

KeyTalk Linux agent quick installation and usage guide

Introduction

This document outlines installation and usage of KeyTalk Linux agent for the purpose to obtain and effectuate X.509 certificates. The most typical use of these certificates is client certificate authentication, which this guide primarily focuses on. For obtaining SSL certificates meant for securing Apache and Tomcat web servers, please refer to the appropriate guides.

System Requirements

- Ubuntu 18.04 LTS or Ubuntu 20.04 LTS x64 or Ubuntu 22.04 LTS x64

Compatible KeyTalk versions

This admin manual requires KeyTalk agent v6.4.5 or higher and KeyTalk server version 6.4.4 and higher

KeyTalk agent configuration

Install and customize KeyTalk agent

Extract:

```
tar -xf KeyTalkClient-<version>.linux.tar
```

Install using:

```
sudo ./install.sh
```

Alternatively, on Ubuntu 20 and higher you can install from .deb package:

```
sudo apt -y update
sudo apt -y install KeyTalkClient_<version>.deb
```

Customize KeyTalk installation with a RCCD file:

```
/usr/local/bin/keytalk/ktconfig --rccd-path <path-to-rccd>
```

To explore more configuration options, run

```
/usr/local/bin/keytalk/ktconfig --help
```

Obtaining a certificate for generic usage

```
/usr/local/bin/keytalk/ktclient --user <keytalk-user> --password <keytalk-password>
```

Or, when there are multiple KeyTalk templates:

```
/usr/local/bin/keytalk/ktclient --service <keytalk-template> --user <keytalk-user> --password <keytalk-password>
```

This will obtain a certificate and store it in PEM format under ~/.keytalk/keystore/

To explore more configuration options, run

```
/usr/local/bin/keytalk/ktclient --help
```

Obtaining a certificate for use by Google Chrome

Ensure that Google Chrome has been opened at least once. This will create NSS security Db where Chrome stores certificates. You can check its existence using

```
ls -la ~/.pki/nssdb
```

If NSS Db was created after KeyTalk agent was customized, you will have to customize KeyTalk again to import KeyTalk trusted Certificate Authority into NSS.

```
/usr/local/bin/keytalk/ktconfig --rccd-path <path-to-rccd>
```

Check the KeyTalk trusted CA is installed in NSS Db

```
/usr/local/bin/keytalk/ktclient --nss-list
```

Obtain a certificate and install it in NSS Db

```
/usr/local/bin/keytalk/ktclient --nss-add --user <keytalk-user> --password <keytalk-password>
```

Or, when there are multiple KeyTalk templates:

```
/usr/local/bin/keytalk/ktclient --nss-add --service <keytalk-template> --user <keytalk-user> --password <keytalk-password>
```

Check the KeyTalk client certificate is installed in NSS Db

```
/usr/local/bin/keytalk/ktclient --nss-list
```

Obtaining a certificate backed by TPM on local machine

This feature is only available on Ubuntu 20 and higher

To obtain a certificate for which the private key is secretly kept in the TPM, your machine must be TPM enabled. To check that, run the following command:

```
ls -l /dev/tpm*
```

Two devices should be listed with corresponding ownership: - /dev/tpm0 (tss:root) - /dev/tpmrm0 (tss:tss)

If you are going to use the certificates obtained this way for Google Chrome, then the NSS DB in the current user space must be already present (see the previous section).

After the installation (and with initialized NSS DB if applicable), run the following command as NON-ROOT user:

```
/usr/local/bin/keytalk/ktconfig --rccd-path <path-to-rccd>
```

There are several use cases for obtaining TPM backed certificates (all as NON-ROOT user).

Obtaining a TPM backed certificate the first time requires both a PKCS11 SO (Security Officer) PIN and a User PIN to be specified:

```
/usr/local/bin/keytalk/ktclient --service <keytalk-template> --user <keytalk-user> --password <keytalk-password> --pkcs11-so-pin <so-pin> --pkcs11-user-pin <user-pin> [--pkcs11-cert-label <cert-label>]
```

After successful certificate retrieval (and indirectly local TPM provisioning), the following command can be used for retrieving additional certificates:

```
/usr/local/bin/keytalk/ktclient --service <keytalk-template> --user <keytalk-user> --password <keytalk-password> --pkcs11-user-pin <user-pin> [--pkcs11-cert-label <cert-label>]
```

In case the TPM is also used by other software and already provisioned with authority passwords/hashes specified elsewhere you also might need to provide those as well:

```
/usr/local/bin/keytalk/ktclient --service <keytalk-template> --user <keytalk-user> --password <keytalk-password> --pkcs11-so-pin <so-pin> --pkcs11-user-pin <user-pin> --tpm-owner-auth <auth-value> [--tpm-endorsement-auth <auth-value>] [--tpm-lockout-auth <auth-value>]
```

For offline mode (no Internet, only working connection with KeyTalk server), use the following command:

```
/usr/local/bin/keytalk/ktclient --service <keytalk-template> --user <keytalk-user> --password <keytalk-password> --pkcs11-so-pin <so-pin> --pkcs11-user-pin <user-pin> --tpm-offline-mode
```

In case the SO and/or User PINs are lost, additionally specify `-tpm-clean-provision` option:

```
/usr/local/bin/keytalk/ktclient --service <keytalk-template> --user <keytalk-user> --password <keytalk-password> --tpm-clean-provision --pkcs11-so-pin <so-pin> --pkcs11-user-pin <user-pin>
```

Listing/saving TPM info, installed keys and certificates (after TPM provision by one of the commands above):

```
/usr/local/bin/keytalk/ktclient --tpm-print-info  
/usr/local/bin/keytalk/ktclient --tpm-save-info <FILE_PATH.json>  
/usr/local/bin/keytalk/ktclient --tpm-list-objects  
/usr/local/bin/keytalk/ktclient --pkcs11-list-objects
```

Manipulating KeyTalk TPM NSS PKCS11 module:

```
/usr/local/bin/keytalk/ktclient --nss-list-modules  
/usr/local/bin/keytalk/ktclient --nss-add-keytalk-module  
/usr/local/bin/keytalk/ktclient --nss-remove-keytalk-module
```

Before uninstalling the KeyTalk agent, it is recommended to remove KeyTalk NSS PKCS11 module as NON-ROOT:

```
/usr/local/bin/keytalk/ktclient --nss-remove-keytalk-module
```

NOTE: The used keytalk-template must be configured for TPM attestation and the corresponding Endorsement certificate Vendor CAs must be uploaded.

NOTE: In case you have more than one provider installed for the current user, you may need to specify the `--provider` option as well in the use cases above.

Automatic renewal of user (seat) certificates (not Apache- or Tomcat-related)

This section explains how to configure KeyTalk agent to automatically renew user (seat) certificates. For renewal of Apache or Tomcat certificate, please refer to the different manuals.

Customize KeyTalk agent with `ktconfig`, if not yet

```
/usr/local/bin/keytalk/ktconfig --rccd-path <path-to-rccd>
```

Copy the sample renewal configuration file to the user's KeyTalk agent directory

```
cp /etc/keytalk/cert-renewal.ini ~/.keytalk/
```

Edit the copied configuration by enabling KeyTalk users (seats) for which the certificates have to be automatically renewed. Each provider and template should exist in the KeyTalk agent configuration (`~/.keytalk/user.ini` and `/etc/keytalk/master.ini`). See the hints in `~/.keytalk/cert-renewal.ini`

Alternatively, you can get away without `cert-renewal.ini`. In this case KeyTalk agent will request the list of templates to renew the certificates for from KeyTalk server. This option has its limitations, because it will assume that Kerberos authentication is in effect and no extra NSS/TPM options can be customized.

Test your setup

```
/usr/local/bin/keytalk/ktclient --renew-cert
```

The renewed certificate will be placed under `~/keytalk/keystore/` and additionally: - under the directory indicated by `ExtraCertSaveLocation` setting - under NSS certificate store (for Chrome browser) when `AddCertToNssDb` is supplied - when TPM is enabled server-side for the given template(s) and TPM is provisioned locally, KeyTalk PKCS11 TPM module is added to NSS certificate store

New certificate will only be requested from KeyTalk server when the locally stored certificate does not exist, is expired or is about to expire.

Finally, automate with cron job e.g.

```
echo -e '#!/bin/sh\n\n/usr/local/bin/keytalk/ktclient --renew-cert' | sudo tee /etc/cron.daily/keytalk-cert-renewal  
sudo chmod 644 /etc/cron.daily/keytalk-cert-renewal
```

Troubleshooting

For ktconfig-related errors

```
tail ~/tmp/ktconfig.log
```

For ktclient-related errors

```
tail ~/.keytalk/ktclient.log
```

For ktclinet-related errors with TPM usage

```
tail ~/.keytalk/ktclient.log  
cat ~/.keytalk/tpm-tss.log
```

I still can't solve my problem

Please create a problem report with

```
/usr/local/bin/keytalk/ktprgen
```

and send it to support@keytalk.com