Acceder a un Servidor Mediante ssh y un Certificado

Introducción

Cuando no es abierta una cuenta en un servidor, el administrador tiene un problema, cual es hacernos llegar la contraseña. Si la envía en abierto, estará comprometida; si la envía cifrada, ¿hasta cuándo puede extenderse la labor de compartir contraseñas?

La solución a esta dificultad se basa en facilitar el acceso del usuario mediante el uso de certificados. El administrador: abrirá la cuenta, habilitará el servidor para poder acceder a las cuentas mediante certificados, notificará al usuario que su cuenta está abierta y quedará a la espera del certificado público de acceso.

Por su parte, el usuario procederá a generar una pareja de certificados: el público y el privado, le hará llegar al administrador el público en abierto y éste procederá a habilitar el acceso mediante él y el privado, que sólo conocerá el usuario y tendrá guardado celosamente incluso bajo un password dado en el momento de la generación.

De esta manera incluso podremos tener una cuenta compartida por varios usuarios, siempre que éstos consientan, de lo que será garante el administrador.

Habilitación del Servidor Remoto

¿Qué ocurre si ignoramos las instrucciones de esta sección titulada "Habilitación del servidor remoto" y no hacemos ninguno de los cambios que se dirán? El inicio de sesión vía `ssh` sólo funcionará para los usuarios que tengan un campo de contraseña relleno en `/etc/shadow` o un `authorized_key` para `ssh`. Se observará que el valor por defecto para `PubkeyAuthentication` es `yes` y para `PermitEmptyPasswords` es `no`, por lo que si incluso ambos son eliminados el comportamiento será el mismo. Los usuarios que tengan un `authorized_key` para `ssh` podrán entrar mediante la pareja de certificados sólo desde los equipos con certificado privado cuyo correspondiente público haya sido incluido de forma oportuna en el `authorized_key` del servidor; desde el resto de equipos podrían entrar con la contraseña bajo la que esté la cuenta del servidor, si existiera. Quien no tuviese un `authorized_key` y sí contraseña, podría seguir usándola en la forma habitual.

Los cambios que se describen a continuación deben ser efectuados cuando se quiera impedir el acceso por contraseña, quedando como única opción permitida el acceso por una pareja de claves pública y privada. En tal caso no será necesaria la verificación de caducidad de la contraseña `PAM` (Pluggable Authentication Modules), por lo que se habría de deshabilitar este servicio como después se dirá.

La labor, de ser llevada a cabo, debe ser hecha por el administrador del servidor. Suponiendo instalado `ssh`, existirá el fichero `sshd_config` en el directorio `/etc/ssh/`, el cual debe contar con los siguientes campos a los valores que se indica (si alguno no existiese, habría de ser creado):

```
 RSAAuthentication yes
PubkeyAuthentication yes
ChallengeResponseAuthentication no
PasswordAuthentication no
UsePAM no
```

Es muy importante observar la buena costumbre de hacer una copia de respaldo del fichero de configuración que vayamos a modificar antes de hacer esa modificación.

```
sudo cp /etc/ssh/sshd_config /etc/ssh/sshd_config.20151214
```

En el caso de editarlo y cometer algún error, quizás lo dejemos corrupto y podríamos tener un problema muy serio.

Hecho ese importante inciso, en el caso de nuestro equipo, un servidor bajo Ubuntu, editamos el mencionado fichero con nuestro editor preferido al efecto, `nano`, según la siguiente orden:

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

y cambiamos la línea:

```
#PasswordAuthentication yes
```

por la línea:

```
PasswordAuthentication no
```

lo cual supone descomentar tal línea y conmutar el `yes` al `no`. Finalmente debimos cambiar la línea:

```
UsePAM yes
```

por la línea

```
UsePAM no
```

lo que supuso conmutar el `yes` al `no`.

Una vez hechos los cambios, los dejamos registrados y salimos `^X`, `y` y seguidamente `Intro`). Hecho esto es preciso relanzar el servicio para que surta efecto. Ello será llevado a cabo con la orden:

```
sudo service ssh reload
```

En Debian/Ubuntu será precisa la orden:

```
sudo service ssh restart
```

y en CentOS/Fedora:

```
sudo service sshd restart
```

Así habremos acabado en lo que a esta gestión respecta.

Generación de la Pareja de Certificados

La generación de los certificados, suponiendo estar en un sistema `Unix` como los basados en `Debian` o el caso de `Mac OS X`, se lleva a cabo con una utilidad llamada `ssh-keygen` que encontraremos a disposición por el simple hecho de tener instalado `ssh`. Obviamente es una operación llevada a cabo por una CA o del lado del usuario en el equipo desde el que desea acceder al servidor. En el caso de ser del lado del usuario, en su equipo deberá estar creado el directorio `/home/login_equip/.ssh` y si no estuviese creado, habríamos de crearlo.

Ejecutaremos la siguiente orden:

```
ssh-keygen -t rsa -b 4096 -C "su_email@dominio.ext"
```

donde la dirección electrónica se utiliza como una mera etiqueta y podría ser eventualmente la que tuviése para comunicarse con el administrador del servidor en el propio servidor. En esa orden puede ser suprimido el retazo `-b 4096` y entonces el valor tomado por defecto es 2048; es posible que no sean aceptadas claves más largas de 4096 por el fin de la protección contra DDoS. El diálogo será como sigue:

```
Generating public/private rsa key pair.
Enter file in which to save the key (/home/login_equip/.ssh/id_rsa):
```

Es muy recomendado por los especialistas pulsar ahora `Intro` y no cambiar los ajustes por defecto. No obstante, si queremos distinguir los certificados por servidores, le cambiaríamos el nombre a uno sugerente poco explícito y, de ser explícito, en modo alguno evadiríamos poner una clave de uso al certificado que estamos generando (es recomendado poner la clave en cualquier caso):

```
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
```

por supuesto que los renglones anteriores se presentan: uno, se establece la contraseña (en `Unix` se escribe sin ver efecto), aparece la segunda la línea y se repite la contraseña que debe coincidir. Finalmente el sistema genera una huella dactilar (en inglés: *fingerprint*):

```
 The key fingerprint is:
SHA256:m+xH/L0ZLRqTFvHhpKMGoDdpNt2Apd89XXeYaUTdI1Y su_email@dominio.ext
The key's randomart image is:
+---[RSA 4096]----+
|        .    .+E.|
|       +     + =o|
|      + .   o O =|
|     . = + . O oo|
|    . B S.o * +  |
|     + + +o. = . |
|        +.o.=.o .|
|       . ....+.+ |
|        ..  . o. |
+----[SHA256]-----+
```

Al acabar la operación se han generado dos ficheros en `home/login_equip/.ssh`:

```
id_rsa
id_rsa.pub
```

El primero `id_rsa` es la clave privada, a la que no debe tener acceso nadie salvo el usuario y que estará bajo custodia y a resguardo por la contraseña. El fichero segundo `id_rsa.pub` será el que pondremos al alcance del administrador del servidor. La forma de hacerlo puede ser enviarla al mismo en abierto, aunque discretamente, quien lo incluirá en el `authorized_keys` de la cuenta destinada al usuario. Usualmente se elimina el certificado público en el servidor una vez ha sido utilizado y surtido efecto. Seguidamente analizamos la forma de hacerlo.

Quitar o Cambiar la Frase de Contraseña de una Clave Privada.

Si ha generado la clave privada con una frase de contraseña y desea cambiarla o eliminarla, puede hacerlo fácilmente. Para ello, debe conocer la frase de contraseña original. Si ha perdido la frase de contraseña de la clave, no hay recurso y tendrá que generar un nuevo par de claves.

Para cambiar o eliminar la frase de contraseña, simplemente escriba:

```
ssh-keygen -p
```

el diálogo que se produce entonces es según lo siguiente:

```
$ ssh-keygen -p
Enter file in which the key is (/home/user/.ssh/id_rsa):
```

Puede escribir la ubicación de la clave que desea modificar o presionar intro para aceptar el valor predeterminado:

```
Enter old passphrase:
```

Ahora deberá introducir la frase de contraseña anterior, la cual desea cambiar. Luego le será solicitada una nueva frase de contraseña:

```
Enter new passphrase (empty for no passphrase):
Enter same passphrase again:
```

Aquí, deberá introducir su nueva frase de contraseña o presionar intro para eliminar la frase de contraseña de su clave.

Visualizar la Huella Digital de la Clave SSH

Cada par de claves SSH comparte una única "huella digital" criptográfica que se puede utilizar para identificar de forma exclusiva las claves. Esto puede ser útil en una variedad de situaciones.

Para averiguar la huella digital de una clave SSH, escriba:

```
ssh-keygen -l
```

El diálogo es según lo siguiente:

```
$ ssh-keygen -l
Enter file in which the key is (/home/user/.ssh/id_rsa):
```

Puede presionar intro si esa es la ubicación correcta de la clave; de lo contrario, introduzca la ubicación revisada. Se le dará una cadena que contiene la longitud de bits de la clave, la huella digital y la cuenta y el host para el que se creó, y el algoritmo utilizado:

```
4096 8e:c4:82:47:87:c2:26:4b:68:ff:96:1a:39:62:9e:4e  demo@test (RSA)
```

Copiar la Clave SSH Pública en un Servidor con `ssh-copy-id`

Para copiar su clave pública a un servidor, lo que le permite autenticarse sin una contraseña, son posibles varios enfoques.

Si actualmente tiene un acceso SSH basado en contraseña configurado en su servidor y tiene instalada la orden `ssh-copy-id`, éste es un proceso simple. La herramienta `ssh-copy-id` se incluye en los paquetes OpenSSH de muchas distribuciones de Linux, por lo que es muy probable que se instale de manera predeterminada.

Si tiene esta opción, puede transferir fácilmente su clave pública ejecutando:

```
ssh-copy-id login_server@nombre_servidor.dominio.ext
```

Esto le pedirá la contraseña de la cuenta de usuario en el sistema remoto:

```
 The authenticity of host '111.111.11.111 (111.111.11.111)' can't be established.
ECDSA key fingerprint is fd:fd:d4:f9:77:fe:73:84:e1:55:00:ad:d6:6d:22:fe.
Are you sure you want to continue connecting (yes/no)? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter
out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are
prompted now it is to install the new keys
demo@111.111.11.111's password:
```

Después de escribir la contraseña, el contenido de su clave `~/.ssh/id_rsa.pub` se agregará al final del archivo `~/.ssh/Authorizedkeys` de la cuenta de usuario:

```
 Number of key(s) added: 1

Now try logging into the machine, with:   "ssh 'demo@111.111.11.111'"
and check to make sure that only the key(s) you wanted were added.
```

Ahora puede iniciar sesión en esa cuenta sin una contraseña:

```
ssh username@remote_host
```

Una variante de lo anterior es lo siguiente. El usuario puede añadir l`id_rsa.pub` al fichero `authorized_keys` de su cuenta mediantela orden:

```
ssh-copy-id -i /home/login_equip/.ssh/id_rsa.pub \
    login_server@nombre_servidor.dominio.ext
```

Para probar el acceso al equipo use la siguiente orden:

```
ssh -i /home/login_equip/.ssh/id_rsa \
    login_server@nombre_servidor.dominio.ext -o VisualHostKey=yes
```

y entonces el mensaje sería algo así como:

```
Host key fingerprint is SHA256:HVRPMg9A/9TXDQQsxbtWBPBcLxNvLUeq8xq7ugjXVYA
+---[ECDSA 256]---+
|         .=OX+* .|
|         .E+oX.O+|
|          ..+oX @|
|         . ..=.*.|
|        S . +o.  |
|         . .oo   |
|      . . ... .  |
|       o .   +   |
|        . oo+.   |
+----[SHA256]-----+

Last login: Sun Nov 29 17:13:47 2015 from 4.127.78.188.dynamic.ctelefono.com
login_server@login_server
```

o bien se puede probar con:

```
ssh login_server@nombre_servidor.dominio.ext -o VisualHostKey=yes
```

o simplemente mediante:

```
ssh login_server@nombre_servidor.dominio.ext
```

Copiar la Clave Pública en un Servidor Sin `ssh-copy-id`

Si no tiene disponible la utilidad `ssh-copy-id`, pero aún tiene acceso SSH basado en contraseña al servidor remoto, puede copiar el contenido de su clave pública de una manera diferente.

Puede generar el contenido de la clave y concatenarlo con una order `ssh`. En el lado remoto puede asegurarse de que exista el directorio `~/.ssh` y luego agregar los contenidos canalizados al archivo `~/.ssh/Authorised_keys`:

```
cat ~/.ssh/id_rsa.pub | ssh username@remote_host "mkdir -p ~/.ssh && \
    cat >> ~/.ssh/authorized_keys"
```

Se le pedirá que proporcione la contraseña para la cuenta remota:

```
The authenticity of host '111.111.11.111 (111.111.11.111)' can't be established.
ECDSA key fingerprint is fd:fd:d4:f9:77:fe:73:84:e1:55:00:ad:d6:6d:22:fe.
Are you sure you want to continue connecting (yes/no)? yes
demo@111.111.11.111's password:
```

Después de introducir la contraseña, su clave se copiará permitiéndole iniciar sesión sin contraseña:

```
ssh username@remote_host
```

Copiar la Clave Pública Manualmente

El usuario sin acceso remoto al servidor (que sin embargo puede acceder a su cuenta en él) o el administrador cuando reciba el certificado público de nombre `id_rsa.pub`, debe incluir su contenido en el fichero `authorized_keys` de la cuenta del usuario en el servidor. Si la operación es llevada a cabo contando con `id_rsa.pub` del usuario en, digamos `/home/login_server/.ssh/id_rsa.pub` del servidor, puede hacerlo con la siguiente orden:

```
cat /home/login_server/.ssh/id_rsa.pub >> /home/login_server/.ssh/authorized_keys
```

Deshabilitar un Acceso Anterior Mediante Certificado

Cuando queremos acceder mediante certificado a un lugar al que en tiempos pudimos hacerlo, pero que ha cambiado, es posible que obtengamos por terminal un mensaje del tipo:

```
 User:~$ ssh remoteuser@server.dom.ext
Warning: the ECDSA host key for 'server.dom.ext' differs from the key for the
IP address '109.217.186.125'
Offending key for IP in /home/user/.ssh/known_hosts:1
Matching host key in /home/user/.ssh/known_hosts:6
Are you sure you want to continue connecting (yes/no)?
```

La forma de suprimir este inconveniente es, según informathegeekstuff, es ejecutando la siguiente orden:

```
sed -i '6d' ~/.ssh/known_hosts
```

donde `6d` debe ser cambiado dependiendo del número de línea mostrado en el mensaje de la terminal.

Compartir una Cuenta

Si una misma cuenta hubiera de ser compartida por varios usuarios, el administrador puede recibir el certificado público `id_rsa.pub` aportado por cada uno de los usuario y reproducir el proceso de la primera vez:

```
cat /home/login_server/.ssh/id_rsa.pub >> /home/login_server/.ssh/authorized_keys
```

o bien que cada usuario que va a compartir ejecute, si puede:

```
ssh-copy-id -i /home/login_equip/.ssh/id_rsa_second.pub \
    login_server@nombre_servidor.dominio.ext
```

Agregar las claves SSH a un Agente SSH para Evitar Escribir la Frase de Contraseña

Si tiene una frase de contraseña en su clave SSH privada, se le pedirá que introduzca la frase de contraseña cada vez que la use para conectarse a un host remoto.

Para evitar tener que hacer esto repetidamente, puede ejecutar un agente SSH. Esta pequeña utilidad almacena su clave privada después de haber ingresado la frase de contraseña por primera vez. Estará disponible durante la sesión de su terminal, lo que le permitirá conectarse en el futuro sin tener que volver a ingresar la frase de contraseña.

Esto también es importante si necesita reenviar sus credenciales SSH (que se muestran a continuación).

Para iniciar el Agente SSH, escriba lo siguiente en su sesión de terminal local:

```
eval $(ssh-agent)

Agent pid 10891
```

Esto iniciará el programa del agente y lo colocará en segundo plano. Ahora, debe agregar su clave privada al agente, para que pueda administrar su clave:

```
ssh-add
```

lo que producirá el siguiente diálogo:

```
Enter passphrase for /home/demo/.ssh/id_rsa:
Identity added: /home/demo/.ssh/id_rsa (/home/demo/.ssh/id_rsa)
```

Tendrá que introducir su frase de contraseña (si está configurada). Luego, su archivo de identidad se agrega al agente, lo que le permite usar su clave para iniciar sesión sin tener que volver a introducir la frase de contraseña nuevamente.

Reenviar sus credenciales SSH para usar en un servidor

Si desea poder conectarse sin una contraseña a un servidor desde otro servidor, deberá reenviar la información de su clave SSH. Esto le permitirá autenticarse en otro servidor a través del servidor al que está conectado, utilizando las credenciales en su computadora local.

Para comenzar, debe tener su agente SSH iniciado y su clave SSH agregada al agente (ver arriba). Una vez hecho esto, debe conectarse a su primer servidor utilizando la opción `-A`. Esto reenvía sus credenciales al servidor para esta sesión:

```
ssh -A username@remote_host
```

Desde aquí, puede enviar SSH a cualquier otro host al que su clave SSH esté autorizada para acceder. Se

conectará como si su clave SSH privada estuviera ubicada en este servidor.