Community

TUTORIAL

Cómo configurar las llaves SSH en Ubuntu 18.04

Linux Basics Getting Started Security System Tools Ubuntu 18.04

By Hanif Jetha
Posted June 21, 2018 62.1k

Select an Option

Introducción

SSH, o blindaje seguro (secure shell) es un protocolo encriptado para administrar y comunicarse con servidores. Al trabajar con servidores Ubuntu, generalmente pasarás la mayor parte del tiempo conectado mediante SSH desde una terminal a tu servidor.

En esta guía, nos enfocaremos en configurar las llaves SSH sobre una instalación de Ubuntu Vanilla 18.04. Las llaves SSH proveen una autenticación fácil y segura para tu servidor, esta autenticación es la recomendada para todos los usuarios.

Paso 1 — Crea un par de llaves RSA

El primer paso consiste en crear un par de llaves en la máquina cliente (usualmente tu computador):

```
$ ssh-keygen
```

De manera predeterminada, `ssh-keygen` creará un par de llaves RSA de 2.048 bits, lo cual

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


Si se han generado previamente un par de llaves SSH, deberías ver la siguiente información:

```
Output
/home/your_home/.ssh/id_rsa already exists.
Overwrite (y/n)?
```

Si se escoge sobreescribir la llave en disco, **no** podrás autenticarte usando las llaves previas de ahora en adelante. Sé muy cuidadoso al seleccionar la opción positiva (y), ya que éste es un proceso destructivo de las llaves que no puede ser reversado.

Debería desplegarse lo siguiente en la línea de comandos:

```
Output
Enter passphrase (empty for no passphrase):
```

En este punto se puede introducir una frase segura que sirva como contraseña, lo cual es altamente recomendado. Esta frase adicionará una capa extra de seguridad para prevenir la autenticación de usuarios no autorizados. Para aprender más sobre seguridad, consulta nuestro tutorial sobre cómo configurar la autenticación mediante llaves SSH en un servidor Linux.

A continuación, deberías ver la siguiente salida:

```
Output
Your identification has been saved in /your_home/.ssh/id_rsa
```

```
|    o. o         |
|       ..        |
|     ..S         |
+-----------------+
```


Ahora ya existe una llave pública y una privada que puedes usar en el proceso de autenticación. El siguiente paso será colocar la llave pública en tu servidor, de tal manera que la puedas usar para acceder a él, mediante una autenticación basada en llaves SSH.

Paso 2 — Copia la llave pública en el servidor Ubuntu

La manera más rápida de copiar tu llave pública en el servidor Ubuntu es mediante el uso de una utilidad llamada `ssh-copy-id`. Gracias a su simplicidad, de estar disponible es un método altamente recomendado. En caso de que `ssh-copy-id` no se encuentre disponible en tu máquina cliente, aún puedes usar uno de los dos métodos que se proveen en esta sección: copiado de una contraseña usando SSH, o el copiado manual de la llave.

Copia de la llave pública usando `ssh-copy-id`

La herramienta `ssh-copy-id` se encuentra incluida de manera predeterminada en varios sistemas operativos, por lo cual existe la posibilidad que esté disponible en tu sistema local. Para que este método funcione es necesario que ya se cuente con acceso por contraseña mediante SSH dentro de tu servidor.

Para usar este método, simplemente se debe especificar el servidor remoto al cual te quieres conectar, así como la cuenta de usuario y su contraseña con acceso SSH. Esta cuenta es a la cual se copiará tu llave SSH pública.

La sintaxis es:

```
Output
The authenticity of host '203.0.113.1 (203.0.113.1)' can't be established.
ECDSA key fingerprint is fd:fd:d4:f9:77:fe:73:84:e1:55:00:ad:d6:6d:22:fe.
```


ede la primera vez que te conectas a una nueva máquina. Digita “yes” y presiona `Enter` para continuar.

A continuación, la utilidad escaneará tu cuenta local en búsqueda de la llave `id_rsa.pub` creada previamente. Cuando la llave es creada, se te solicitará la contraseña para la cuenta del usuario remoto:

```
Output
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are alr
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to inst
username@203.0.113.1's password:
```

Digita la contraseña (lo digitado no será mostrado en pantalla por razones de seguridad), y presiona `Enter`. La utilidad se conectará a la cuenta en la máquina remota usando la contraseña que proveíste. Esto copiará el contenido de tu llave `~/.ssh/id_rsa.pub` en el archivo que se encuentra en el directorio local de la cuenta remota `~/.ssh` llamado `authorized_keys`.

Se te debería desplegar la siguiente salida:

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

Now try logging into the machine, with:   "ssh 'username@203.0.113.1'"
and check to make sure that only the key(s) you wanted were added.
```

convencional.

Esto se puede realizar usando el comando `cat` para leer el contenido de la llave pública

r remoto.

permisos adecuados dentro de la cuenta que se está usando.

Podemos direccionar el contenido enviado a un archivo llamado `authorized_keys` dentro de este directorio. Usaremos el símbolo de redirección `>>` para adicionar el contenido sin reescribirlo, lo que nos permitirá adicionar llaves nuevas sin destruir las adicionadas previamente.

El comando completo lucirá como lo siguiente:

```
$ cat ~/.ssh/id_rsa.pub | ssh username@remote_host "mkdir -p ~/.ssh && touch ~/.ssh/authorized_key
```

Se podría desplegar el siguiente mensaje:

```
Output
The authenticity of host '203.0.113.1 (203.0.113.1)' 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
```

Esto significa que el computador local no reconoce al remoto. Esto sucede la primera vez que se conecta a una máquina remota. Digita “yes” y presiona `Enter` para continuar.

En este momento, se te solicitará que introduzcas la contraseña de la cuenta del usuario remoto:

```
Output
```

Si no se cuenta con un acceso al servidor mediante contraseña SSH, puedes completar el proceso de forma manual.

Para desplegar el contenido de la llave `id_rsa.pub`, digita lo siguiente en la máquina local:

```
$ cat ~/.ssh/id_rsa.pub
```

Verás el contenido de la llave, que debería ser similar a algo como lo siguiente:

```
Output
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAACAQCqql6MzstZYh1TmWWv11q5O3pISj2ZFl9HgH1JLknLLx44+tXfJ7mIrKNxOOw›
```

Accede al computador remoto usando cualquier método que tengas disponible. Una vez hayas accedido, debes asegurarte que el directorio `~/.ssh` exista. De no ser así, lo puedes crear con el siguiente comando:

```
$ mkdir -p ~/.ssh
```

Ahora, se puede modificar, o incluso crear, el archivo `authorized_keys` dentro de este directorio. Puedes adicionar el contenido del archivo `id_rsa.pub` al final de archivo `authorized_keys`, o crearlo de ser necesario, mediante el comando:

```
$ echo public_key_string >> ~/.ssh/authorized_keys
```

En el anterior comando, sustituye `public_key_string` con la salida que habías obtenido del comando `cat ~/.ssh/id_rsa.pub` que ya habías ejecutado en la máquina local. Éste debería comenzar con `ssh-rsa AAAA...`.

Esto remueve recursivamente todos los permisos dados a grupos “group” y a otros “other” en el directorio `~/.ssh/`.

nta de
rio y no al

```
$ chown -R sammy:sammy ~/.ssh
```

En este tutorial nuestro usuario es llamado sammy, en el comando anterior, debes sustituirlo por el nombre de usuario apropiado.

Ahora ya estamos listos para intentar la autenticación sin contraseña en nuestro servidor Ubuntu.

Paso 3 — Autentícate en un servidor Ubuntu usando llaves SSH

Si ya has completado satisfactoriamente uno de los procesos anteriores, ya deberías estar habilitada para autenticarte **sin** necesidad de la contraseña de la cuenta remota.

El proceso básico es el mismo:

```
$ ssh username@remote_host
```

Si es la primera vez que se intenta la conexión con esta máquina (y, por ejemplo, usaste el último método de la sección anterior), quizás se despliegue algo similar a lo siguiente:

```
Output
The authenticity of host '203.0.113.1 (203.0.113.1)' 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
```

(notarás que por motivos de seguridad no se imprimirán los caracteres en la sesión de terminal). Después de la autenticación, un sesión segura se te debería desplegar, configurada con la cuenta del servidor Ubuntu.

orar la

Paso 4 — Deshabilita en tu servidor la autenticación por contraseña

Si tu sistema ya puede ser accedido mediante SSH sin una contraseña de usuario, ya has configurado de manera exitosa la autenticación mediante llave SSH en tu cuenta. Sin embargo, el mecanismo de autenticación mediante contraseña de usuarios sigue activo, lo cual significa que tu servidor aún se encuentra expuesto a posibles ataques de fuerza bruta.

Antes de completar las indicaciones de esta sección, debes asegurarte de tener, o bien, configurada la autenticación de la cuenta de superusuario mediante llave SSH, o preferiblemente, la autenticación mediante SSH configurada para una cuenta diferente a la de superusuario con privilegios de `sudo`. Este paso restringirá los accesos mediante contraseña, por lo cual es crucial que te asegures de tener control administrativo de tu servidor.

Tan pronto confirmes que tu cuenta remota cuenta con privilegios administrativos, accede al servidor remoto utilizando las llaves SSH, ya sea como superusuario o con una cuenta con privilegios de `sudo`. Después, abre el archivo de configuración del demonio SSH:

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

Dentro del archivo, busca la directiva llamada `PasswordAuthentication`. Ésta podría estar en comentario. Si lo está, retira el marcador de comentario sobre esa línea y fija el valor en

...

Salve y cierre el archivo cuando haya terminado digitando `CTRL-X`, después `Y` para confirmar los cambios, y luego `ENTER` para salir. Para implementar estos cambios, ejecute el siguiente comando:


```
$ sudo systemctl restart ssh
```

Como precaución, abre una nueva ventana de la terminal y verifica que el servicio de SSH se encuentre funcionando correctamente antes de cerrar esta sesión:

```
$ ssh username@remote_host
```

Una vez hayas verificado tu servicio SSH, puedes cerrar de manera segura todas las sesiones actuales del servidor.

Ahora, el demonio SSH solo responderá a llaves SSH. Autenticación mediante contraseña de usuarios habrá sido deshabilitada exitosamente.

Conclusión

Para este momento ya deberías haber configurado en tu servidor, la autenticación mediante llaves SSH, habilitándote a acceder sin proveer una contraseña de una cuenta.

Si quieres aprender más acerca de trabajar con SSH, puedes usar nuestra guía esencial de SSH.