

**Autores: [Francisco Isgleas Guzman](#) & [Maikel Stinga Ruiz](#)**

© 1999-2009 Linux Para Todos. Algunos Derechos Reservados 2007 Factor Evolución SA de CV. Usted es libre de copiar, distribuir y comunicar públicamente la obra y hacer obras derivadas bajo las condiciones siguientes: a) Debe reconocer y citar al autor original. b) No puede utilizar esta obra para fines comerciales. c) Si altera o transforma esta obra, o genera una obra derivada, sólo puede distribuir la obra generada bajo una licencia idéntica a ésta. Al reutilizar o distribuir la obra, tiene que dejar bien claro los términos de la licencia de esta obra. Alguna de estas condiciones puede no aplicarse si se obtiene el permiso del titular de los derechos de autor. Los derechos derivados de usos legítimos u otras limitaciones no se ven afectados por lo anterior. Licencia completa en castellano. La información contenida en este documento y los derivados de éste se proporcionan tal cual son y los autores no asumirán responsabilidad alguna si el usuario o lector hace mal uso de éstos.

## **Introducción.**

### **Acerca de SSH.**

SSH (Secure Shell) es un conjunto de estándares y protocolo de red que permite establecer una comunicación a través de un canal seguro entre un cliente local y un servidor remoto. Utiliza una clave pública cifrada para autenticar el servidor remoto y, opcionalmente, permitir al servidor remoto autenticar al usuario. SSH provee confidencialidad e integridad en la transferencia de los datos utilizando criptografía y MAC (Message Authentication Codes, o Códigos de Autenticación de Mensaje). De modo predeterminado, escucha peticiones a través del puerto 22 por TCP. Acerca de SFTP.

SFTP (SSH File Transfer Protocol) es un protocolo que provee funcionalidad de transferencia y manipulación de ficheros a través de un flujo confiable de datos. Comúnmente se utiliza con SSH para proveer a éste de transferencia segura de ficheros. Acerca de SCP.

SCP (Secure Copy, o Copia Segura) es una protocolo seguro para transferir ficheros entre un anfitrión local y otro remoto, a través de SSH. Básicamente, es idéntico a RCP (Remote Copy, o Copia Remota), con la diferencia de que los datos son cifrados durante la transferencia para evitar la extracción potencial de información a través de programas de captura de las tramas de red (packet sniffers). SCP solo implementa la transferencia de ficheros, pues la autenticación requerida es realizada a través de SSH. Acerca de OpenSSH.

OpenSSH (Open Secure Shell) es una alternativa de código abierto, con licencia BSD, hacia la implementación propietaria y de código cerrado SSH creada por Tatu Ylönen. OpenSSH es un proyecto creado por el equipo de desarrollo de OpenBSD y actualmente dirigido por Theo de Raadt. Se considera es más segura que su contraparte propietaria debido a la constante auditoría que se realiza sobre el código fuente por parte de una gran comunidad de desarrolladores, una ventaja que brinda al tratarse de un proyecto de fuente abierta.

OpenSSH incluye servicio y clientes para los protocolos SSH, SFTP y SCP.

URL: <http://www.openssh.org/>.

Sustento lógico necesario.

- openssh-3.5p1-6
- openssh-clients-3.5p1-6
- openssh-server-3.5p1-6

Antes de continuar verifique siempre la existencia de posibles actualizaciones de seguridad:

```
$yum -y install openssh openssh-server openssh-clients
```

### **Ficheros de configuración.**

```
$/etc/ssh/sshd_config
```

 Fichero central de configuración del servicio SSH.

### **Procedimientos.**

Edite `/etc/ssh/sshd_config`. A continuación se analizarán los parámetros a modificar.

### **Parámetro Port.**

Una forma de elevar considerablemente la seguridad al servicio de SSH, es cambiar el número de puerto utilizado por el servicio, por otro que solo conozca el administrador del sistema. A este tipo de técnicas se les conoce como Seguridad por Oscuridad. La mayoría de los delincuentes informáticos utiliza guiones que buscan servidores que respondan a peticiones a través del puerto 22. Cambiar de puerto el servicio de SSH disminuye considerablemente la posibilidad de una intrusión a través de este servicio. Port 22

SSH trabaja a través del puerto 22 por TCP. Puede elegirse cualquier otro puerto entre el 1025 y 65535. ejemplo:Port 52341

### **Parámetro ListenAddress.**

Por defecto, el servicio de SSH responderá peticiones a través de todas las interfaces del sistema. En algunos casos es posible que no se desee esto y se prefiera limitar el acceso sólo a través de una interfaz a la que sólo se pueda acceder desde la red local. Para tal fin puede establecerse lo siguiente, considerando que el servidor a configurar posee la IP 192.168.1.254:ListenAddress 192.168.1.254

### **Parámetro PermitRootLogin.**

Establece si se va a permitir el acceso directo del usuario root al servidor SSH. Si se va a permitir el acceso hacia el servidor desde redes públicas, resultará prudente utilizar este parámetro con el valor no. PermitRootLogin no

### **Parámetro X11Forwarding.**

Establece si se permite o no la ejecución remota de aplicaciones gráficas. Si se va a acceder hacia el servidor desde red local, este parámetro puede quedarse con el valor yes. Si se va a permitir el

acceso hacia el servidor desde redes públicas, resultará prudente utilizar este parámetro con el valor `no.X11Forwarding yes`

### **Parámetro AllowUsers.**

Permite restringir el acceso por usuario y, opcionalmente, anfitrión desde el cual pueden hacerlo. El siguiente ejemplo restringe el acceso hacia el servidor SSH para que solo puedan hacerlo los usuarios fulano y mengano, desde cualquier anfitrión. `AllowUsers fulano mengano`

Permite restringir el acceso por usuario y, opcionalmente, anfitrión desde el cual pueden hacerlo. El siguiente ejemplo restringe el acceso hacia el servidor SSH para que solo puedan hacerlo los usuarios fulano y mengano, solamente desde los anfitriones 10.1.1.1 y 10.2.2.1. `AllowUsers fulano@10.1.1.1 mengano@10.1.1.1 fulano@10.2.2.1 mengano@10.2.2.1`

### **Aplicando los cambios.**

El servicio de SSH puede iniciar, detenerse o reiniciar a través de un guión similar a los del resto del sistema. De tal modo, podrá iniciar, detenerse o reiniciar a través del mandato `service` y añadirse al arranque del sistema en un nivel o niveles de corrida en particular con el mandato `chkconfig`.

Para ejecutar por primera vez el servicio, utilice: `service sshd start`

Para hacer que los cambios hechos a la configuración surtan efecto, utilice: `service sshd restart`

Para detener el servicio, utilice: `service sshd stop`

De forma predeterminada, el servicio SSH está incluido en todos los niveles de corrida con servicio de red. Para desactivar el servicio Sshd de los niveles de corrida 2, 3, 4 y 5, ejecute: `chkconfig --level 2345 sshd off`

### **Probando OpenSSH.**

Acceso a través de intérprete de mandatos.

Para acceder a través de intérprete de mandatos hacia el servidor, basta con ejecutar desde el sistema cliente el mandato `ssh` definiendo el usuario a utilizar y el servidor al cual conectar: `ssh usuario@servidor`

Para acceder hacia un puerto en particular, se utiliza el parámetro `-p`. En el siguiente ejemplo, utilizando la cuenta del usuario juan, se intentará acceder hacia el servidor con dirección IP 192.168.0.99, el cual tiene un servicio de SSH que responde peticiones a través del puerto 52341. `ssh -p 52341 juan@192.168.0.99`

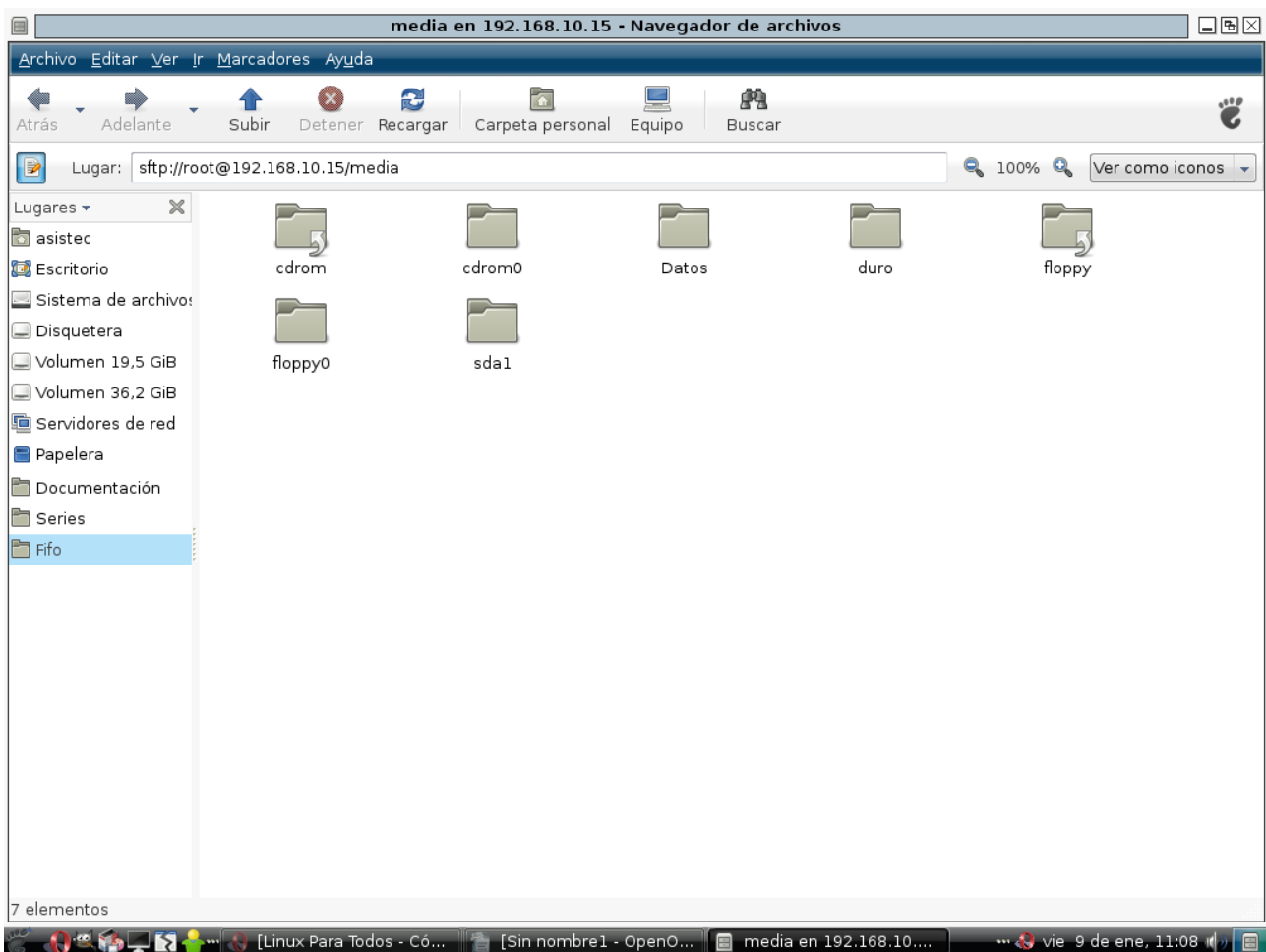
## Transferencia de ficheros a través de SFTP.

Para acceder a través de SFTP hacia el servidor, basta con ejecutar desde el sistema cliente el mandato sftp definiendo el usuario a utilizar y el servidor al cual conectar: `sftp usuario@servidor`

El intérprete de mandatos de SFTP es muy similar al utilizado para el protocolo FTP y tiene las mismas funcionalidades.

Para acceder hacia un puerto en particular, en el cual está trabajando el servicio de SSH, se hace través de el parámetro `-o`, con la opción `Port=número de puerto`. En el siguiente ejemplo, utilizando la cuenta del usuario `juan`, se accederá a través de SFTP hacia el servidor `192.168.0.99`, el cual tiene trabajando el servicio de SSH en el puerto `52341`. `sftp -o Port=52341 juan@192.168.0.99`

Si dispone de un escritorio en GNU/Linux, con GNOME 2.x, puede acceder hacia servidores SSH a través del protocolo SFTP utilizando el administrador de ficheros (Nautilus) para realizar transferencias y manipulación de ficheros, especificando el URI (Uniform Resource Locator o Localizador Uniforme de Recursos) «sftp:», seguido del servidor y la ruta hacia la que se quiere acceder, seguido del puerto, en el caso que sea distinto al 22.



## Nautilus, accediendo hacia un directorio remoto a través de SFTP.

### Transferencia de ficheros a través de SCP.

Para realizar transferencias de ficheros a través de SCP, es necesario conocer las rutas de los directorios objetivo del anfitrión remoto. A continuación se describen algunas de las opciones más importantes del mandato scp.

<b>-p</b>	Preserva el tiempo de modificación, tiempos de acceso y los modos del fichero original.
<b>-P</b>	Especifica el puerto para realizar la conexión.
<b>-r</b>	Copia recursivamente los directorios especificados.

En el siguiente ejemplo, se transferirá el fichero algo.txt, preservando tiempos y modos, hacia el directorio de inicio del usuario fulano en el servidor 192.169.0.99.

```
$scp -p algo.txt fulano@192.168.0.99:~/
```

En el siguiente ejemplo, se transferirá la carpeta Mail, junto con todo su contenido, preservando tiempos y modos, hacia el directorio de inicio del usuario fulano en el servidor 192.169.0.99.

```
$scp -rp Mail fulano@192.168.0.99:~/
```

En el siguiente ejemplo, se transferirá la carpeta Mail, junto con todo su contenido, desde el directorio de inicio del usuario fulano en el servidor 192.169.0.99, cuyo servicio de SSH escucha peticiones a través del puerto 52341, preservando tiempos y modos, hacia el directorio del usuario con el que se está trabajando en el anfitrión local.

```
$scp -P 52341 -rp fulano@192.168.0.99:~/Mail ./
```

### Modificaciones necesarias en el muro cortafuegos.

Si se utiliza un cortafuegos con políticas estrictas, como por ejemplo Shorewall, es necesario abrir el puerto 22 por UDP (SSH).

Las reglas para el fichero /etc/shorewall/rules de Shorewall correspondería a algo similar a lo siguiente:

```
#ACTION SOURCE DEST PROTO DEST SOURCE
# PORT PORT(S)1
ACCEPT net fw tcp 22
#LAST LINE -- ADD YOUR ENTRIES BEFORE THIS ONE -- DO NOT REMOVE
```

Si la red de área local (LAN) va a acceder hacia el servidor recién configurado, es necesario abrir el puerto correspondiente.

```
#ACTION SOURCE DEST PROTO DEST SOURCE
# PORT PORT(S)1
ACCEPT net fw tcp 22
ACCEPT loc fw tcp 22
#LAST LINE -- ADD YOUR ENTRIES BEFORE THIS ONE -- DO NOT REMOVE
```

Lo descargaste en la Web de [Yoshiro](#)