Products & Services › Product Documentation › Red Hat Enterprise Linux › 6 › Deployment Guide ›
14.3.3. Creating SSH CA Certificate Signing Keys

14.3.3. CREATING SSH CA CERTIFICATE SIGNING KEYS

Two types of certificates are required, host certificates and user certificates. It is considered better to have two separate keys for signing the two certificates, for example `ca_user_key` and `ca_host_key`, however it is possible to use just one CA key to sign both certificates. It is also easier to follow the procedures if separate keys are used, so the examples that follow will use separate keys.

The basic format of the command to sign user's public key to create a user certificate is as follows:

```
ssh-keygen -s ca_user_key -I certificate_ID id_rsa.pub
```

Where `-s` indicates the private key used to sign the certificate, `-I` indicates an identity string, the *certificate_ID*, which can be any alpha numeric value. It is stored as a zero terminated string in the certificate. The *certificate_ID* is logged whenever the certificate is used for identification and it is also used when revoking a certificate. Having a long value would make logs hard to read, therefore using the host name for host certificates and the user name for user certificates is a safe choice.

To sign a host's public key to create a host certificate, add the `-h` option:

```
ssh-keygen -s ca_host_key -I certificate_ID -h ssh_host_rsa_key.pub
```

Host keys are generated on the system by default, to list the keys, enter a command as follows:

```
~]# ls -l /etc/ssh/ssh_host* -rw-------. 1 root root 668 Jul 9 2014
/etc/ssh/ssh_host_dsa_key -rw-r--r--. 1 root root 590 Jul 9 2014
/etc/ssh/ssh_host_dsa_key.pub -rw-------. 1 root root 963 Jul 9 2014
/etc/ssh/ssh_host_key -rw-r--r--. 1 root root 627 Jul 9 2014
/etc/ssh/ssh_host_key.pub -rw-------. 1 root root 1671 Jul 9 2014
/etc/ssh/ssh_host_rsa_key -rw-r--r--. 1 root root 382 Jul 9 2014
/etc/ssh/ssh_host_rsa_key.pub
```

Important

It is recommended to create and store CA keys in a safe place just as with any other private key. In these examples the `root` user will be used. In a real production environment using an offline computer with an administrative user account is recommended. For guidance on key lengths see *NIST Special Publication 800-131A*.

Procedure 14.1. Generating SSH CA Certificate Signing Keys

1. On the server designated to be the CA, generate two keys for use in signing certificates. These are the keys that all other hosts need to trust. Choose suitable names, for example `ca_user_key` and `ca_host_key`. To generate the user certificate signing key, enter the following command as `root`:

   ```
   ~]# ssh-keygen -t rsa -f ~/.ssh/ca_user_key Generating public/private rsa
   key pair. Created directory '/root/.ssh'. Enter passphrase (empty for no
   passphrase): Enter same passphrase again: Your identification has been saved
   in /root/.ssh/ca_user_key. Your public key has been saved in
   /root/.ssh/ca_user_key.pub. The key fingerprint is:
   11:14:2f:32:fd:5d:f5:e4:7a:5a:d6:b6:a0:62:c9:1f root@host_name.example.com
   The key's randomart image is: +--[ RSA 2048]----+ | .+. o| | . o +.| | o + .
   . o| | o + . . ..| | S . ... *| | . . . .*.| | = E .. | | . o . | | . | +---
   --------------+
   ```

   Generate a host certificate signing key, `ca_host_key`, as follows:

```
~]# ssh-keygen -t rsa -f ~/.ssh/ca_host_key Generating public/private rsa
key pair. Enter passphrase (empty for no passphrase): Enter same passphrase
again: Your identification has been saved in /root/.ssh/ca_host_key. Your
public key has been saved in /root/.ssh/ca_host_key.pub. The key fingerprint
is: e4:d5:d1:4f:6b:fd:a2:e3:4e:5a:73:52:91:0b:b7:7a
root@host_name.example.com The key's randomart image is: +--[ RSA 2048]----+
| .. | | . ....| | . . o +oo| | o . o *o| | S = .| | o. .| | *.E. | | +o= |
| .oo. | +-----------------+
```

If required, confirm the permissions are correct:

```
~]# ls -la ~/.ssh total 40 drwxrwxrwx. 2 root root 4096 May 22 13:18 . dr-
xr-x---. 3 root root 4096 May 8 08:34 .. -rw-------. 1 root root 1743 May 22
13:15 ca_host_key -rw-r--r--. 1 root root 420 May 22 13:15 ca_host_key.pub -
rw-------. 1 root root 1743 May 22 13:14 ca_user_key -rw-r--r--. 1 root root
420 May 22 13:14 ca_user_key.pub -rw-r--r--. 1 root root 854 May 8 05:55
known_hosts -r--------. 1 root root 1671 May 6 17:13 ssh_host_rsa -rw-r--r-
-. 1 root root 1370 May 7 14:30 ssh_host_rsa-cert.pub -rw-------. 1 root
root 420 May 6 17:13 ssh_host_rsa.pub
```

2. Create the CA server's own host certificate by signing the server's host public key together with an identification string such as the host name, the CA server's *fully qualified domain name* (FQDN) but without the trailing `.`, and a validity period. The command takes the following form:

```
ssh-keygen -s ~/.ssh/ca_host_key -I certificate_ID -h -Z
host_name.example.com -V -start:+end /etc/ssh/ssh_host_rsa.pub
```

The `-Z` option restricts this certificate to a specific host within the domain. The `-V` option is for adding a validity period; this is highly recommend. Where the validity period is intended to be one year, fifty two weeks, consider the need for time to change the certificates and any holiday periods around the time of certificate expiry.

For example:

```
~]# ssh-keygen -s ~/.ssh/ca_host_key -I host_name -h -Z
host_name.example.com -V -1w:+54w5d /etc/ssh/ssh_host_rsa.pub Enter
passphrase: Signed host key /root/.ssh/ssh_host_rsa-cert.pub: id "host_name"
serial 0 for host_name.example.com valid from 2015-05-15T13:52:29 to 2016-
06-08T13:52:29
```

14.3.2. Support for SSH Certificates

14.3.4. Distributing and Trusting SSH CA Public Keys