Recipe for Generating Digital Certificates with Post-Quantum Cryptography Using INeS

Introduction:

OpenQuantumSafe (OQS) is an open-source project designed to integrate post-quantum cryptography into widely used cryptographic libraries like OpenSSL. With quantum computing on the horizon, OQS provides tools to future-proof security implementations. This recipe will guide you through generating a key and certificate request using the OQS OpenSSL provider, leveraging the Dilithium 3 algorithm, and sign the certificate using a Certification Authority in the INeS platform.

Ingredients:

- An INeS account (see the recipe to create your Free trial account!)
- A computer capable of running Docker.
- A command-line interface (CLI).
- Basic knowledge of OpenSSL commands (recommended)

Instructions:

Step 1: Ensure you have a working Docker environment

1. Open your command line and run this command:

```
docker run hello-world
```

You should see a message like:

```
Hello from Docker!
This message shows that your installation appears to be
working correctly.
```

2. Depending on the result:

- If the test fails, check the available documentation on how to install Docker in your particular system. For Windows and macOS, we recommend you to install “**Docker Desktop**”
- If the test works, you’re good to continue with the recipe!

Step 2: Install and run the OpenQuantumSafe docker image

1. Open your terminal or CLI.
2. Pull the OQS OpenSSL Docker image from the Docker registry by running:

```
docker pull openquantumsafe/oqs-ossl3
```

3. Verify the image is downloaded by listing your Docker images:

```
docker images
```

4. Start a Docker container with the OQS image:

```
docker run -it openquantumsafe/oqs-ossl3 /bin/sh
```

5. Try to run OpenSSL with the command:

```
openssl –version
```

You should get an output similar to:

```
OpenSSL 3.5.0-dev  (Library: OpenSSL 3.5.0-dev )
```

If you get an error, then find where OpenSSL is located (i.e. with the command “” and navigate to the right path with something like:

```
cd /opt/openssl32/bin
```

Once you can run OpenSSL in your docker container, you can continue with the next step

Step 3: Generate a private key and CSR using the ML-DSA65 (Dilithium 3) algorithm:

1. Generate the private key with the command:

```
openssl genpkey -algorithm mldsa65  -out private_key.pem
```

2. Create a CSR (Certificate Signing Request) using the private key:

```
openssl req -new -key private_key.pem -out csr.pem -subj
"/CN=device-name-12345/O=Maker2025/C=CH"
```

Replace `device-name-12345`, `Maker2025`, and `CH` with the names you want for your certificate, organization name, and country code, respectively.

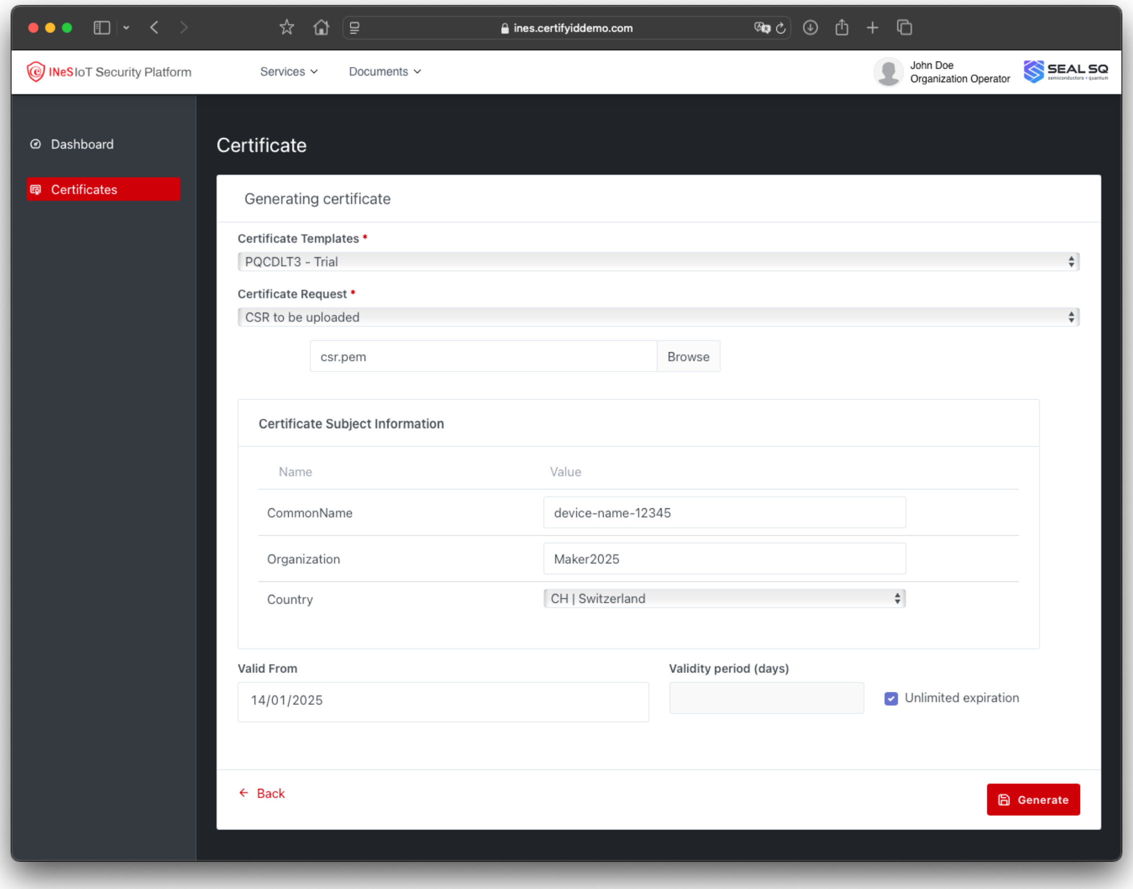
3. Copy the CSR.pem file out of the container. This can be done easily in Docker Desktop using the “Files” tab, but you can also simply copy the content of the CSR and paste it in a text editor in your host OS, with a command such as:

```
cat csr.pem
```

You will use this file in the last step to generate the certificate using INeS.

Step 4: Generate the certificate using INeS

1. Log in INeS using your account and open the “Certificate Management” service
2. Select the “Certificates” menu option and click on “New”
3. Enter the certificate request details. You can simply load the “csr.pem” file and modify any detail If needed:



Certificate

Generating certificate

Certificate Templates *
PQCDLT3 - Trial

Certificate Request *
CSR to be uploaded

csr.pem

Certificate Subject Information

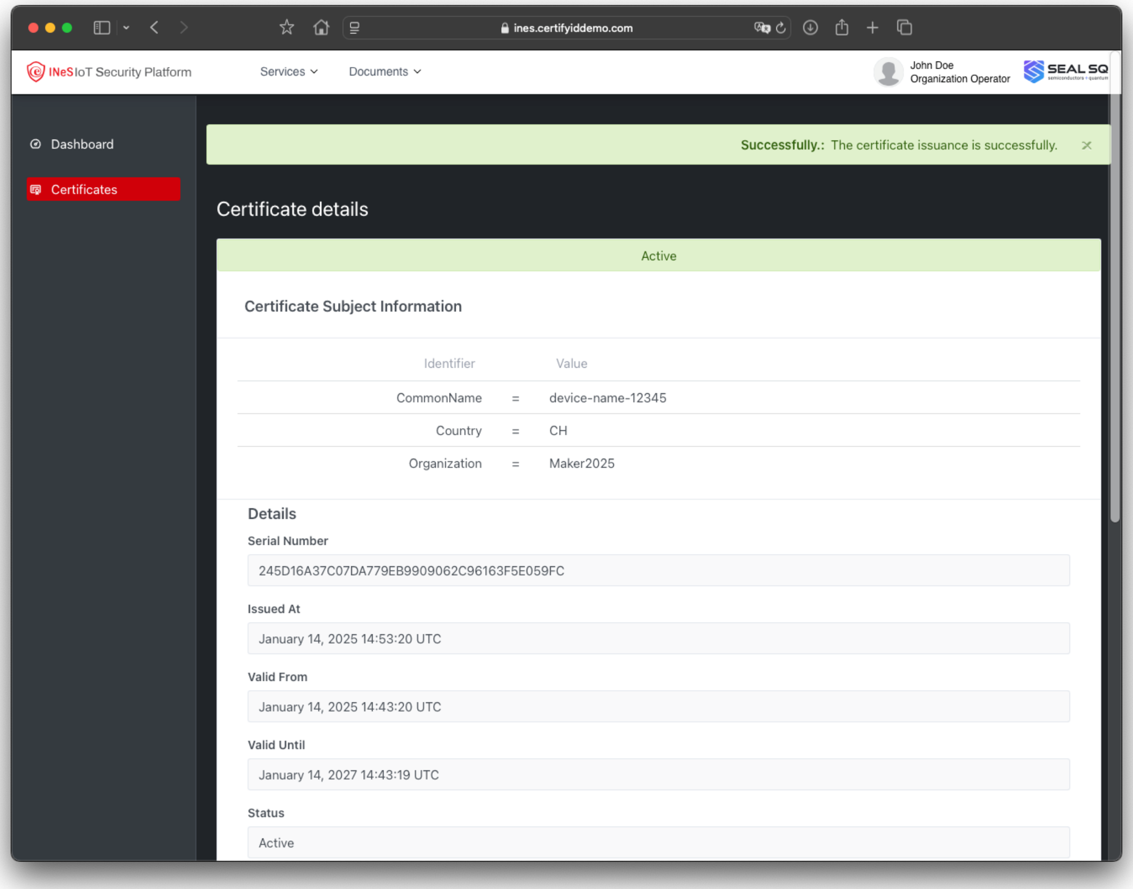
Name	Value
CommonName	device-name-12345
Organization	Maker2025
Country	CH | Switzerland

Valid From: 14/01/2025

Validity period (days):

☑ Unlimited expiration

4. Click on “Generate” and you got your PQC Certificate!



Successfully.: The certificate issuance is successfully. ×

Certificate details

Active

Certificate Subject Information

Identifier		Value
CommonName	=	device-name-12345
Country	=	CH
Organization	=	Maker2025

Details

Serial Number: 245D16A37C07DA779EB9909062C96163F5E059FC

Issued At: January 14, 2025 14:53:20 UTC

Valid From: January 14, 2025 14:43:20 UTC

Valid Until: January 14, 2027 14:43:19 UTC

Status: Active

Conclusion:

By following this recipe, you have successfully generated a digital certificate using the Dilithium 3 post-quantum algorithm through OpenQuantumSafe's OpenSSL implementation and INeS. This process demonstrates how containerized environments simplify the adoption of cutting-edge cryptographic tools. Your new certificate is now ready to secure your digital communications, providing resilience against potential quantum threats. Explore further by integrating this certificate into your web servers or secure communication protocols.