

KeyTalk automated Apache certificate renewal remote deployment instructions

Introduction

KeyTalk provides automatic Apache certificate renewal functionality. To update and reconfigure a server farm with many servers however is tedious work. To automate deployments to many servers, the remote deployment tool allows you to: * Configure all virtual hosts of your server farm in one place * Install KeyTalk Apache certificate updater to each remote machine of your server farm, which includes: - Customizing your KeyTalk client with an RCCD file - Customizing your certificate renewal script configuration - Enabling periodic certificate renewal (by default checking every 10 minutes) * Remotely uninstall the KeyTalk client from a server

The following sections explain how to: * Prepare web servers * Configure Virtual Hosts * Deploy Remotely * Uninstall Remotely

Prepare web servers

First of all, make sure that all the servers you want to deploy to have the following software installed:

- Ubuntu 20.04 LTS x64 or Ubuntu 22.04 LTS x64 or Ubuntu 24.04 LTS x64
- Apache 2.2 - 2.4

This deployment script requires password-less SSH access to the machines you want to deploy to.

Setting up password-less SSH logins requires the following steps:

1. Generate an SSH key pair (one-time, only if you don't have one yet)
2. Copy your identity (public key) to all remote servers
3. Cache your key passphrase using ssh agent (only if you have a passphrase)

For example:

```
$ ssh-keygen # only if there is no ssh key pair yet

$ ssh-copy-id admin@10.0.0.1
admin@10.0.0.1's password: ****
Number of key(s) added: 1

$ ssh-copy-id admin@10.0.0.2
admin@10.0.0.1's password: ****
Number of key(s) added: 1

...

$ eval `ssh-agent -s`
$ ssh-add
Enter passphrase for /home/me/.ssh/id_rsa: ****
Identity added: /home/me/.ssh/id_rsa (/home/me/.ssh/id_rsa)
```

Log in to the each remote server

```
$ ssh admin@10.0.0.1
(no password should be asked)

Grant password-less sudo rights to the logged-in user (skip this if you use direct root access)
$ echo "$USER ALL=(ALL:ALL) NOPASSWD: ALL" | sudo tee /etc/sudoers.d/$USER
```

More information on how to use ssh-copy-id: <http://www.lindonslog.com/linux-unix/ssh-keygen-keys/>

More information on how to manually copy SSH keys: <http://mah.everybody.org/docs/ssh>

Create mass deployment configuration file

The configuration format of the deployment configuration is the same as apache.ini described in [KeyTalk automated Apache certificate renewal installation instructions](#) with an addition of the RemoteHost setting for each virtual host.

The `RemoteHost` should contain `<user>@<host>` (e.g. `keytalk@10.0.0.1`). The value should be a user/host combination to which password-less ssh logins are enabled.

The following sample configuration contains deployment on 2 servers:

```
[
{
  "RemoteHost" : "keytalk@10.0.0.1",
  "VHost" : "_default_:443",
  "KeyTalkProvider" : "MyProvider",
  "KeyTalkService" : "MY_SERVICE",
  "KeyTalkUser" : "a.example.com"
},
{
  "RemoteHost" : "keytalk@10.0.0.2",
  "VHost" : "_default_:443",
  "KeyTalkProvider" : "MyProvider",
  "KeyTalkService" : "MY_SERVICE",
  "KeyTalkUser" : "b.example.com"
}
]
```

The following sample configuration contains 2 name-based Virtual Hosts hosted on the same server:

```
[
{
  "RemoteHost" : "keytalk@10.0.0.1",
  "VHost" : "_default_:443",
  "ServerName" : "a.example.com",
  "KeyTalkProvider" : "MyProvider",
  "KeyTalkService" : "MY_SERVICE",
  "KeyTalkUser" : "a.example.com"
},
{
  "RemoteHost" : "keytalk@10.0.0.1",
  "VHost" : "_default_:443",
  "ServerName" : "b.example.com",
  "KeyTalkProvider" : "MyProvider",
  "KeyTalkService" : "MY_SERVICE",
  "KeyTalkUser" : "b.example.com"
}
]
```

Save your deployment configuration to `deployment.ini`

Deploy Remotely

After preparing your servers for password-less SSH login and configuring your VHosts you can use the following command to start deployment:

```
$ ./ktclient_remote_deploy install-for-apache /path/to/deployment.ini /path/to/KeyTalkClient-X.Y.Z-ubuntu-version.tgz /path/to/rccd-file.rccd | tee install.log
```

If deployment on one of the hosts fails, the deployment script will attempt to uninstall the failed installation and continue with the next host.

Uninstall Remotely

To uninstall an installed KeyTalk linux client remotely you also need to have password-less SSH access to the system.

To uninstall remotely, you can use the following command:

```
$ ./ktclient_remote_deploy remove <user>@<host>
```

For example:

```
$ ./ktclient_remote_deploy remove keytalk@10.0.0.1
```