fichero de configuración del servidos OpenSSH. En el se detallan todas las opciones que permiten modificar el comportamiento de servidor.

Hemos realizado las siguientes modificaciones

1. Habilitamos sftp Server

SFTP-server es un programa que 'habla' el lado del servidor del protocolo SFTP. Es un servidor seguro de ftp, por lo que la comunicación se realiza de forma cifrada.

Normalmente SFTP-server no está pensado para ejecutarse directamente, sino desde sshd mediante la opción 'Subsystem'. Para ello se ha activado dicho subsistema en el fichero /etc/ssh/sshd_config:

```
Subsystem      sftp          /usr/libexec/openssh/sftp-server
```

Dado que el servicio SFTP es utilizado por varios usuarios para actualizar los ficheros de las fotos en el servidor, es necesario que dichos ficheros dispongan de los permisos necesarios. Con ese objetivo, se debe crear el siguiente script que se ejecutará antes de lanzar el servidor de SFTP:

```
#!/bin/bash
umask 000
exec /usr/libexec/openssh/sftp-server.real "$@"
```

Para ello se debe renombrar el fichero `/usr/libexec/openssh/sftp-server` por `/usr/libexec/openssh/sftp-server-real` y guardar el script anterior con el nombre `/usr/libexec/openssh/sftp-server`

Esta modificación dificultará la actualización del paquete OpenSSH. Debido a que cualquier actualización sobre éste reemplazaría los archivos ejecutables (incluido nuestra versión modificada de *sftp*) se deben realizar actualizaciones de forma manual..

2. Creación de Usuario para el Acceso SSh

Se ha creado un nuevo usuario *ssh* con el objetivo que éste sea el único usuario que pueda abrir una shell en el sistema. También se ha denegado el acceso al usuario privilegiado *root* para aumentar la seguridad. Esta medida evita que alguien pudiera encontrar la contraseña de *root* mediante técnicas de *password cracking* sobre el servicio OpenSSH

Usamos para este fin el siguiente comando

```
groupadd ssh
useradd -g ssh -d /home/ssh ssh -s /bin/bash
```

Le asignamos una contraseña adecuada

```
passwd ssh
```

Para conectarse via SSH al servidor, se utiliza (si nos encontramos en modo comandos desde linux)

```
ssh -l <login_name> <hostname>
```

Si nos queremos conectar desde windows podemos usar clientes de SSH como PuTTY.

En nuestro caso concreto, sólo podemos entrar con el usuario ssh

```
ssh -l ssh xxxxx.iddover.net
```

De esta forma si se necesita realizar cualquier operación con privilegios del usuario *'root'*, deberemos entrar en el sistema con el usuario *ssh* y una vez dentro cambiar a *'root'*:

```
su -
```

La opción del guión (-) indica que han de ejecutarse los scripts de inicio y las variables propias del usuario *root*

3. Limitar Usuarios

Se ha limitado el acceso a los servicios SSh, permitiendo que únicamente dispongan de acceso a los mismos el usuario ssh y los usuarios de favshare. Esta medida evita el riesgo de intrusión por contraseñas por defecto y las técnicas de *password cracking*

```
AllowUsers ssh
```

Los usuarios definidos sólo tienen acceso al servicio SFTP (secure ftp) y el usuario ssh dispone además de acceso a una shell en el sistema.

Para conectarse via sftp al servidor. se utiliza (si nos encontramos en modo comandos desde linux):

```
sftp <login_name>@<host_name>
```

Si nos queremos conectar desde windows podemos utilizar clientes de FTP como CuteFTP Pro.

Nota: También es recomendado limitar el acceso a aquellos usuarios que no deseamos se conecten remotamente (ssh, telnet, et.) con una shell inoperativa, como por ejemplo /sbin/nologin

Modificaciones en el fichero /etc/passwd

Es uno de los ficheros críticos del sistema. En este fichero se almacenan los usuarios y sus características (nombre y descripción, contraseña, directorio de entrada, shell utilizada, etc.)

1. Creación de una Shell Específica

Para limitar el acceso de los usuarios al servicio SFTP, se ha creado una shell específica para ellos *dummy_shell* que en realidad es una shell simulada y que, por lo tanto, no la pueden abrir:

Para elaborar la shell (originalmente en c) podemos usar el compilador *gcc*, mediante el comando siguiente

```
gcc -o dummy_shell dummy_shell.c
```

Ubicaremos la shell en el fichero /usr/dummy_shell

2. Modificaciones del fichero /etc/passwd

Para aquellos usuarios que deseamos tengan acceso a el servicio sftp debemos notificarlo en el fichero /etc/passwd, remplazando la shell actual (normalmente /bin/nologin) por la nueva shell que hemos creado /bin/dummy_shell

```
ssh:x:3825:2975::/home/ssh:/bin/dummy_shell
```

3. Modificaciones en el fichero /etc/shells

Es un fichero que contiene las shells válidas del sistema. Este fichero sirve como prevención ante usuarios que modifique una shell en /etc/passwd, ya que en caso de no encontrarse en este fichero no será considerada como válida

Para que los cambios en la shell de cada usuario tengan efecto debemos añadir la nueva shell en este archivo. A continuación se muestra el contenido de este fichero:

```
/bin/sh  
/bin/bash  
/sbin/nologin  
/bin/tcsh  
/bin/csh  
/bin/ksh  
/bin/dummy_shell
```

Modificaciones de permisos para sshd

SSH es el script responsable de arrancar y parar el demonio de OpenSSH en el sistema.

1. Limitar el acceso y ejecución a root

Para hacer el script ejecutable y cambiar los permisos por defecto, se ha ejecutado:

```
chmod 700 /etc/rc.d/init.d/sshd
```

¿Que es iddover.net?

Iddover Hosting es una iniciativa española en el sector de servidores dedicados. Nace en contraposición de los servicios actuales de hosting . Pretende ofrecer un hosting de alta calidad gestionable a través de un potente panel propio que se adecua a los requisitos de seguridad y que proporciona al cliente potentes herramientas al alcance de un solo click

Desde 69€ dispondrás de un servidor dedicado administrado auditado con todas las medidas de seguridad ya aplicadas.

<http://www.iddover.net>