SSH (OpenSSH)

Fernando Reyero Noya
Universidad de León, España

camarlengo@yahoo.es

Sinopsis

OpenSSH es un sustituto del conjunto de protocolos SSH libre (en el sentido de libertad. Ver Proyecto GNU (http://www.gnu.org/philosophy/free-sw.es.html)), que permite la comunicación segura o la sustitución de servicios como telnet, rlogin o ftp. OpenSSH encripta todo el tráfico (incluyendo contraseñas) para evitar distintos tipos de ataques de red. Además, proporciona múltiples capacidades de tunneling y métodos de autentificación.

OpenSSH es desarrollado de forma primordial por el Proyecto OpenBSD (http://www.openbsd.org).

Página oficial de OpenSSH (http://www.openssh.com)

Introducción

El paquete ssh que incluye Debian recoge OpenSSH, la implementación libre de SSH (versiones 1 y 2) que lleva a cabo principalmente OpenBSD. Dentro del paquete, se encuentra un conjunto de herramientas seguras como:

- ssh

: el sustituto de rloginy telnet

- scp

: que sustituye a rcp

- sftp

: el reemplazo de ftp

- ssh-agent

, ssh-keygeno sftp-server

El uso de SSH es fundamental en las comunicaciones actuales, ya que la mayoría de servicios de red se llevan a cabo sin cifrar. Como consecuencia, cualquiera puede "escuchar" los datos que transferimos o recibimos.

La versión que usa Debian, al igual que el resto de sistemas que no son OpenBSD, es la basada en las *"versiones p"*. El desarrollo de OpenSSH se centra en dos equipos: el primero realiza el código más

limpio y sencillo posible, teniendo en mente OpenBSD; el segundo toma esta versión y la porta a otro sistemas.

El paquete usado se basaba en la vesión 3.0.2p1

Instalación

La instalación de ssh es realmente sencilla. El paquete Debian no sólo contiene el servidor SSH (sshd), sino también los clientes y un conjunto amplio de aplicaciones. En capítulos posteriores, nos centraremos prioritariamente en la configuración del servidor SSH, que es lo que más concierne a este proyecto.

En la sección de ports (http://www.openssh.com/es/portable.html) de la página oficial de OpenSSH podrás encontrar enlaces a los paquetes correspondientes a cada sistema.

La instalación se reducirá al paquete *ssh*. El resto de paquetes necesarios son gestionados como dependencias de este. Para realizar la instalación, siendo superusuario, escribe en la consola:

potasio:/~# apt-get install ssh

Debian se encargará de instalar la aplicación, realizar una configuración básica y situar los cada fichero en el lugar adecuado. Visita */usr/share/doc/ssh/* para obtener más detalles sobre SSH: su changelog, los comentarios del mantenedor,...

Configuración

Generalmente, por no decir casi siempre, la configuración por defecto de un paquete Debian permite al usuario comenzar a trabajar al instante. En la mayoría de los casos, esta es la situación del paquete *ssh*. Así todo, es muy conveniente repasar alguna de sus posibilidades, con el fin de adecuar el daemon a nuestras necesidades.

Pasar una opción a sshd se puede realizar de dos maneras: mediante una opción en la línea de comandos o el fichero de configuración situado en `/etc/ssh/sshd_config`.

Se debe tener en cuenta que las opciones de la línea de comandos se imponen sobre los valores contenidos en el fichero de configuración y que el envío de una señal *SIGHUP* fuerza al daemon a releer `sshd_config`.

A continuación repasaremos alguna de las opciones más interesantes que aporta `sshd_config`

Opciones de configuración

AllowGroups

Esta opción puede ir seguida de una lista de grupos de nombres, separados por espacios. Si se especifica, sólo se permite realizar un login a los usuarios cuyo grupo principal o suplementarios coincida con uno de los patrones establecidos. Se puede usar '*' y '?' como comodines en los patrones. Sólo se aceptan nombres de grupo; identificadores numéricos de grupo no están reconocidos. Por defecto, se acepta el login independientemente del grupo de pertenencia.

AllowUsers

Esta opción puede ir seguida de una lista de usuarios, separados por espacios. Si se especifica, sólo se permite realizar login a los usuarios cuyo nombre concuerde con el patrón. Se pueden usar '*' y '?' para la construcción de patrones. Sólo se admiten nombres de usuario; nada de identificadores numéricos. Por defecto, se permite realizar login independientemente del nombre de usuario. Si los patrones siguen la foma USUARIO@HOST, entonces USUARIO y HOST se comprueban por separado, restringiéndose el acceso a determinados usuarios de ciertos equipos.

Ciphers

Especifica que cifrados son admitidos para la versión 2. Si especificas múltiples cifrados, sepáralos con comas. El valor por defecto es "aes128-cbc,3des-cbc,blowfish-cbc,cast128-cbc,arcfour".

ClientAliveInterval

Establece el tiempo (en segundos) después del cual, si no se ha recibido datos del cliente, sshd enviará un mensaje a través de un canal encriptado solicitando respuesta al cliente. El valor por defecto es 0, indicando que no se envía mensaje alguno. Esta opción sólo concierne a SSH2.

ClientAliveCountMax

Establece el número de mensajes solicitando una respuesta (ver la opción anterior) que se enviarán sin ninguna contestación por parte del cliente. Si se alcanza este límite, sshd desconectará al cliente, terminando la sesión. Es importante hacer constar la diferencia entre los mensajes de comprobación de la existencia del cliente (client alive) y los mensajes KeepAlive (más abajo). Los primeros, se envían a través de un canal encriptado y no pueden ser interceptados. Los últimos, TCP KeepAlive, sí son interceptables y posibilitan que un cliente o servidor tengan conocimiento de cuándo una conexión se ha vuelto inactiva.

El valor por defecto es 3. Si la opción ClientAliveInterval (anterior) es 15 y ClientAliveCountMax se deja con su valor por defecto, los clientes que no respondan serán desconectados después de 45 segundos aproximadamente.

DenyGroups

Similar a la opción AllowGroups pero con un enfoque de denegación.

DenyUsers

Similar a la opción AllowUsers pero con un enfoque de denegación.

KeepAlive

Especifica si el sistema debe enviar mensajes keepalive al otro extremo. Si se envían, la muerte de la conexión o la interrupción en el funcionamiento en una de las máquinas serán notificadas pertinentemente. Sin embargo, esto significa que las conexiones morirán si la ruta cae temporalmente. Por otro lado, si no se envían mensajes keepalive, las sesiones pueden colgarse indefinidamente en el servidor, dejando usuarios fantasma y consumiendo recursos del sistema.

El valor por predeterminado es "yes" (sí envía mensajes keepalive) y el servidor tendrá conocimiento de si la red cae o el host reinicia. Esto evita sesiones fantasma.

Para deshabilitar keepalives, el valor debe ser "no", tanto en el servidor como en el cliente.

KerberosAuthentication

Especifica si el método de autentificación Kerberos está permitido. Esto puede llevarse a cabo mediante un ticket Kerberos o si la opción PasswordAuthentication está habilitada. En ese caso, la contraseña suministrada por el usuario puede ser validada contra un KDC Kerberos. Para usar esta opción, el servidor necesita un servtab Kerberos que permita la verificación de la identidad del KDC. Por defecto, el valor es "yes".

ListenAddress

Especifica las direcciones locales a las cuales escucha sshd. Se pueden usar las siguientes sintaxis:

- L
istenAddress host|IPv4_addr|IPv6_addr
- L
istenAddress host|IPv4_addr:port
- L
istenAddress [host|IPv6_addr]:port

Si no se especifica el puerto, sshd escuchará en la dirección y todos los puertos que se hayan especificado anteriormente. Por defecto, se escucha en todas las direcciones locales. Se pueden enunciar múltiples opciones ListenAddress.

LoginGraceTime

El servidor desconecta después de este tiempo a los usuarios que no se hayan validado correctamente. Si el valor es 0, no hay límite de tiempo.

Por defecto, 600 (segundos).

LogLevel

Nivel de verbosidad de sshd que se usa en los mensajes del log. Los valores posibles son: QUIET, FATAL, ERROR, INFO, VERBOSE and DEBUG. El predeterminado, INFO. DEBUG viola la privacidad de los usuarios y no se recomienda.

MaxStartups

Especifica el número máximo de conexiones concurrentes sin autentificar que se efectúan contra el servidor sshd. Las conexiones adicionales serán denegadas hasta que la autentificación se tenga lugar o LoginGraceTime expire para una conexión determinada. El valor predeterminado es 10.

PasswordAuthentication

Especifica si la autentificación por password está admitida. Por defecto, "yes".

PermitRootLogin

Especifica si el superusuario puede validar usando ssh. Los argumentos posibles son: "yes", "without-password", "forced-commands-only" o "no". El valor por defecto es "yes".

Si a esta opción se le asigna "without-password", la autentificación por password se deshabilita para el usuario root.

Si a esta opción se le asigna "forced-commands-only", el login del superusuario con autentificación de clave pública será admitido, pero sólo si la opción del comando se ha especificado (lo que puede ser útil para realizar backups remotos incluso si el login de superusuario no está admitido normalmente). El resto de métodos de autentificación están vetados para el root.

Si a la opción se le asigna "no", el root no puede hacer login.

Port

Especifica el número de puerto al que escucha sshd. El puerto predeterminado es el 22. Se admiten opciones múltiples.

Protocol

Especifica las versiones del protocolo que soporta sshd. Las posibilidades son "1" y "2". Las opciones múltiples se separan por comas. El valor por defecto es "2,1".

PubkeyAuthentication

Especifica si se admite la autentificación mediante clave pública. Por defecto, "yes". Esta opción sólo se aplica a la versión 2 del protocolo.

StrictModes

Especifica si sshd debe comprobar los permisos y propietarios de los ficheros del usuario y el directorio home antes de aceptar el login. Es recomendable habilitar esta opción pues los usuarios noveles a veces dejan accidentalmente su directorio o sus ficheros con permisos de escritura universales. El valor por defecto, "yes".

Para obtener la información completa sobre las opciones que admite sshd, lee la página del manual que incluye el paquete.

Fichero de configuración

Ejemplo de */etc/ssh/sshd_config*.

```
# Package generated configuration file
# See the sshd(8) manpage for defails

# What ports, IPs and protocols we listen for
Port 22
# Use these options to restrict which interfaces/protocols sshd will bind to
#ListenAddress ::
#ListenAddress 0.0.0.0
Protocol 2
# HostKeys for protocol version 2
HostKey /etc/ssh/ssh_host_rsa_key
HostKey /etc/ssh/ssh_host_dsa_key
# Lifetime and size of ephemeral version 1 server key
```

```
KeyRegenerationInterval 3600
ServerKeyBits 768

# Logging
SyslogFacility AUTH
LogLevel INFO

# Authentication:
LoginGraceTime 600
PermitRootLogin yes
StrictModes yes

RSAAuthentication yes
PubkeyAuthentication yes
#AuthorizedKeysFile     %h/.ssh/authorized_keys

# rhosts authentication should not be used
RhostsAuthentication no
# Don't read the user's ~/.rhosts and ~/.shosts files
IgnoreRhosts yes
# For this to work you will also need host keys in /etc/ssh_known_hosts
RhostsRSAAuthentication no
# similar for protocol version 2
HostbasedAuthentication no
# Uncomment if you don't trust ~/.ssh/known_hosts for RhostsRSAAuthentication
#IgnoreUserKnownHosts yes

# To enable empty passwords, change to yes (NOT RECOMMENDED)
PermitEmptyPasswords no

# Uncomment to disable s/key passwords
#ChallengeResponseAuthentication no

# To disable tunneled clear text passwords, change to no here!
PasswordAuthentication yes

# Use PAM authentication via keyboard-interactive so PAM modules can
# properly interface with the user
PAMAuthenticationViaKbdInt yes

 To change Kerberos options
#KerberosAuthentication no
#KerberosOrLocalPasswd yes
#AFSTokenPassing no
#KerberosTicketCleanup no

# Kerberos TGT Passing does only work with the AFS kaserver
#KerberosTgtPassing yes

X11Forwarding no
X11DisplayOffset 10
PrintMotd no
#PrintLastLog no
```

```
KeepAlive yes
#UseLogin no

#MaxStartups 10:30:60
#Banner /etc/issue.net
#ReverseMappingCheck yes

Subsystem       sftp    /usr/lib/sftp-server
```

Generación del artículo

Este artículo ha sido producido usando DocBook XML 4.1.2

DocBook es una aplicación XML (también hay una versión SGML) que facilita los sistemas de documentación, al dotar de semática a los textos desde el punto de vista de los sistemas informáticos. Además permite la versatilidad de dar como salida casi cualquier formato documental: LaTeX, TeX, TeXinfo, PDF, RTF, xhtml,...

La edición del texto ha sido realizada con Emacs 21, en el major mode *xml-mode*. Los paquetes Debian usados han sido *psgml* (http://packages.debian.org/testing/text/psgml.html) y *xae (http://packages.debian.org/testing/text/xae.html)*. Las transfomaciones de prueba han sido realizadas con las hojas de estilo XSL mantenidas por Normal Walsh (http://sourceforge.net/projects/docbook), recogidas en el paquete *docbook-xsl (http://packages.debian.org/testing/text/docbook-xsl.html)* y con el procesador *xsltproc (http://packages.debian.org/testing/text/xsltproc.html)*, recogido en el paquete con el mismo nombre.

Sobre este documento

Este documento ha sido escrito en formato XML utilizando la DTD de DocBook (http://www.docbook.org). Mediante este sistema, puede ser fácilmente transformado a múltiples formatos (HTML, TXT, PDF, PostScript, LaTeX, DVI, ...). Se recomienda su utilización como herramienta de documentación potente y libre.

Bibliografía

Página del manual de sshd (http://www.openbsd.org/cgi-bin/man.cgi?query=sshd). 25 de Septiembre de 1999.

Lista de correo de SSH: secureshell-subscribe@securityfocus.com

Patrick Hearon y Algis Rudys *Getting started with SSH (http://linux.rice.edu/help/tips-ssh.html)*. 23 de Septiembre de 2000

The Secure Shell (SSH) Frequently Asked Questions
(http://www.ldeo.columbia.edu/NETWORK/ssh/ssh-faq.html)