#### Table of contents

# Chapter 15. Configuring custom SSL/TLS certificates

You can configure the undercloud to use SSL/TLS for communication over public endpoints. However, if want to you use a SSL certificate with your own certificate authority, you must complete the following configuration steps.

## 15.1. Initializing the signing host $\mathscr{S}$

The signing host is the host that generates and signs new certificates with a certificate authority. If you have never created SSL certificates on the chosen signing host, you might need to initialize the host so that it can sign new certificates.

#### **Procedure**

1. The /etc/pki/CA/index.txt file contains records of all signed certificates. Check if this file exists. If it does not exist, create an empty file:

```
$ sudo touch /etc/pki/CA/index.txt
```

2. The /etc/pki/CA/serial file identifies the next serial number to use for the next certificate to sign. Check if this file exists. If the file does not exist, create a new file with a new starting value:

\$ echo '1000' | sudo tee /etc/pki/CA/serial

## 15.2. Creating a certificate authority &

Normally you sign your SSL/TLS certificates with an external certificate authority. In some situations, you might want to use your own certificate authority. For example, you might want to have an internal-only certificate authority.

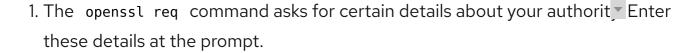
#### **Procedure**

1. Generate a key and certificate pair to act as the certificate authority:

\$ openssl genrsa -out ca.key.pem 4096

\$ openssl req -key ca.key.pem -new -x509 -days 7300 -extensions





These commands create a certificate authority file called ca.crt.pem .

## 15.3. Adding the certificate authority to clients &

For any external clients aiming to communicate using SSL/TLS, copy the certificate authority file to each client that requires access to your Red Hat OpenStack Platform environment.

#### **Procedure**

1. Copy the certificate authority to the client system:



2. After you copy the certificate authority file to each client, run the following command on each client to add the certificate to the certificate authority trust bundle:

\$ sudo update-ca-trust extract

## 15.4. Creating an SSL/TLS key &

Enabling SSL/TLS on an OpenStack environment requires an SSL/TLS key to generate your certificates. This procedure shows how to generate this key.

#### **Procedure**

1. Run the following command to generate the SSL/TLS key ( server.key.pem ):

## 15.5. Creating an SSL/TLS certificate signing request



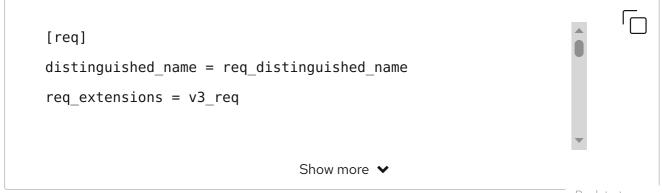
Complete the following procedure to create a certificate signing request.

#### **Procedure**

1. Copy the default OpenSSL configuration file:

```
$ cp /etc/pki/tls/openssl.cnf .
```

2. Edit the new openssl.cnf file and configure the SSL parameters to use for the director. An example of the types of parameters to modify include:



Set the <code>commonName\_default</code> to one of the following entries:

- If using an IP address to access the director over SSL/TLS, use the undercloud public host parameter in undercloud.conf .
- If using a fully qualified domain name to access the director over SSL/TLS, use the domain name.

Add subjectAltName = @alt names to the v3 req section.

Edit the alt names section to include the following entries:

- IP A list of IP addresses that clients use to access the director over SSL.
- DNS A list of domain names that clients use to access the director over SSL.
   Also include the Public API IP address as a DNS entry at the end of the alt\_names section.

Note

25/05/2025, 18:35

3. Run the following command to generate a certificate signing request (server.csr.pem):

command.

```
$ openssl req -config openssl.cnf -key server.key.pem -new -out
server.csr.pem
```

Ensure that you include your OpenStack SSL/TLS key with the -key option. This command results in an server.csr.pem file, which is the certificate signing request. Use this file to create your OpenStack SSL/TLS certificate.

## 15.6. Creating the SSL/TLS certificate &

This procedure shows how to generate the certificate for your OpenStack environment. This requires the following files:

#### openssl.cnf

The customized configuration file specifying the v3 extensions.

#### server.csr.pem

The certificate signing request to generate and sign the certificate with a certificate authority.

#### ca.crt.pem

The certificate authority, which signs the certificate.

#### ca.key.pem

The certificate authority private key.

#### **Procedure**

1. Run the following command to create a certificate for your undercloud or overcloud:

```
$ sudo openssl ca -config openssl.cnf -extensions v3_req -days
3650 -in server.csr.pem -out server.crt.pem -cert ca.crt.pem -
keyfile ca.key.pem
```

This command uses the following options:

#### -config

Use a custom configuration file, which is our openssl.cnf file with v3 extensions.

#### -extensions v3\_req

Enabled v3 extensions.

#### -days

Defines how long in days until the certificate expires.

#### -in '

The certificate signing request.

#### -out

The resulting signed certificate.

#### -cert

The certificate authority file.

### -keyfile

The certificate authority private key.

This command creates a new certificate named server.crt.pem . Use this certificate in conjunction with your OpenStack SSL/TLS key

## 15.7. Adding the certificate to the undercloud $\mathscr{D}$

Complete the following steps to add your OpenStack SSL/TLS certificate to the undercloud trust bundle.

#### **Procedure**

1. Run the following command to combine the certificate and key:

```
$ cat server.crt.pem server.key.pem > undercloud.pem

This command creates a undercloud.pem file.
```

 $https://docs.red hat.com/en/documentation/red\_hat\_openstack\_platform/15/html/director\_installation\_and\_usage/configuring-...$ 

2. Copy the undercloud.pem file to a location within your /etc/pki directory and set the necessary SELinux context so that HAProxy can read it:



3. Add the undercloud.pem file location to the undercloud\_service\_certificate option in the undercloud.conf file:

```
undercloud_service_certificate = /etc/pki/undercloud-
certs/undercloud.pem
Back to top
```

4. Ensure you add the certificate authority that signed the certificate to the undercloud's list of trusted Certificate Authorities so that different services within the undercloud have access to the certificate authority:

```
$ sudo cp ca.crt.pem /etc/pki/ca-trust/source/anchors/
$ sudo update-ca-trust extract
```

Continue installing the undercloud.

Previous Next

Back to top



About Red Hat Documentation

We help Red Hat users innovate and achieve their goals with our products and services with content they can trust. Explore our recent updates.

Making open source more inclusive

#### 25/05/2025, 18:35 Chapter 15. Configuring custom SSL/TLS certificates | Director Installation and Usage | Red Hat OpenStack P...

Red Hat is committed to replacing problematic language in our code, documentation, and web properties. For more details, see the <u>Red Hat Blog</u>.

#### About Red Hat

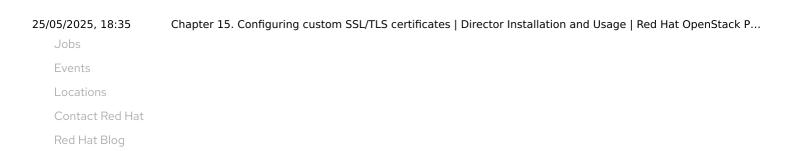
We deliver hardened solutions that make it easier for enterprises to work across platforms and environments, from the core datacenter to the network edge.

Theme

System default



About Red Hat Back to top



#### All systems operational

© 2025 Red Hat, Inc.

Inclusion at Red Hat
Cool Stuff Store
Red Hat Summit

Privacy statement

Terms of use

All policies and guidelines

Digital accessibility

Cookie preferences