Create Certificate Signing Request (CSR) – November 1, 2025

To request a PKIoverheid Private Server Certificate, Digidentity offers the option to submit your own Certificate Signing Request (CSR). This allows administrators to generate their own private key and produce a CSR.

Digidentity also supports generating the CSR directly in your browser.

CSR Requirements

A CSR for PKIoverheid Private Server Certificates must include the following fields:

- commonName = [name of the primary server]
- organizationName = [organization name as registered in the Chamber of Commerce]
- organizationIdentifier = [OIN (Organization Identification Number)]
- serialNumber = [OIN (Organization Identification Number)]
- country = [two-letter country code, e.g., NL]
- subjectAltName = [server names – up to ten]*

*The first server name listed in subjectAltName must match the server name in commonName.

To generate a CSR, you'll need to know how to use OpenSSL. You'll be required to run a few commands to create your private key, set up a configuration file, and generate your CSR.

Below is a step-by-step guide based on an example case:

CSR Creation Instructions

Create a server certificate for the company Example B.V., with registration number '12345679' in the Chamber of Commerce (KvK), for the server name 'server1.voorbeeld.nl'.

An OIN (Organization Identification Number) is required for any server certificate. If your organization already has an OIN, enter it here. If not, you can convert your KvK registration number to an OIN. An OIN based on the KvK number starts with00000003 followed by your registration number and ends with 0000.

00000003 12345678 0000 = OIN = 00000003123456780000

[1] Page 2 of 3 Create a plain text file (using Notepad, TextEdit, or Sublime Text) named "voorbeeldbv.cnf". Add the following content to the file:

```
oid_section = OIDs

[ req ]
default_bits            = 2048
encrypt_key             = no
default_md              = sha256
utf8                    = yes
string_mask             = utf8only
prompt                          = no
distinguished_name              = req_distinguished_name
req_extensions                  = req_ext
[ OIDs ]
organizationOID=2.5.4.97
[ req_distinguished_name ]
countryName = NL
organizationName = Voorbeeld B.V.
organizationOID = 00000003123456780000
serialNumber = 00000003123456780000
commonName = server1.voorbeeld.nl
[ req_ext ]
subjectAltName = @alt_names
[alt_names]
DNS.1 = server1.voorbeeld.nl
```

Save the voorbeeldbv.cnf file. IMPORTANT: The information in the file, such as organization name, OIN, country, and server domain name, must exactly match the details verified by Digidentity. If any information does not match, your CSR will be rejected.

[2] Generate a private key (key length 2048) using the following command:

```
openssl genrsa -out voorbeeldbv.key 2048
```

This command will create a file named "voorbeeldbv.key". All files must be in the same folder when creating a CSR.

[3] Next, create your CSR by running the following command:

```
openssl req -new -sha256 -out voorbeeldbv.csr -key voorbeeldbv.key -config voorbeeldbv.cnf
```

This command generates the file "voorbeeldbv.csr" using the configuration file "voorbeeldbv.cnf" and the key file "voorbeeldbv.key".

You can upload the "voorbeeldbv.csr" file to Digidentity.

Adding extra server names to your certificate

You may include up to ten (10) server names for the domain in your certificate. The first server name appears in both the CommonName and DNS.1 fields. For each additional server name, add a DNS.# entry in the file up to DNS.10.

[4] If you want to create a CSR with three server names (server.voorbeeld.nl, applicatie.voorbeeld.nl, and eherkenning.voorbeeld.nl), include the following content in the alt_names section of your file:

```
[alt_names]
DNS.1        = server.example.nl
DNS.2        = application.example.nl
DNS.3        = eherkenning.example.nl
```

Digidentity will review the contents of your CSR. If the organization name, country, OIN, and domain name match the verified details, you can have your CSR signed and then download the certificate (signed CSR) as a .PEM file.