Zertifikate

- .Net 6 - https
- openssl - selfsign - certificate

.Net 6 - https

Problembehebung beim Starten mit https

1. dotnet dev-certs https --clean
2. dotnet dev-certs https --verbose
3. dotnet dev-certs https --trust

openssl - selfsign - certificate

To create a self-signed PFX certificate using OpenSSL, you can follow these steps:

1. **Generate a private key**:

You can generate a private key using the `openssl genpkey` command. Here, I'll generate a 2048-bit RSA private key and save it to a file named `private.key`:

```
openssl genpkey -algorithm RSA -out private.key -aes256
```

This command will prompt you to enter a passphrase to protect the private key.

2. **Create a certificate signing request (CSR)**:

You can create a CSR using the private key generated in the previous step. This CSR will be used to generate the self-signed certificate:

```
openssl req -new -key private.key -out certificate.csr
```

Follow the prompts to fill in the information for your certificate.

3. **Generate a self-signed certificate**:

Now, you can generate a self-signed certificate using the CSR and the private key:

```
openssl x509 -req -days 365 -in certificate.csr -signkey private.key -out certificate.crt
```

This command will generate a self-signed certificate valid for 365 days.

4. **Create the PFX file**:

To create a PFX file (also known as PKCS#12), you need both the private key and the certificate:

```
openssl pkcs12 -export -out certificate.pfx -inkey private.key -in certificate.crt
```

This command will prompt you to enter a passphrase to protect the PFX file.

Now, you have a self-signed PFX certificate file named `certificate.pfx`. Make sure to securely store both the private key and the PFX file, as they contain sensitive information.