Conección con SSH y OpenSSH con Procolos Versión 2

13 de septiembre de 2001

Este documento detalla los pasos para lograr una conección a una máquina remota sin escribir la clave de acceso (sin introducir el *password*). Esto se logra con la *autenticación basada en host* (host-based authentication). Se explicarán tres casos:

1. Host-host ejecutando SSH
2. Host-host ejecutando OpenSSH
3. Host con SSH a host ejecutando OpenSSH

1 Autenticación basada en HOST con SSH

Los siguientes términos van a usarse en el siguiente ejemplo:

Remota es el servidor SSH Secure Shell en el cual se va tratar de conectarse. UsuarioRemoto es el nombre del usuario del servidor al que se desea conectarse. Local es la máquina que corre el cliente SSH Secure Shell. UsuarioLocal es el nombre de la máquina cliente que va a permitir que entre Remota como UsuarioRemoto.

Primero debe instalarse el SSH Secure Shell tanto en la máquina Local como en la maquina Remota. No olvide generar una llave host. Si la instalación se hizo así o si se tiene una copia lista de /etc/ssh2/hostkey y /etc/ssh2/hostkey.pub ya no es necesario crear la llave host. De lo contrario debe realizarse esto:

```
# ssh-keygen2 -P
```

1. Las llaves publicas y privadas se crean ejecutando ssh–keygen (ó ssh–keygen2 en una máquina con SSH) en las maquinas locales.

   ```
   Local > ssh-keygen
   ```
   Generando par de llaves $1024 - bit$ dsa
   o.o0o..o0o.o
   Llave generada
   1024–bit dsa, creado para user@hosts Wed Sep 23 07:11:02 2001
   Passphrase:
   Again:
   Llave privada fue salvada en $HOME/.ssh2/id_dsa_1024_a
   Llave pública fue salvada en $HOME/.ssh2/id_dsa_1024_a.pub

   ssh–keygen va a preguntará por el passphrase para la nueva llave. Se introducirse cualquier secuencia de caracteres ordinarios (con espacios esta bien) con una longitud de 20 caracteres; ssh–keygen crea el directorio ".ssh2" en el directorio de $HOME y guarda la nueva llave de autenticación en dos archivos separados. Una es tu llave privada (id_dsa_1024_a) y ésta no debe de abrirse más que por el UsuarioLocal (esto es, debe de tener los permisos 0600). La llave pública es asegurada para que se pueda abrir y distribuir a las otras computadoras.

2. Crea un archivo de "identificación" en el directorio ".ssh2" de la máquina local:

   ```
   Local > cd ~/.ssh2
   Local > echo ‘‘IdKey id_dsa_1024_a > identification
   ```

   Este va a crear un archivo de identificación (de nombre en inglés *identification*) en el directorio ".ssh2", el cual tiene una línea que denota que archivo contiene la identificación. La identificación corresponde al passphrase. Se pueden crear múltiples identificaciones ejecutando susesivamente el comando ssh–keygen.

3. Realizar los mismos pasos (1 y opcionalmente el 2) en la máquina Remota. Esto solo se necesita para crear el directorio ".ssh2" en la máquina Remota. El passphrase debe ser diferente.

4. Copia tu pública llave en Local (id_dsa_1024_a.pub) al directorio ".ssh2" de la máquina Remota bajo el nombre de "Local.pub".

   El directorio ".ssh2" de la máquina Remota ahora contiene:

   ```
   Remota > ls -F ~/.ssh2
   ```
   Local.pub
   authorization
   hostkeys
   id_dsa_1024_a
   id_dsa_1024_a.pub
   identification
   random_seed

5. Se crea un archivo "authorization" en el directorio ".ssh2" en la máquina Remota. Se adiciona la siguiente línea en el archivo de "authorization"

   ```
   Key Local.pub
   ```

La máquina Remota ahora tiene la llave publica de la máquina Local, así que la máquina Remota puede verificar la identidad de la máquina Local basándose en las llaves publicas. A diferencia, rsh solamente usa la dirección IP para la autentificación del host.

Para asegurar que SSH va a encontrar sin ningún problema el nombre del dominio, no solamente el nombre del host, debe editarse la siguiente linea en el archivo /etc/ssh2/ssh2_config en la máquina Local:

```
DefaultDomain yourdomain.com
```

Checar en los archivos /etc/ssh2/sshd2_config en la máquina Remota y /etc/ssh2_config en la máquina Local.

Asegurarse que el campo AllowedAuthentifications contiene la palabra hostbased. Por ejemplo:

```
AllowedAuthentications hostbased,passwd
```

No tiene importancia si algun otro esta ahí. Solo debe asegurarse de que la palabra clave *hostbased* es el primero en la lista.

También debe checase que el rhost este deshabilitado en el /etc/ssh2/sshd2_config en la máquina Remota.

```
Ignore Rhosts  no
```

Si se ha modificado el archivo sshd2_config, se debe de enviar una señal HUP al sshd2 para que el cambio tenga efecto:

```
# kill -HUP ‘cat /var/run/sshd2_22.pid‘
```

2 Autenticación basada en HOST con OpenSSH

Tanto en la máquina remota como en la máquina Local hay que generar el par de llaves, pública y privada, para el algoritmo DSA. Esto se realiza así:

```
ssh-keygen -d -N ’’
```

para la versión 7.0 de RedHat, y

```
ssh-keygen -t dsa -N ’’
```

para la versión 7.1 (Seawolf) de RedHat.

El comando `ssh-keygen` genera el par de archivos id_dsa y id_dsa.pub, donde se guardan la dos llaves DSA privada y pública, respectivamente.

La llave pública de cada máquina debe adicionarse en el archivo `$HOME/.ssh/authorized_keys2` de la contraria, esto es, en todas las máquinas donde se desee entrar usando la autenticación por DSA. Esto se puede hacer, por ejemplo, si estamos en la máquina Local haciendo:

```
cd
scp UsuarioRemoto@Remota:.ssh/id_dsa.pub .
cat id_dsa.pub >> .ssh/authorized_keys2
```

Y lo mismos pasos deben hacerse si se está en la máquina Remota.

3 Autenticación basada en HOST entre SSH y OpenSSH

El comando *ssh–keygen* de OpenSSH tiene las banderas necesarias para cambiar de formato la llave pública de SSH y para generar una llave pública compatible para SSH. Esto se logra así:

1. Cambiar una llave pública de SSH para OpenSSH:
   `ssh-keygen -f key_from_ssh.com -X >> ~/.ssh/authorized_key2`

2. Para generar una llave pública de OpenSSH compatible con SSH: `ssh-keygen -f private`
   `echo "Key mykey.pub" >> ~/.ssh2/authorization`

Las banderas de OpenSSH que cambian el formato son (del manual de referencia de ssh–keygen):

`-x` Esta opción leerá un archivo con formato DSA privado de OpenSSH e imprimá a la salida estándar una llave pública compatible con SSH2

`-X` Esta opción leerá un archivo con una llave pública compatible con SSH2 e imprimirá a la salida estándar una llave pública compatible con DSA de OpenSSH.