XProtect Mobile Server — Certificates guide

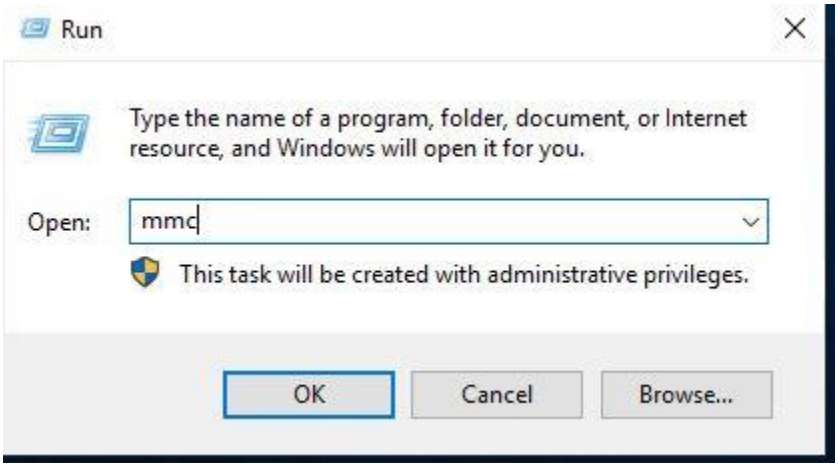
A certificate has two key parts — the public key, and the private key. The private key never leaves your machine except in very rare situations. For the purposes of this guide, it is just the **Mobile Server** we will be dealing with, and all steps are to be performed locally where the Mobile Server is installed.

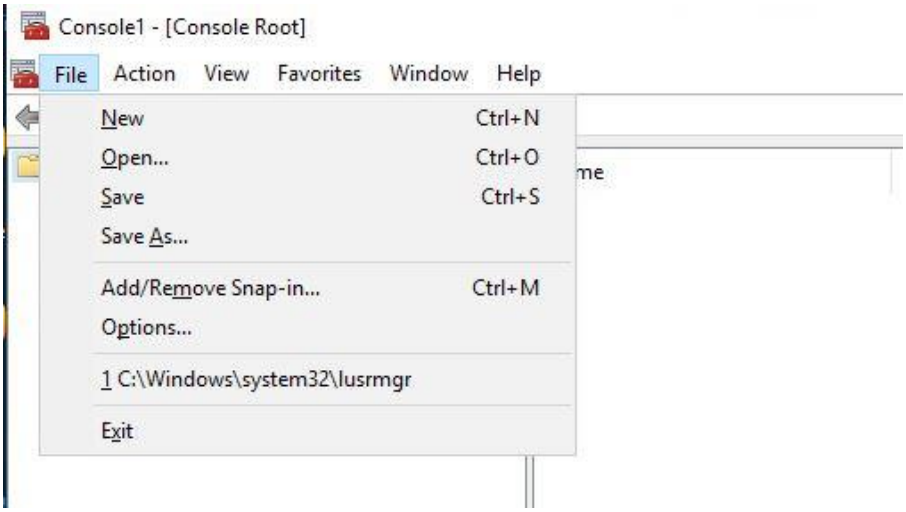
First, we need a domain that is going to be registered to the public external IP address. For DDNS you will need one that supports the certificate management. You need to register your domain for your IP address with a domain registrar (such as GoDaddy, Bluehost, Dreamhost, etc.).

Make sure it is **local machine** at the top, not **current user**.

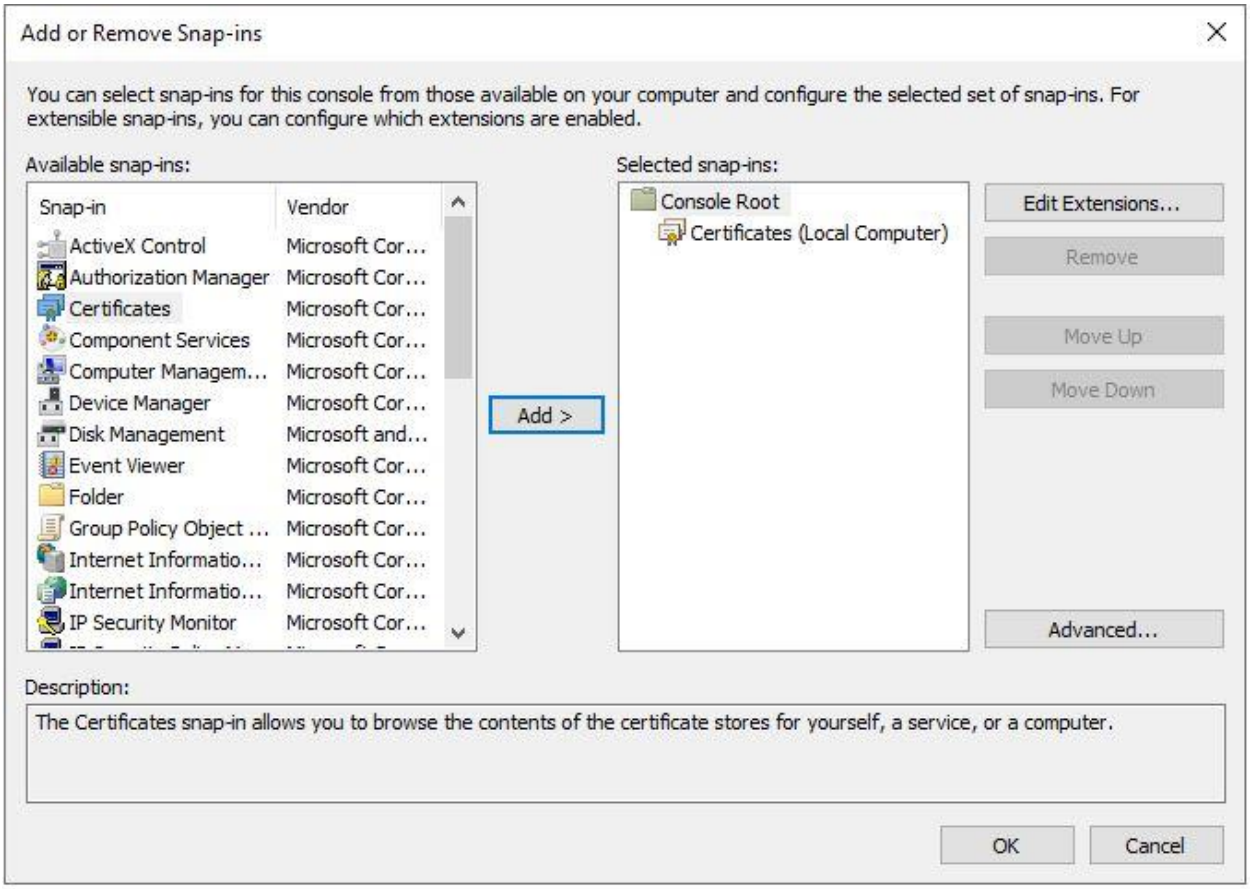
To get this started, we need to create a new certificate signing request.

Run `mmc` and add the following snap-ins by going to **File** → **Add/Remove Snap-ins**:





Select **Certificates** and press **Add**, selecting **Service account** then **Local Computer**:

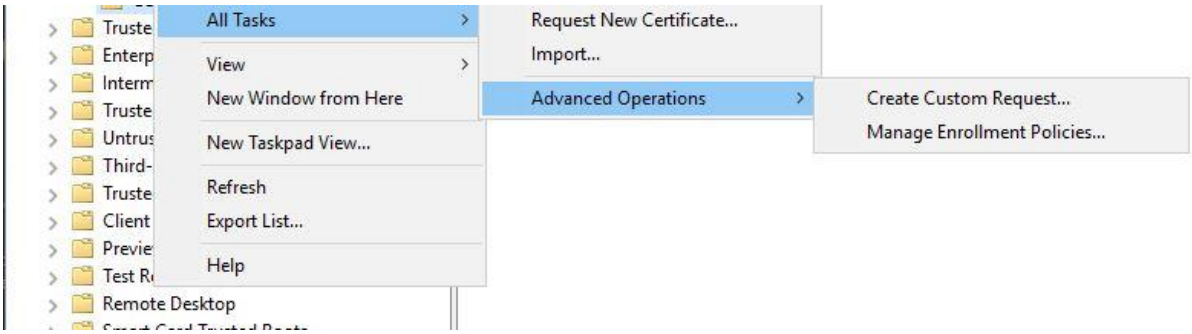


Press **OK**. It should say **(Local Computer)** next to **Certificates**.

Expand the **Personal** object in the tree and click on **Certificates**.

Right-click the **Certificates** folder under **Personal**.

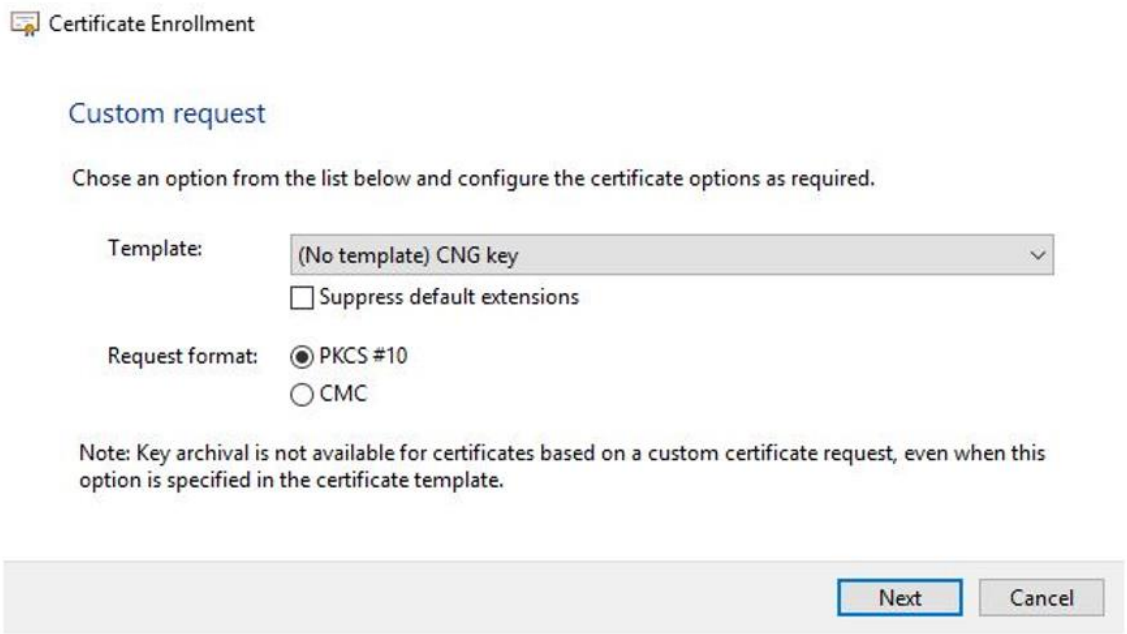
Select **All Tasks** → **Advanced Operations** → **Create Custom Request**.



The screen **Before you Begin** should appear, select **Next**.

Proceed without enrollment policy should be the only thing there under **Custom Request**.

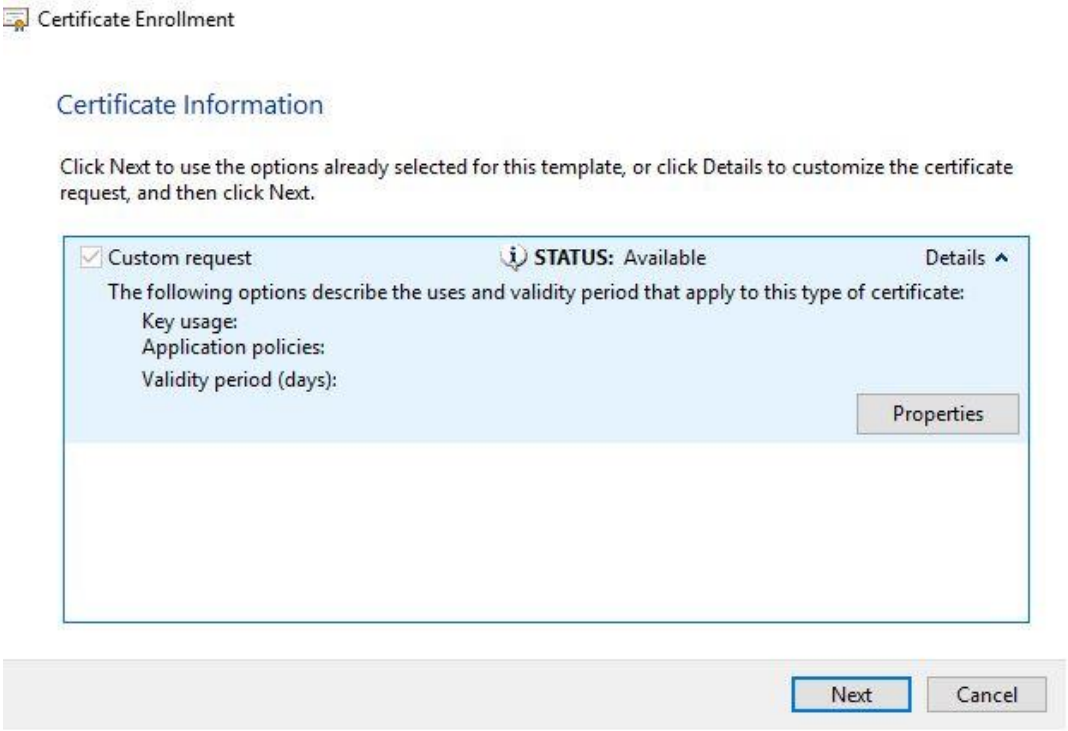
Press **Next**. The template usually used is **(No template) CNG key**.



The **Request format** is largely dependent on the Certificate Authority. If you submit a wrong format or missing property they may return an **Invalid** format or **Error**.

We used GoDaddy and they require **PKCS#10** — https://www.godaddy.com/community/SSL-And-Security/Guide-for-CSR-on-Windows-10/td-p/166745

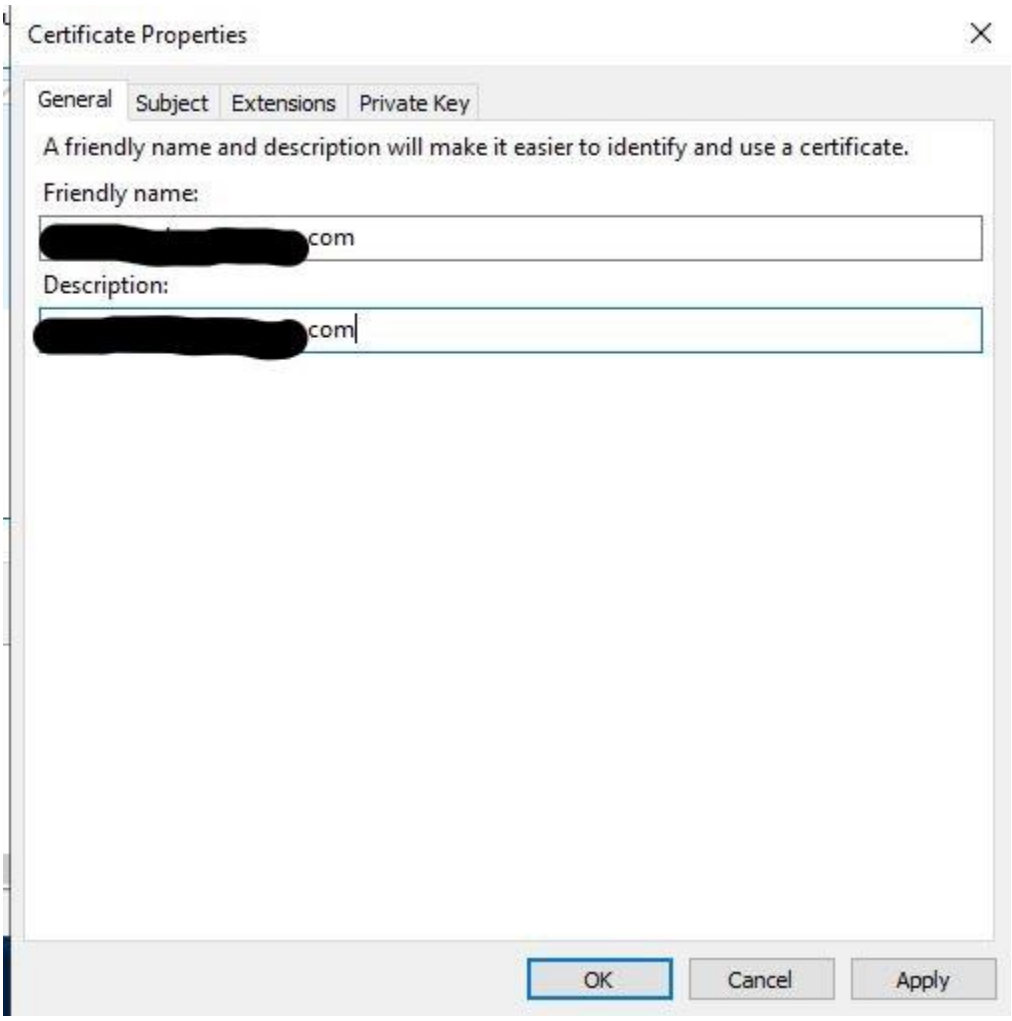
Press **Next** and expand the **Details** by hitting the little arrow on the right-hand side then go to **Properties**.



Note: For easier management purposes, **Friendly name** and **Description** are best to be the **same** as the domain name.

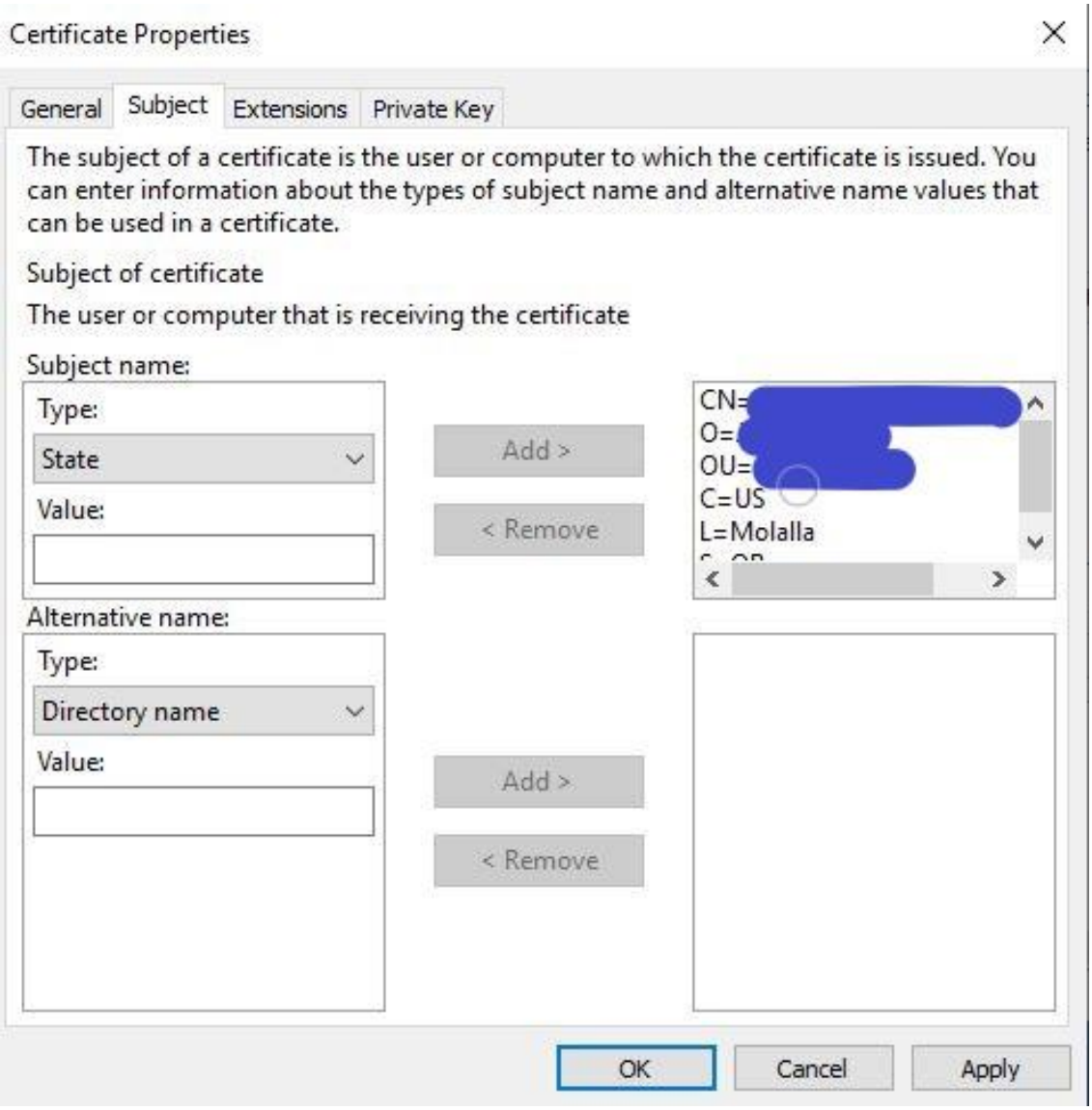
Friendly name: `Myexampledomain.com`

Description: `Myexampledomain.com`



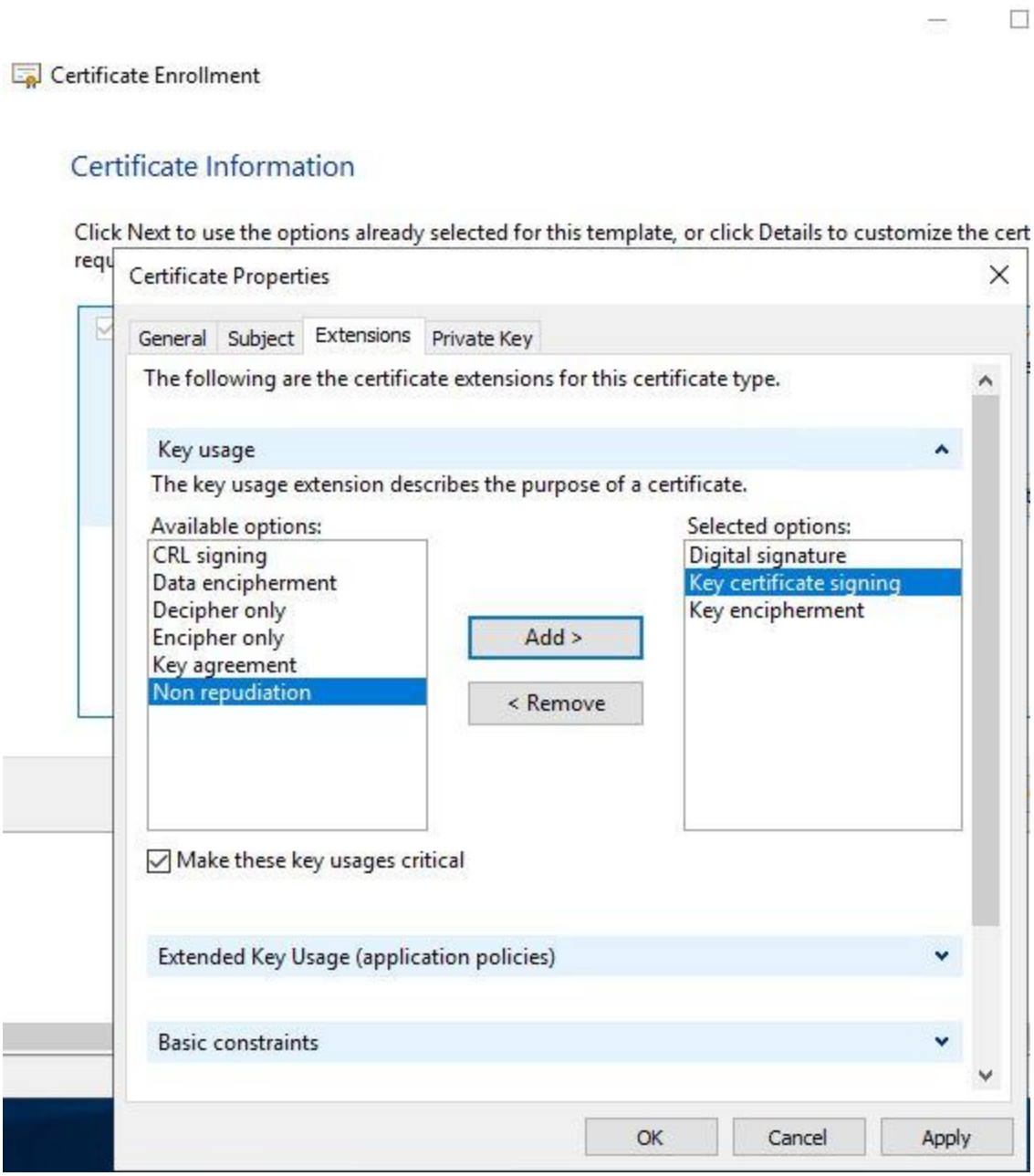
Next, go to the **Subject** tab. The **Subjects** needed are different based on the Certificate Authority, in our case GoDaddy requires the following:

```
CommonName=Myexampledomain.com

Organization=Mycompany

OrganizationalUnit=Mycompany

Country=US

Locality=Mytown

State=LA (full state or postal code for state)
```



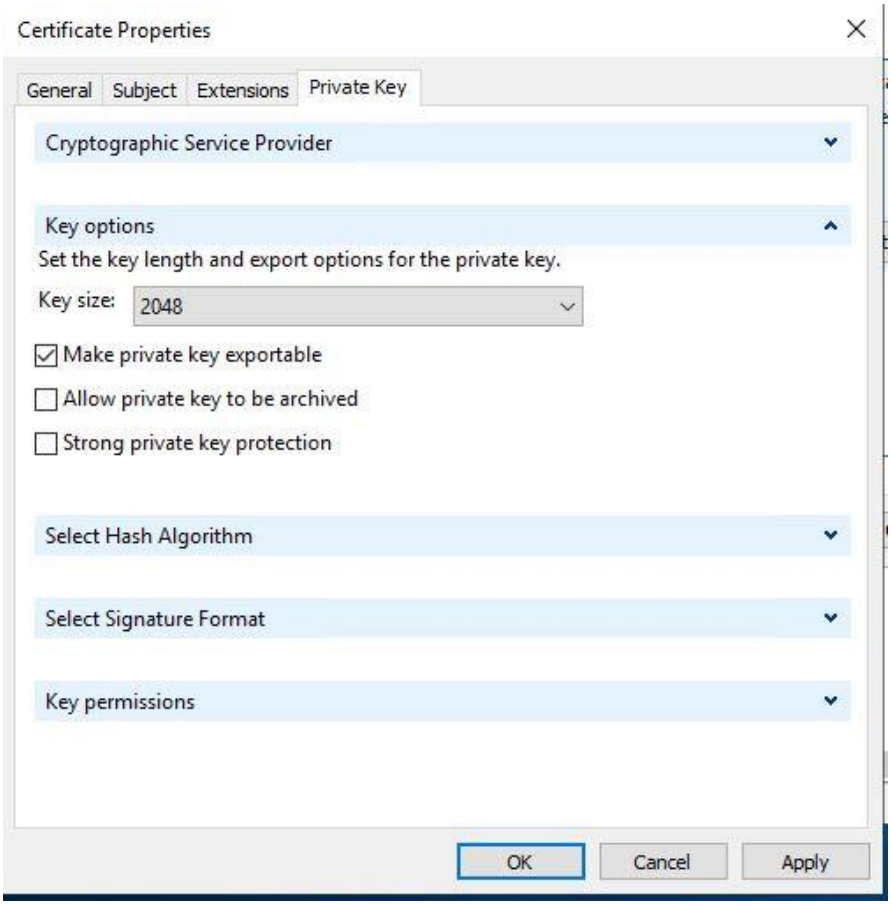
Next, under **Extensions** you will need to select **Key usage** and they require:

- Digital Sig
- Key Encipherment
- Key certificate Signing



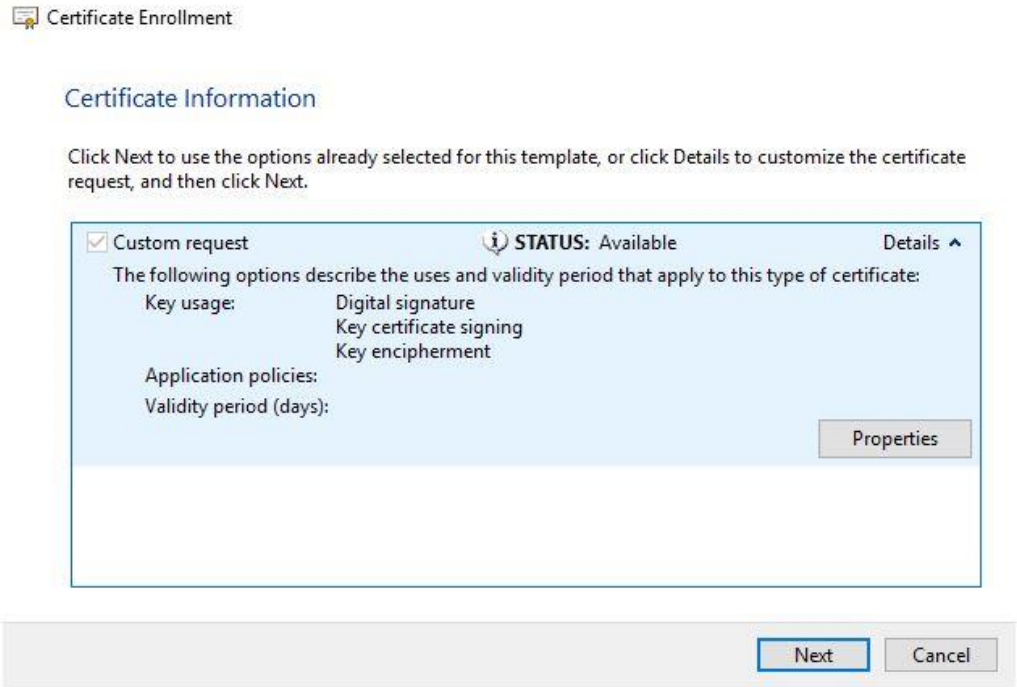
Now on the **Private Key** tab, go to **Key Options** and expand it, selecting **Key Size 2048**.

Note: Some providers may require a higher key size!



Check the option **Make private key exportable**.

Press **OK**. You should see the following:



Press **Next** then **Browse** and select a place to save it.

We saved it on the Desktop as “CSR” with the file extension `*.req`.

Default file format is **Base64**, some may require **Binary** — this is usually defined by the CA.

Next, open the file with Notepad and you should see a block of text:

You will need to upload the block of text *including* the **BEGIN** and **END** lines.

This block of text is what usually needs to be uploaded to the CA.

Once uploaded — and if it doesn't return an error (in case you missed a property field or used a wrong format) — the Certificate Authority should notify you when the certificate is ready.

Some CA’s will email you the files as a `*.zip.tz`, or in `*.tar` format. We didn’t select IIS this time, we selected “Other”.

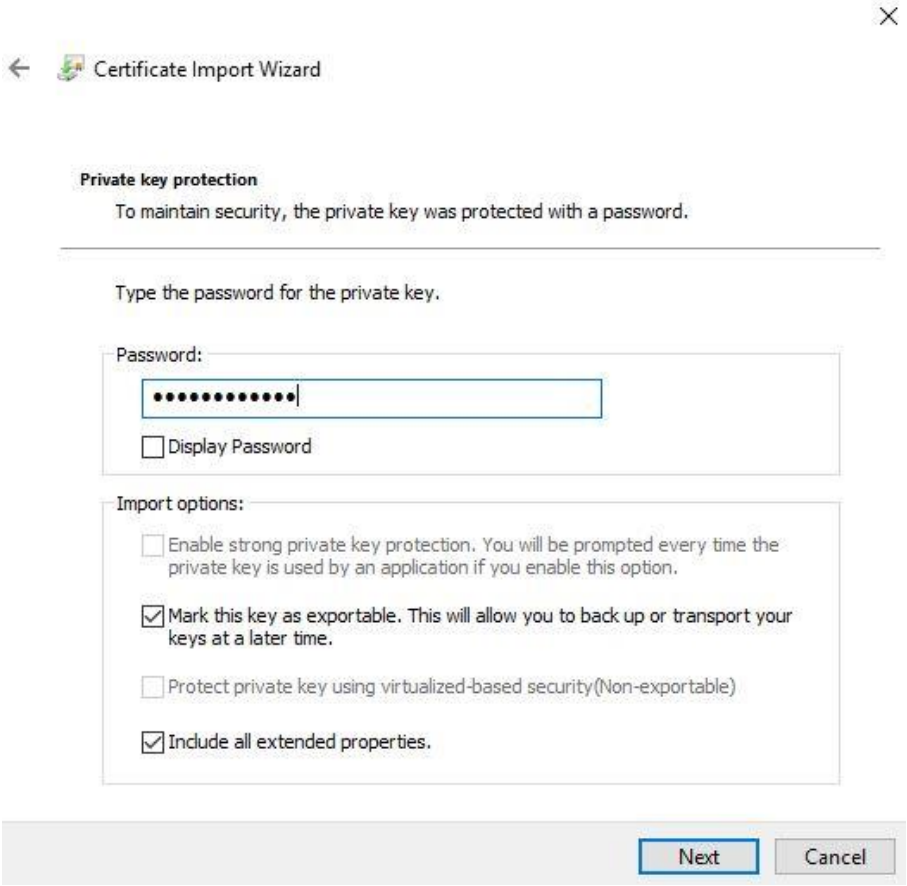
Once received, you will need to extract all contents.

In our case there were three files and the one at the top with the string of letters and numbers that ended in `*.crt`.

We right-clicked the file and selected **Install**. We then made sure it says **Local Machine**.

Note: It should autofill the path for you since it was right-clicked on.

Sometimes if you have a pair of PEM’s that were converted to a `*.pfx` file you have to set a password and importing requires a password. This may not be the case if they sent you a `*.cer` or `*.crt` file.



The option **Mark this key as exportable** needs to be checked so you would be able to backup the cert and key associated with it. Press **Next**.

Change the option from **Automatic** to **Place all certificates in the following store** and select **Personal**.

You should see a summary window where you can press **Finish**.

The last step is to go to the **Mobile Server** and selecting **Edit Certificate**.

(In older versions of XProtect you select the cert and it has a field to enter the password for the cert, but newer versions you get a pop-up and you just select the cert since it was imported with the password.)

Test your connection to ensure it works.

On the **XProtect Management Server** you may also need to go to **Connectivity** and select **Disable Default Address** to prevent the internal FQDN from causing a cert error.

This will allow you to have only one mobile profile for the mobile app for inside and outside the network.

The network engineer can also set up NAT “Hairpinning” or “Reflection” so it can forward the public external domain name of the XProtect Mobile Server locally without the request leaving the network and travelling all the way back in.