

# JupyterHub - Spawn failed - SSL: CERTIFICATE\_VERIFY\_FAILED

Full error code, which is displayed after you log in and JupyterHub tries to spawn a pod for you:



```
Spawn failed: Cannot connect to host 10.152.183.1:443
ssl:True [SSLCertVerificationError: (1, '[SSL:
CERTIFICATE_VERIFY_FAILED] certificate verify failed:
certificate has expired (_ssl:c:1006)')]
```

Some specific numbers might be different like the IP for example.

It could be solved by ssh'ing into the VM running Kubernetes (in our case it was our node00).

To solve the certificate expiration issue we need to first list and see which certificates have expired.

To list certificates in microk8s:

```
sudo microk8s refresh-certs -c
```

This command will output a list of certificates and the time left for their expiration. The output looks like this:

```
The CA certificate will expire in 3284 days.
The server certificate will expire in 364 days.
The front proxy client certificate will expire in 364 days.
```

In our case the server certificate and the front proxy certificate had expired

To renew the certificates we run the following commands:

```
sudo microk8s refresh-certs -e server.crt
sudo microk8s refresh-certs -e front-proxy-client.crt
```

With these commands the certificates are updated and the services are automatically restarted.

Source: <https://discuss.kubernetes.io/t/renew-certificate-in-microk8s-cluster/19826>

From:  
<https://wiki.eolab.de/> - HSRW EOLab Wiki

Permanent link:  
[https://wiki.eolab.de/doku.php?id=eolab:crunchy\\_cloud:issues:jupyterhub-spawn-failed-certificate-verify-failed](https://wiki.eolab.de/doku.php?id=eolab:crunchy_cloud:issues:jupyterhub-spawn-failed-certificate-verify-failed)

Last update: 2024/12/18 13:46

