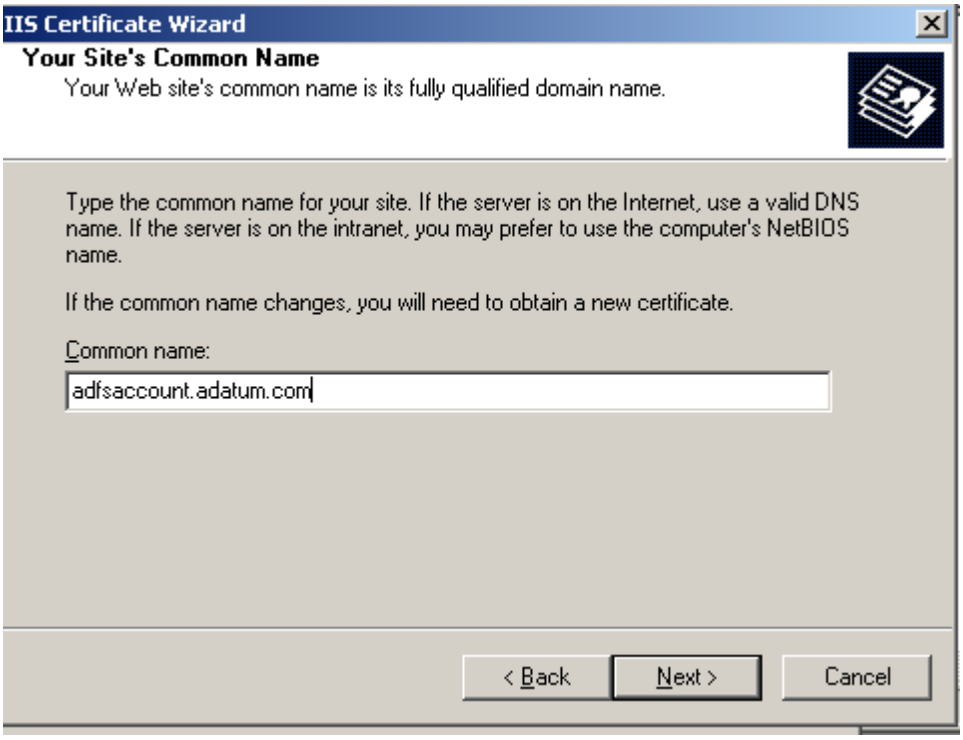
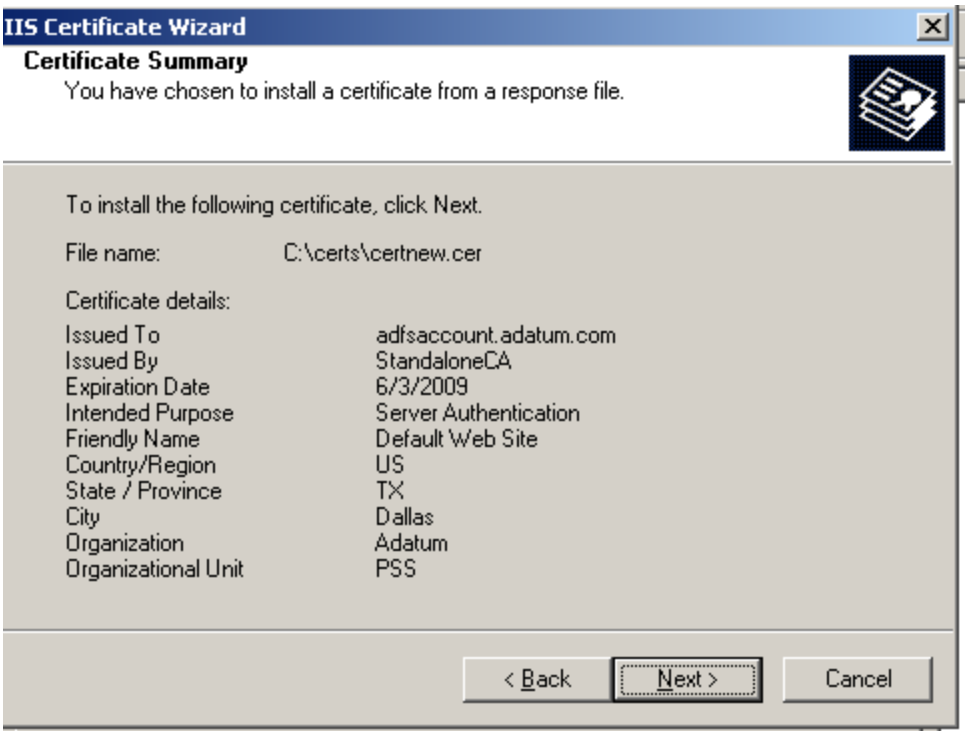
In this blog, I will discuss the steps needed to add an ADFS Proxy to your environment. I will also outline a couple of gotchas that I ran into along the way.

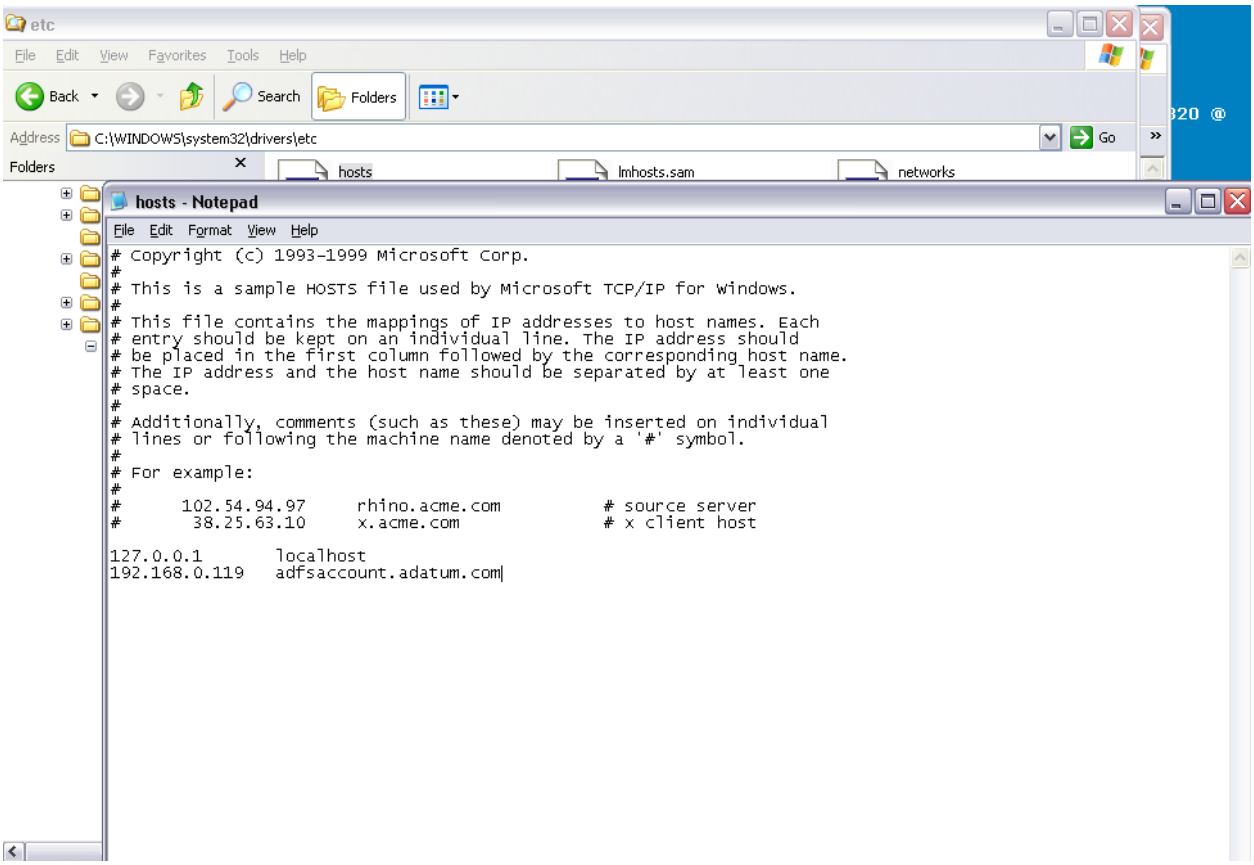
First, we will start with the certificates…**We need an SSL certificate on the default web site that has a subject name which matches the Federation Server URL**. Since I am adding a proxy on the Account side, I need a SSL certificate with adfsaccount.adatum.com





A good checkpoint would be to simply visit https://adfsaccount.adatum.com and make sure you can get to the Under Construction page without any certificate errors. In order to do this, we need to make sure that the name adfsaccount.adatum.com resolves to the IP address of the Proxy machine instead of the FS-A server. My DNS server currently resolves adfsaccount.adatum.com to the IP address of the FS-A. So, the easiest way to do this in a lab environment like this is by using a host file entry.

My Proxy Server has an IP of 192.168.0.119 – so I can use the host file to bypass DNS resolution for this name. It is easy to comment out the entry and put it back so you can simulate an external client and internal client quickly.



Now that we have SSL setup properly and our client machine resolves the name to the IP of the proxy server, we are ready to request and install a Client Authentication Certificate in the local computer store.

In my first attempt and putting this blog together, I ran into some issues with the client auth certificate, so it may be good information for you to read that blog before going any further.

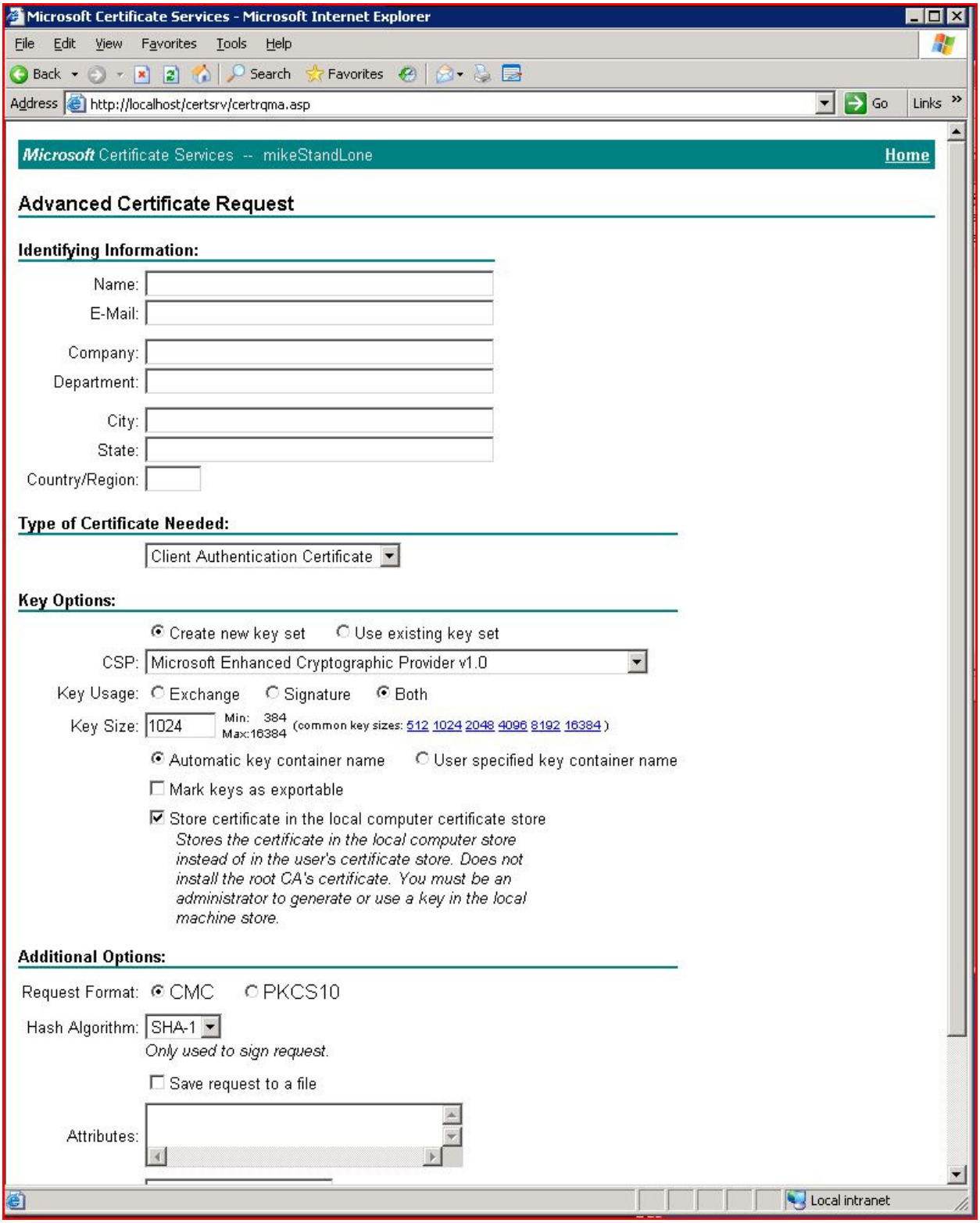
The client authentication certificate will be used by the Proxy server to authenticate with the Federation Server. We will install it into the local computer personal store, then export the public key and add it to the Trust Policy on the Federation Server.

Unlike the SSL certificate, we don't need to worry about any specific name. We only care that the EKU has client authentication.

Below is a shot of my certificate server web page after doing "advanced certificate request" then "Create and submit a request to this CA"

In the name field, I just put something useful to identify the certificate quickly when I view my local computer store.

If you have a plain Standalone CA, your screen should look like this:

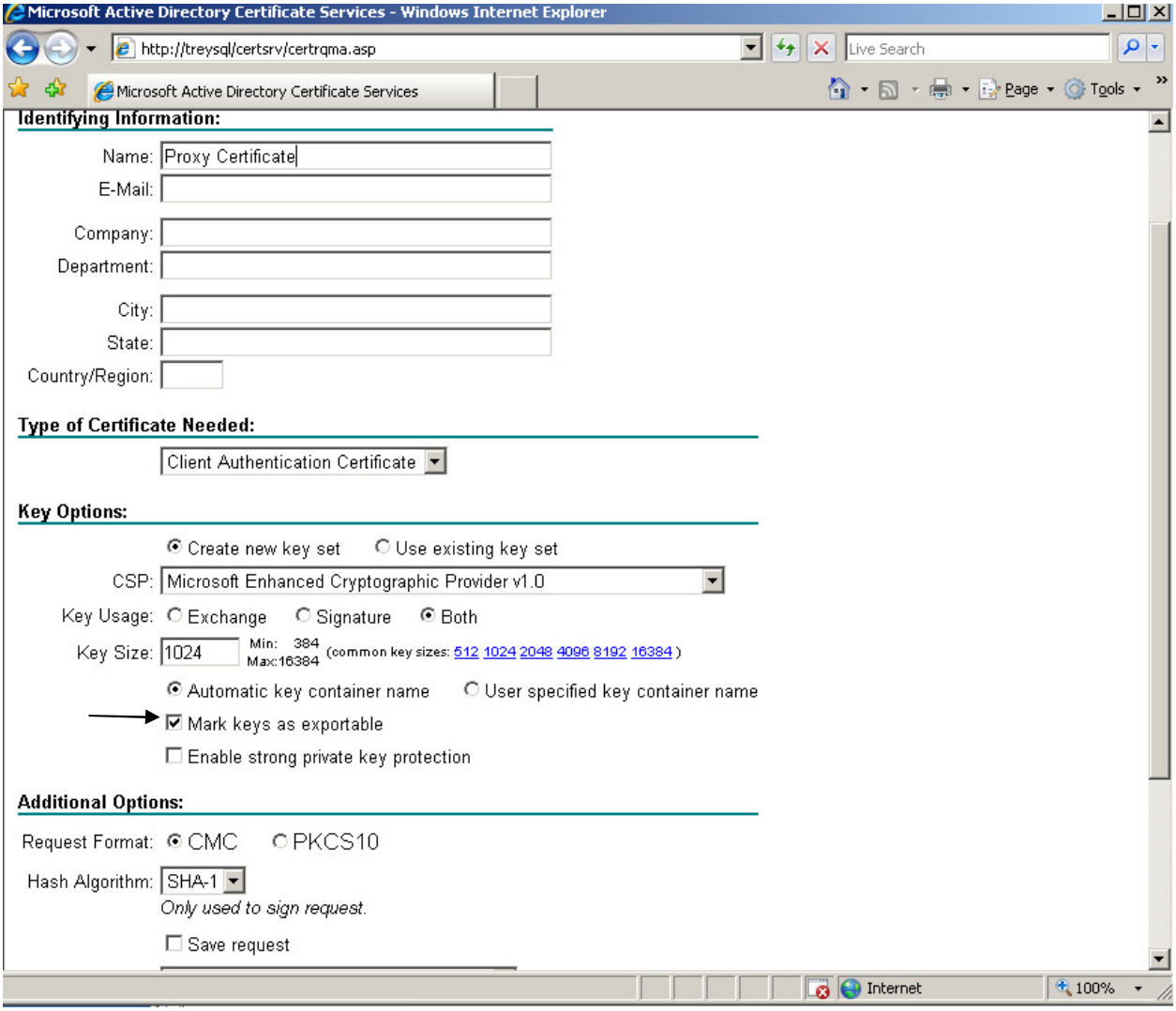


You can check the box to store the certificate in your local store and give it a name like ADFS Proxy Certificate. This will save you some extra steps that I had to go through.

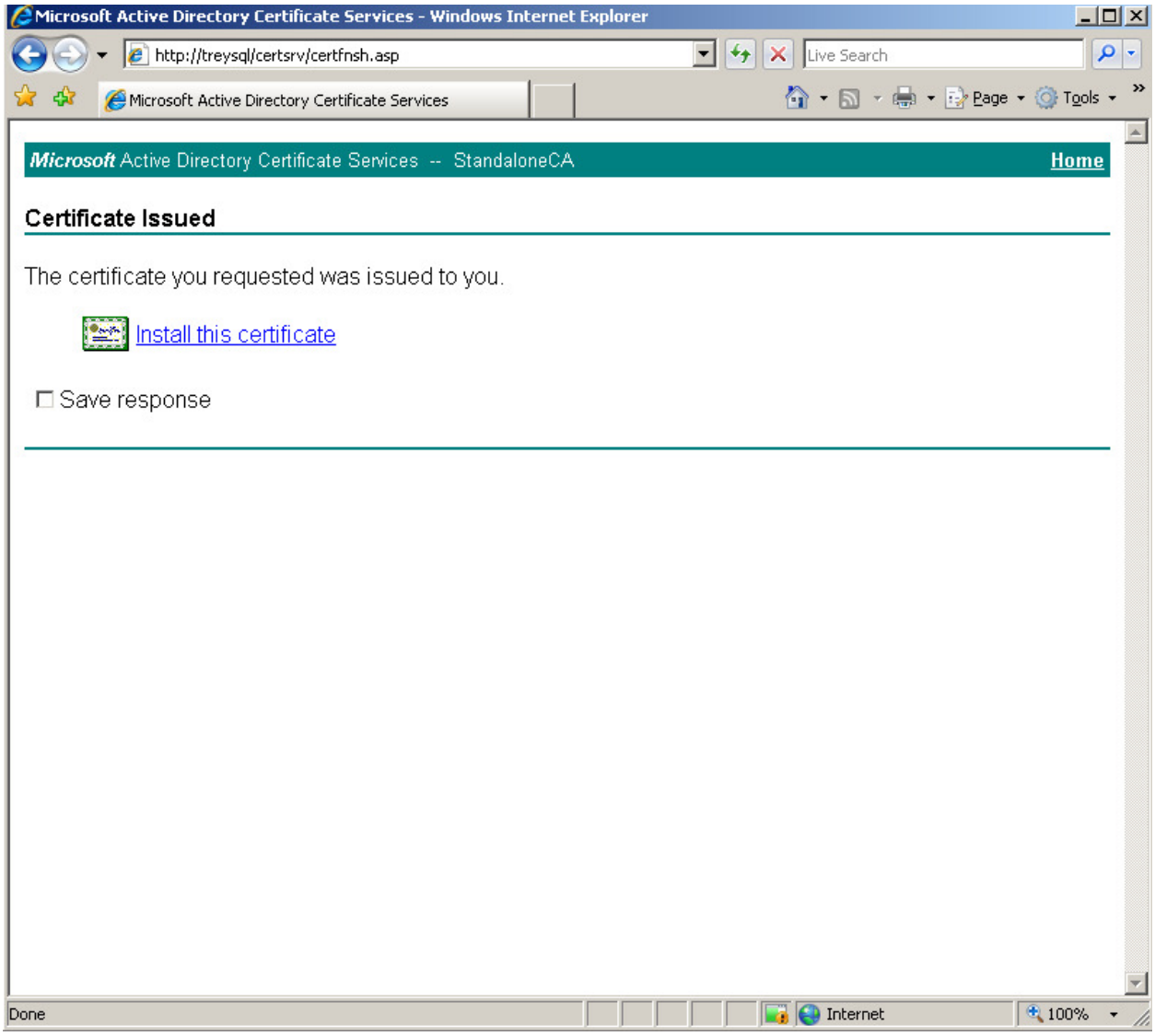
On my CA, I have had to issue some certificates to some Vista and WS08 machines. In order to do this from a 2003 CA, you need to update the web enrollment pages. Instructions and the hotfix needed to do this are outlined in this KB article. In the article it states the following about computer enrollement:

> Computer certificate enrollment
>
> Administrative rights are required to request a computer certificate. In Windows Vista, Microsoft Internet Explorer does not use administrative rights to run. Therefore, the option to store a computer certificate in the computer store was removed from the Windows Server 2008 certificate enrollment pages.

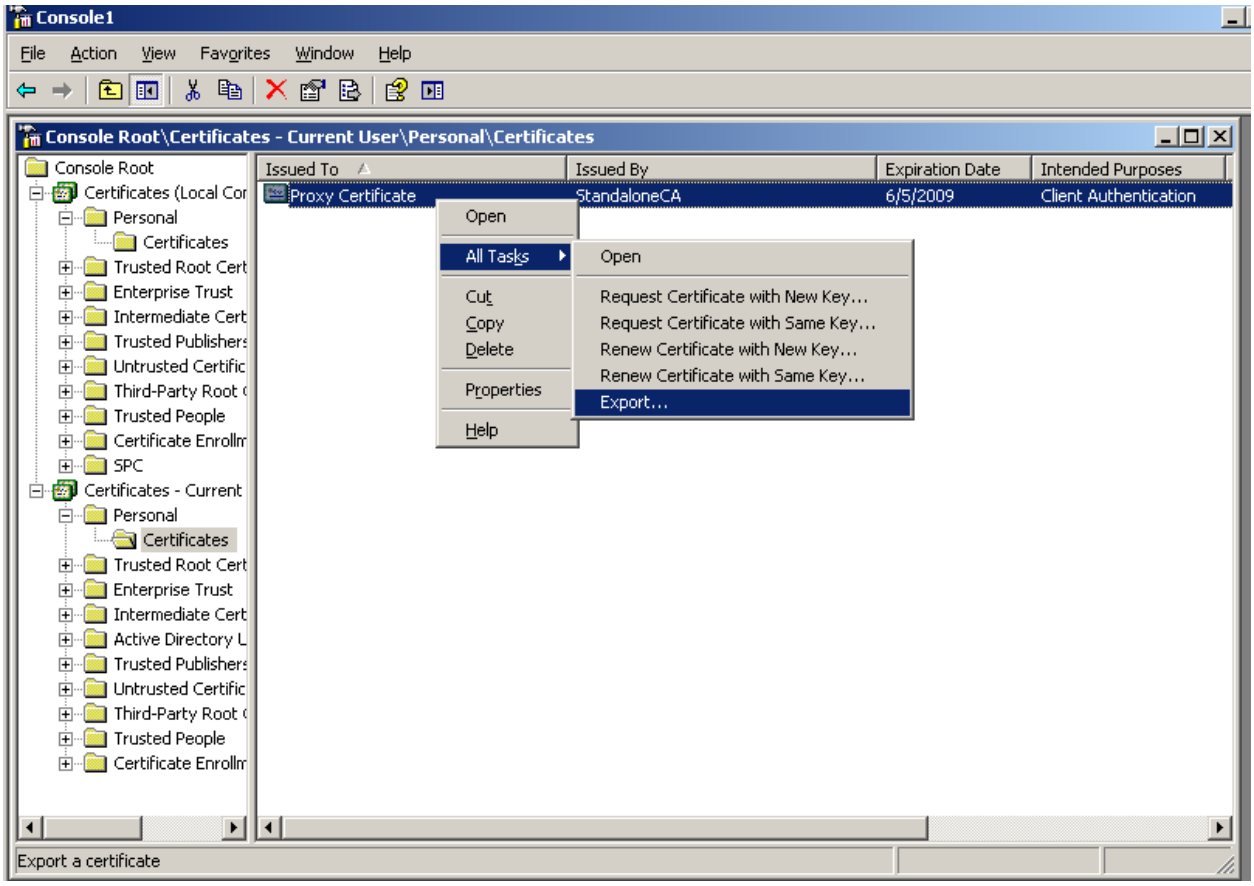
Note the lack of the ability to install the certificate directly into the computer store. So I had to install it in the user store, then export with the private key, then import to the computer store. I had to check the “mark keys as exportable” checkbox before placing the request.



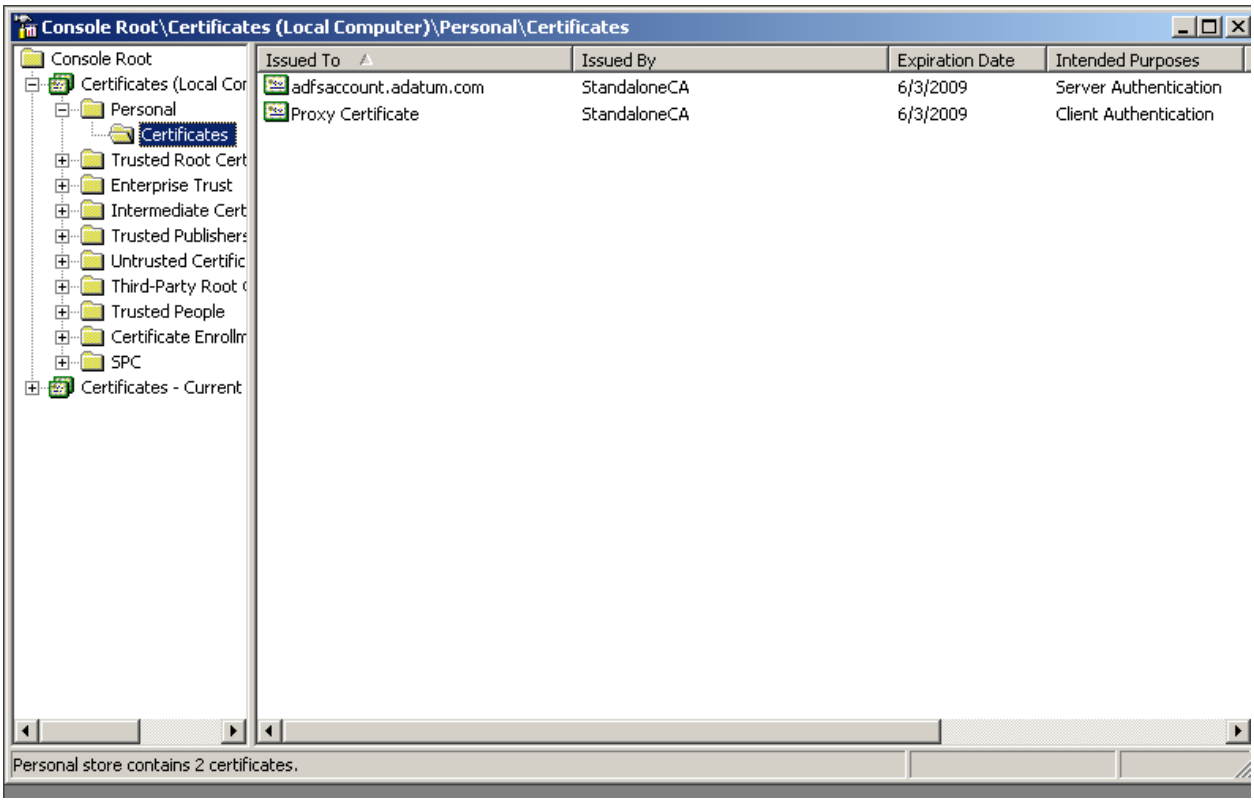
After approving the request on my CA, then going back to check on the status of a pending request, the only option is to “Install this certificate” and when you do this, it is placed in the user store.



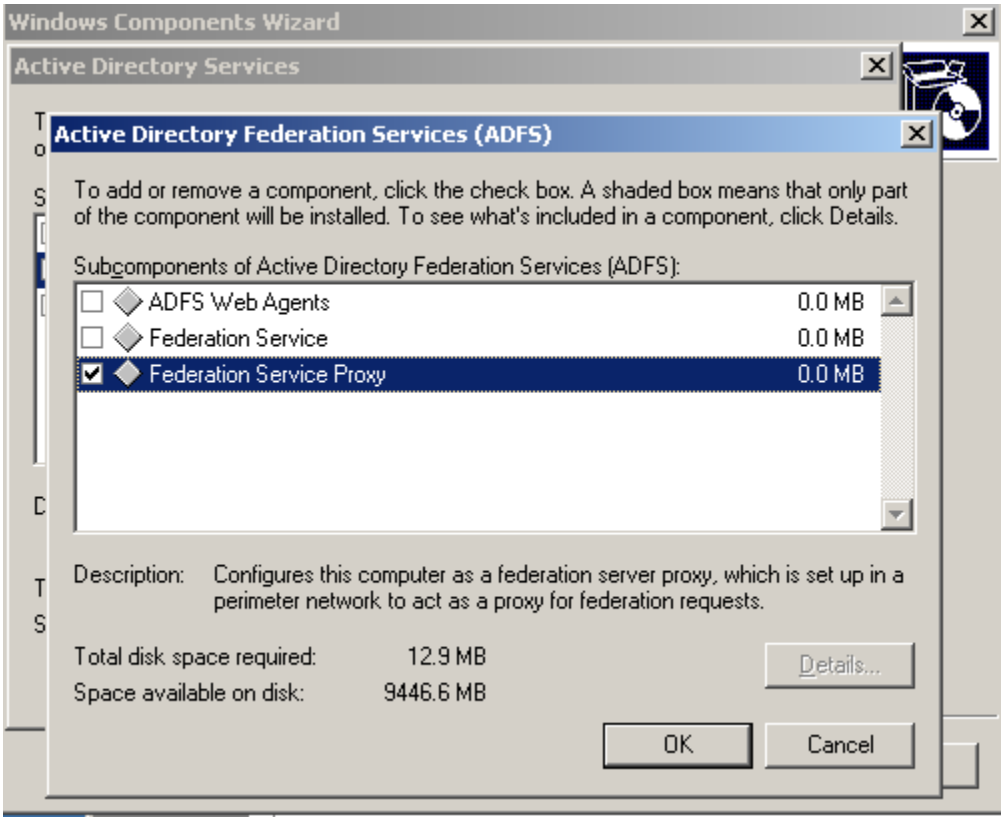
From the user store, do an export with the private key



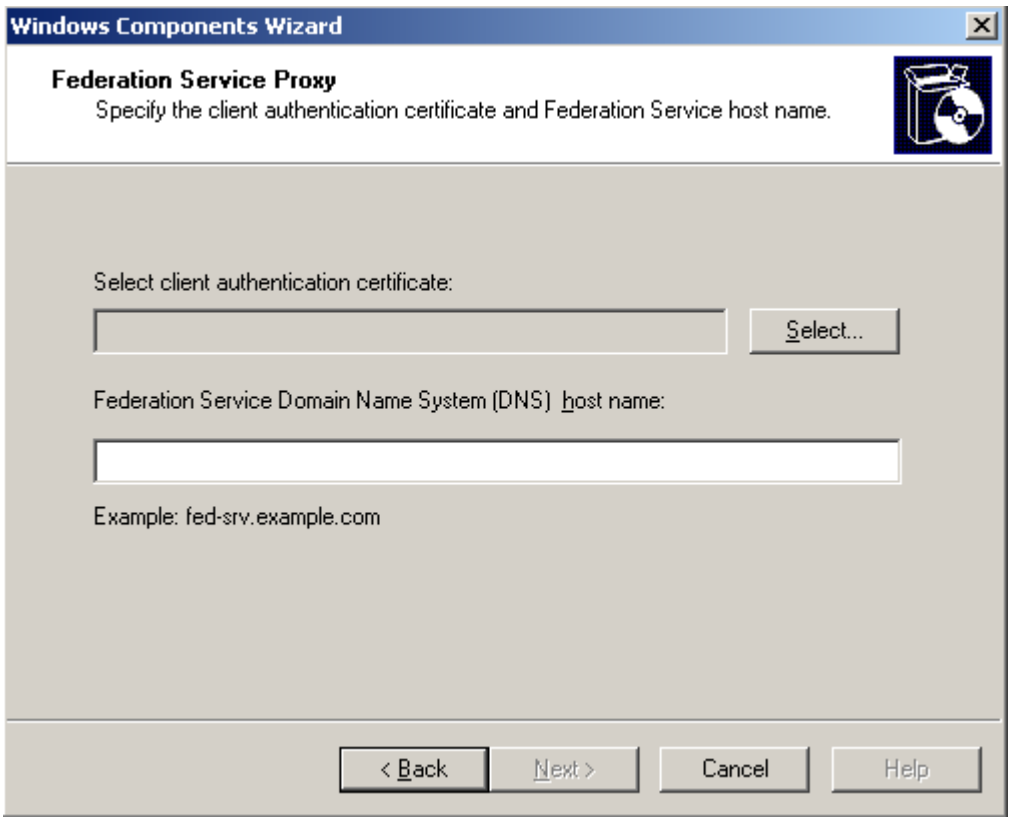
Then import to the local computer store and this will complete the Client Authentication Certificate request and your local computer store should look like this:



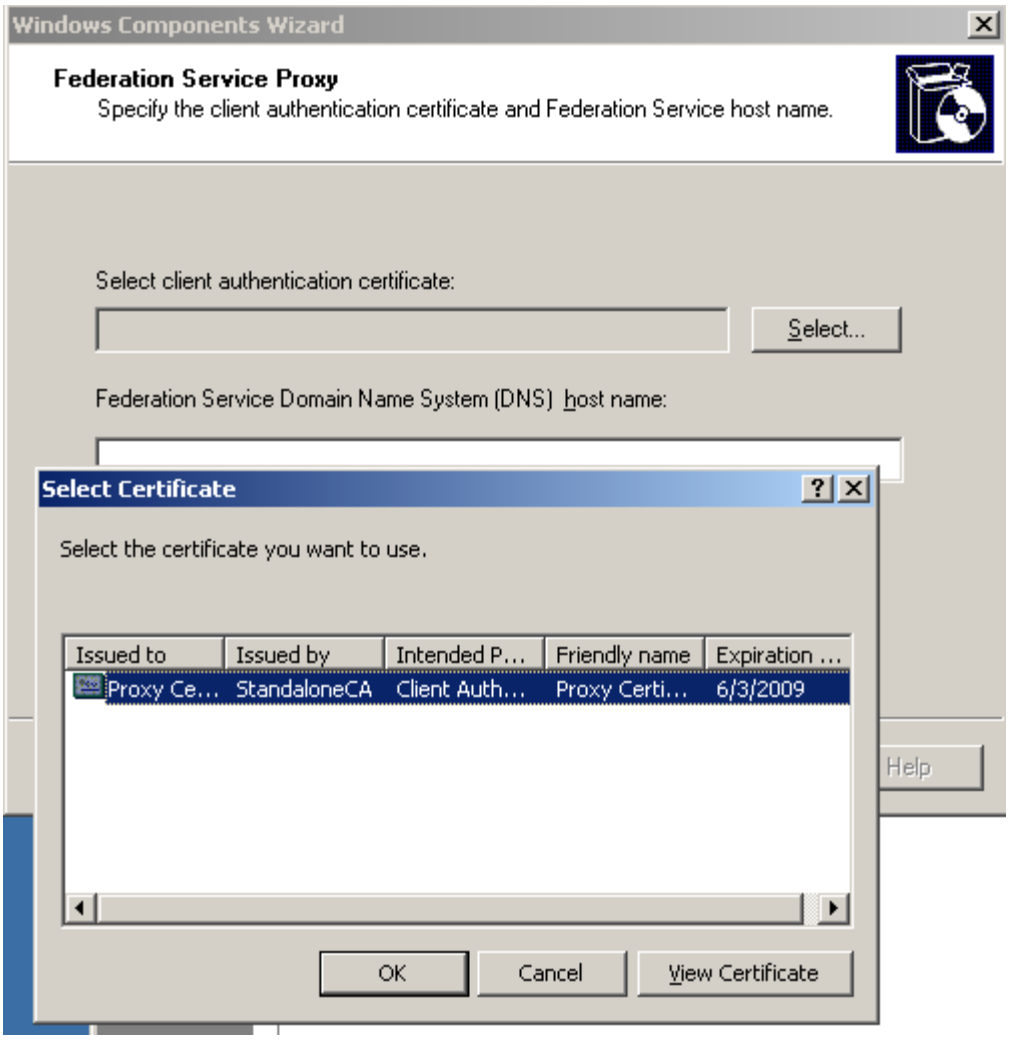
The next step is to install the ADFS Proxy component on this server.



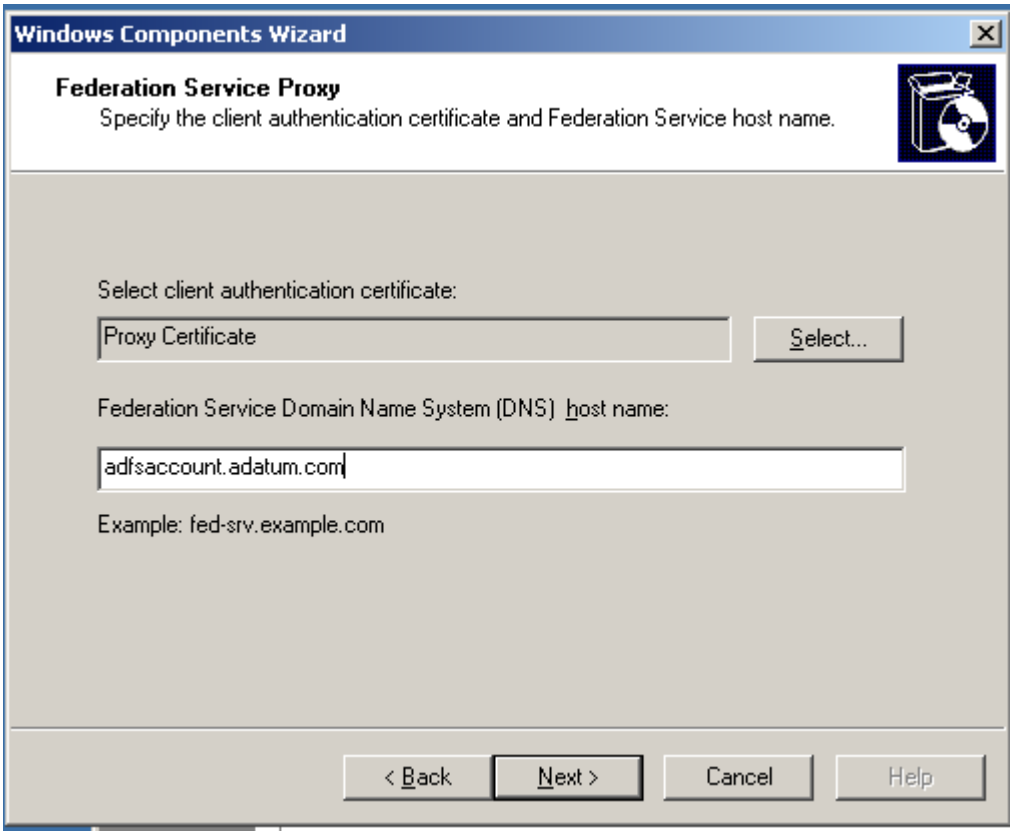
Setup will prompt you to choose a Client Authentication Certificate.



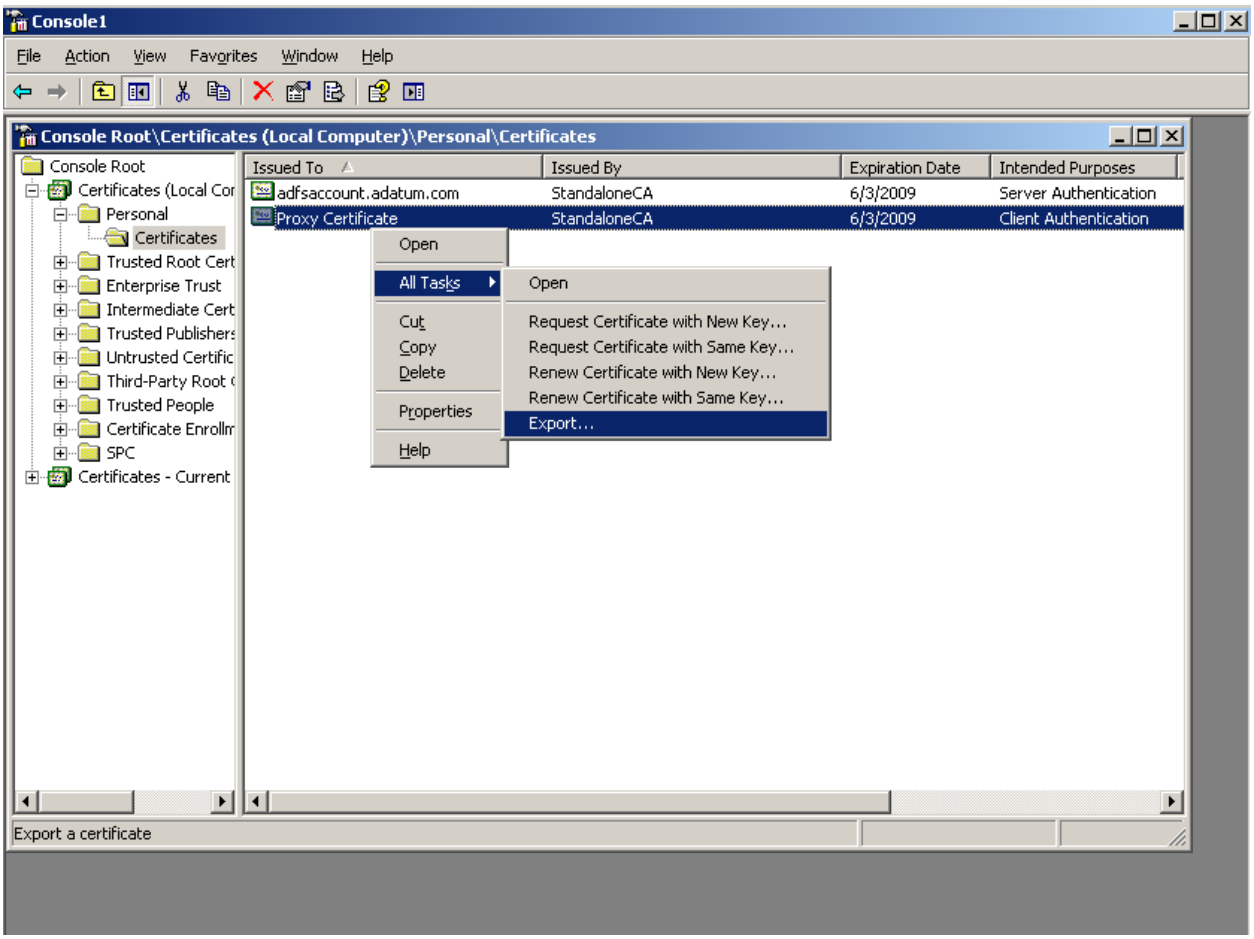
After choosing Select – you will be displayed with a list of all certificates that have the Client EKU in the local computer store. In this setup, I only have one.



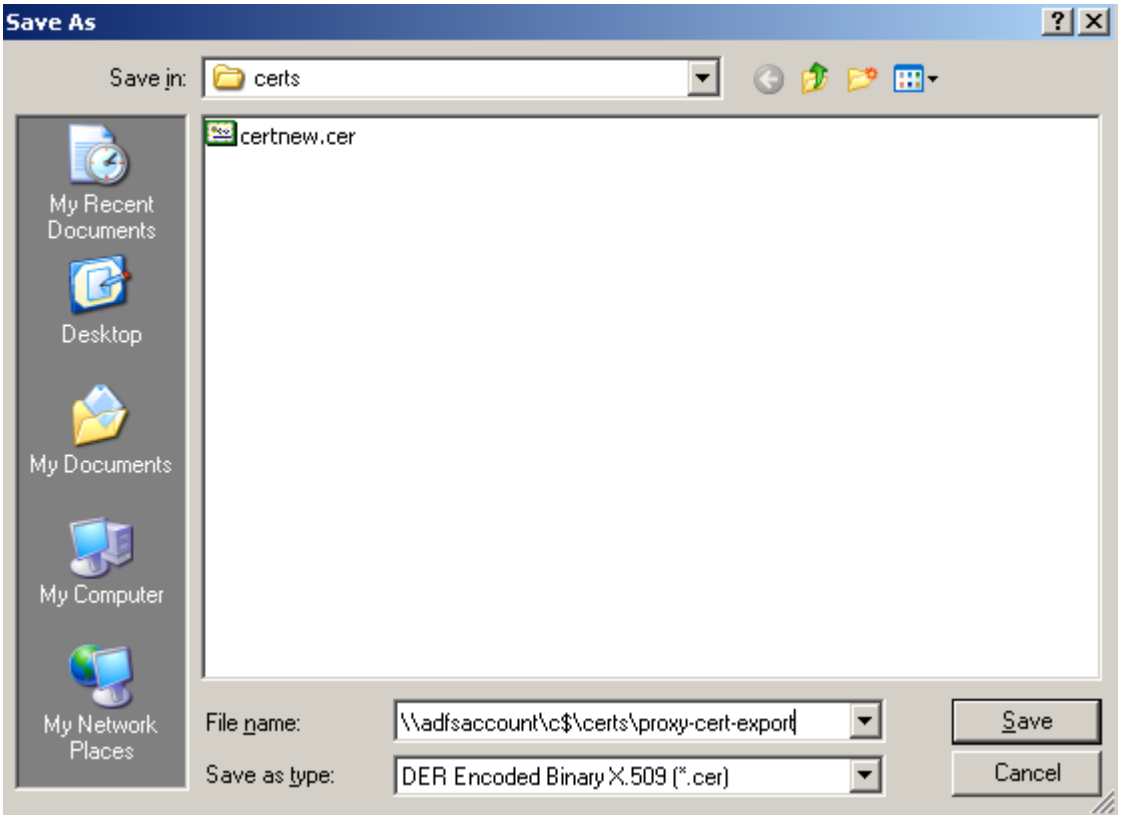
The next piece of information that setup will want is the FQDN of the Federation Server. We also should ensure that the Proxy Server resolves this name to the IP of the actual Federation Server. In most cases, this is accomplished with a host file entry. I will explain the name resolution portion of this more at the end of this blog.



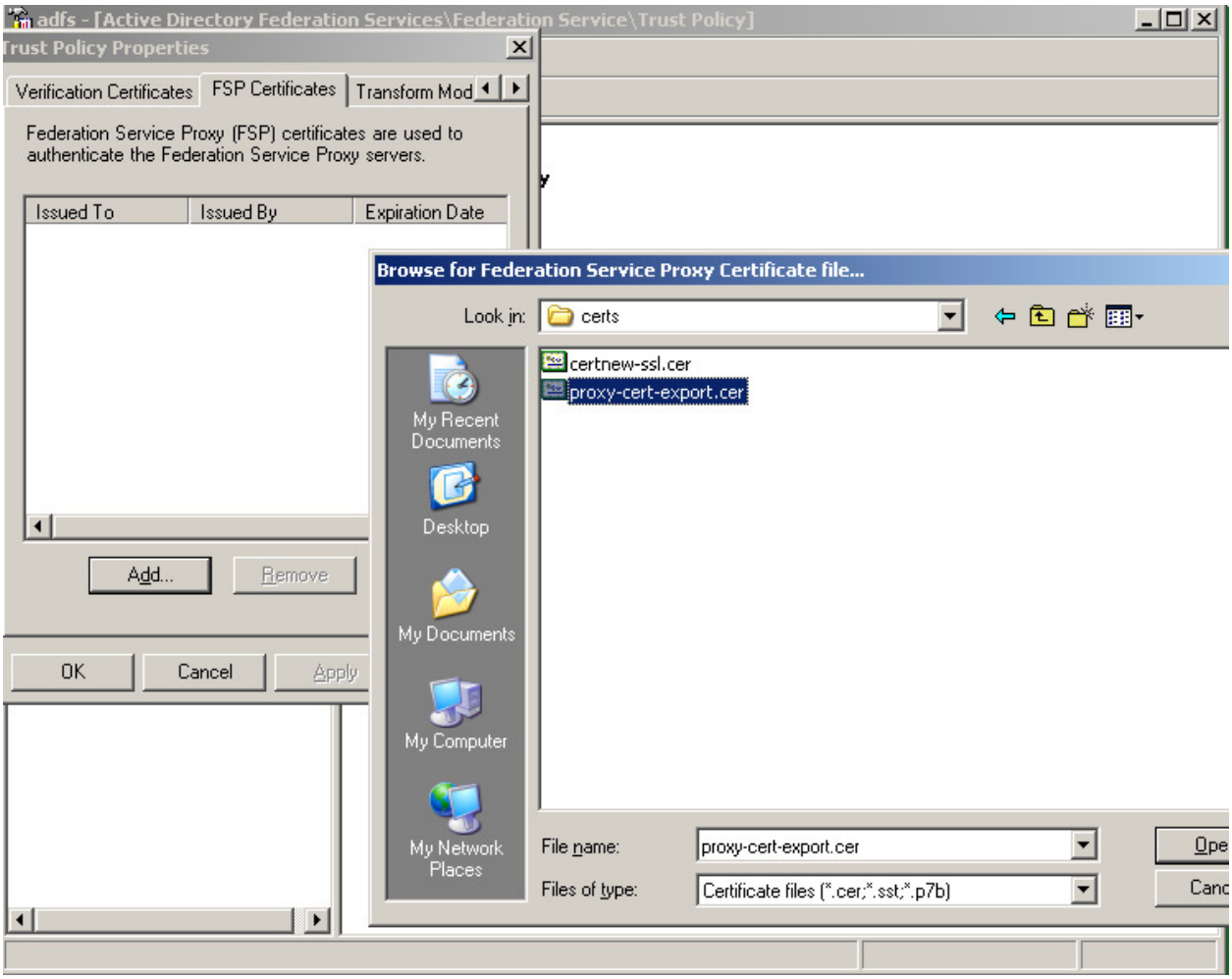
The next item we need to do is export our Client Authentication Certificate (only the public key is needed) and copy it to the Federation Server.

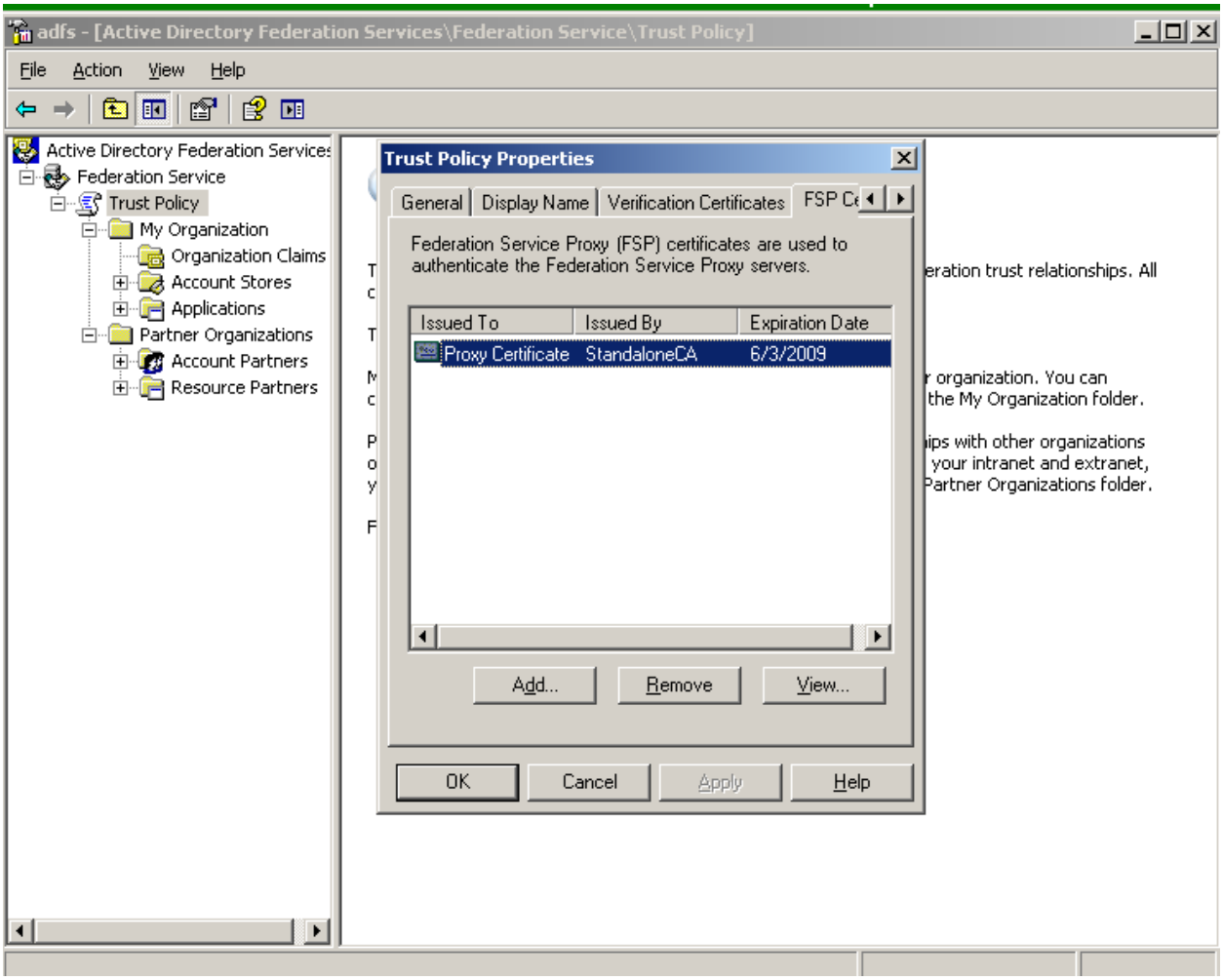

Console1
File Action View Favorites Window Help
Console Root\Certificates (Local Computer)\Personal\Certificates
Console Root
Certificates (Local Cor
Personal
Certificates
Trusted Root Cert
Enterprise Trust
Intermediate Cert
Trusted Publishers
Untrusted Certific
Third-Party Root
Trusted People
Certificate Enrollm
SPC
Certificates - Current
Issued To
Issued By
Expiration Date
Intended Purposes
adfsaccount.adatum.com
StandaloneCA
6/3/2009
Server Authentication
Proxy Certificate
StandaloneCA
6/3/2009
Client Authentication
Open
All Tasks
Cut
Copy
Delete
Properties
Help
Open
Request Certificate with New Key...
Request Certificate with Same Key...
Renew Certificate with New Key...
Renew Certificate with Same Key...
Export...
Export a certificate


Certificate Export Wizard
Export Private Key
You can choose to export the private key with the certificate.
Private keys are password protected. If you want to export the private key with the certificate, you must type a password on a later page.
Do you want to export the private key with the certificate?
Yes, export the private key
No, do not export the private key
Note: The associated private key is marked as not exportable. Only the certificate can be exported.
< Back
Next >
Cancel

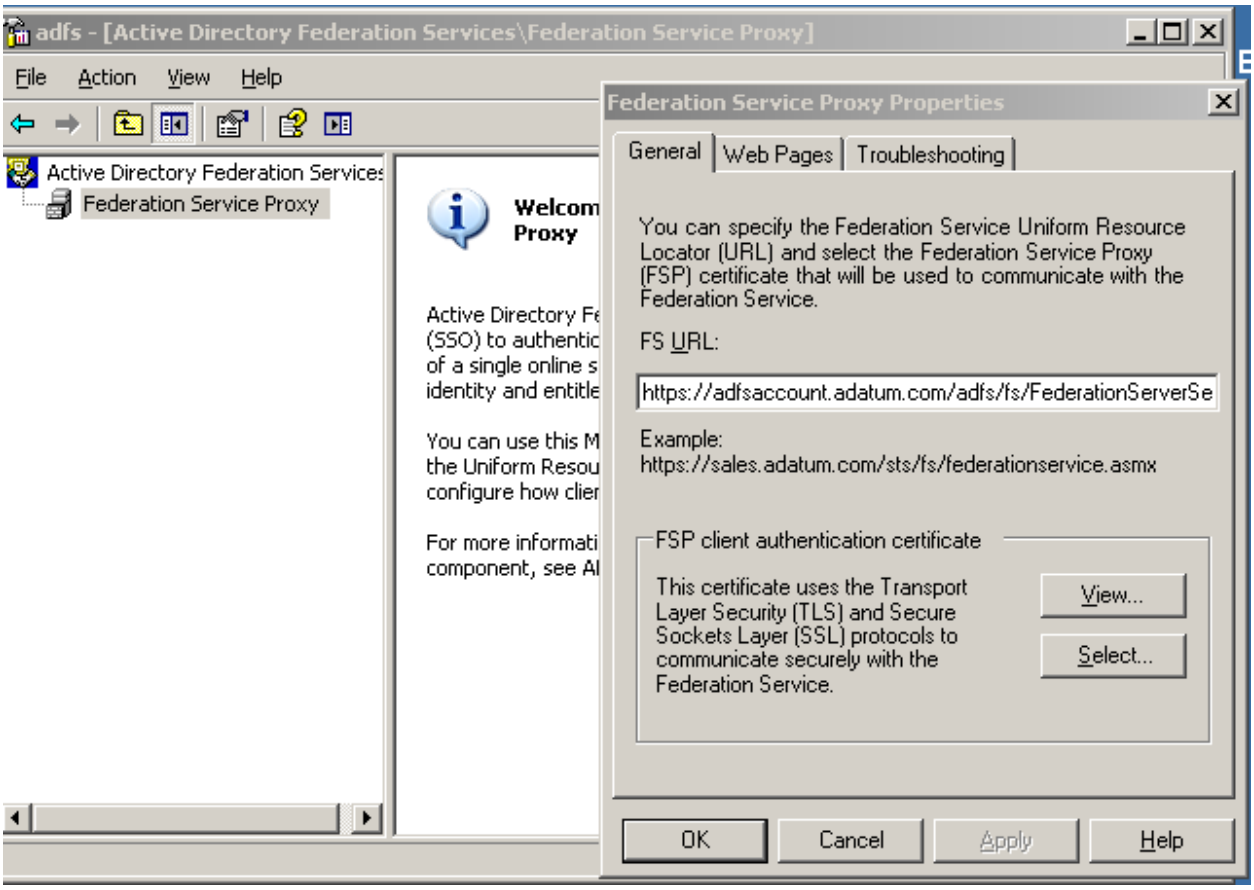

Save As
Save in:
certs
certnew.cer
My Recent Documents
Desktop
My Documents
My Computer
My Network Places
File name:
\\adfsaccount\c$\certs\proxy-cert-export
Save as type:
DER Encoded Binary X.509 (*.cer)
Save
Cancel

Now we need to go to the Federation Server itself and launch ADFS.MSC. From the snap-in, go to properties of the Trust Policy and then go to the FSP Certificates tab. This is where we are going to add the exported client authentication certificate.

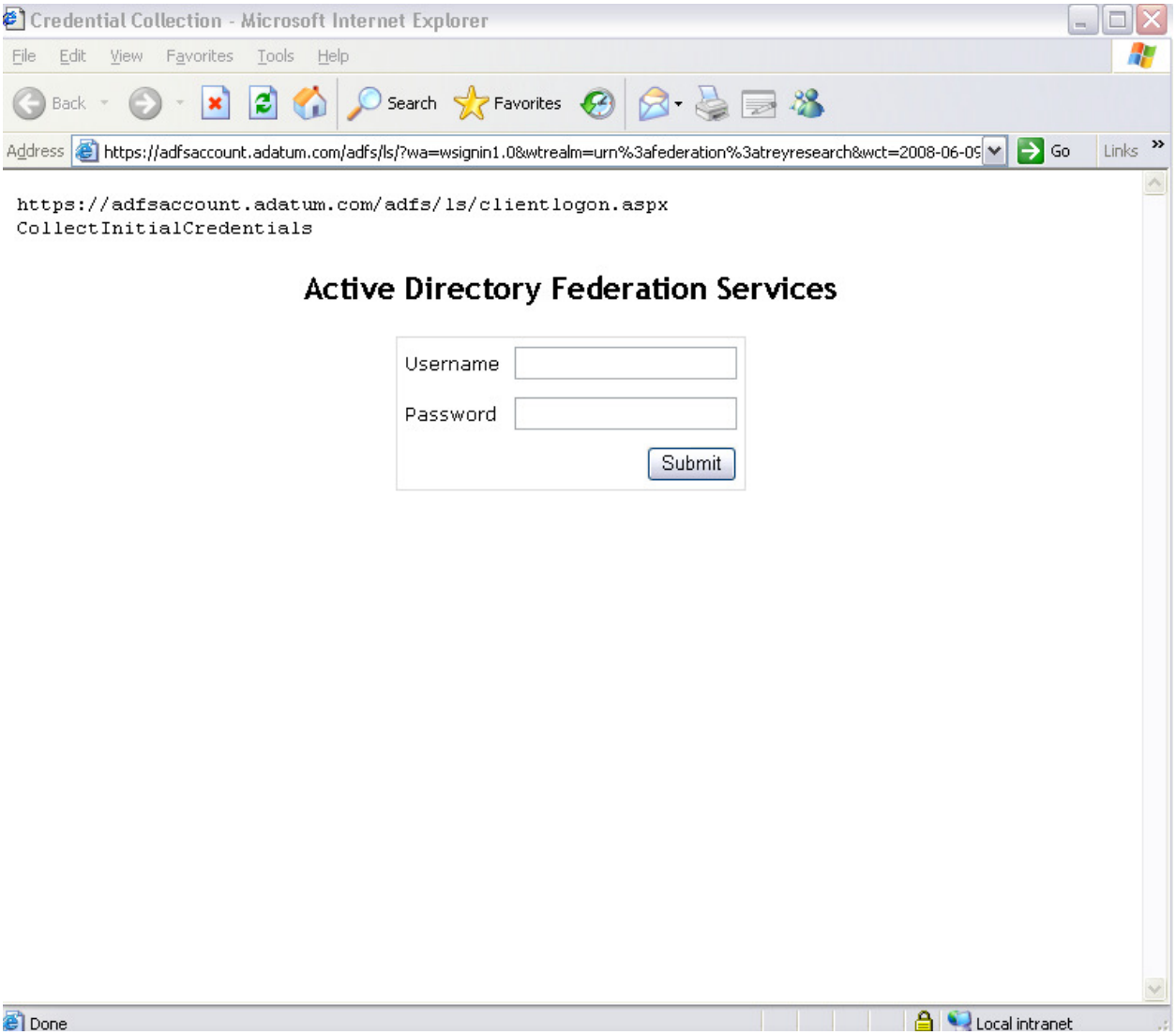




If you go back to the Proxy server and launch ADFS.MSC, you will notice there isn't much to configure here and all the information needed should already be present.



Next, from the client machine that we have a host file entry on, we will enter the web application URL. Instead of being redirected to the FS-A when the client resolves adfsaccount.adatum.com it will go to the FS-A Proxy and we get a Forms Based Auth page like this:



This is the clientlogon.aspx page from the Proxy Server and the user is prompted for Username/Password each time they access an ADFS enabled application.

I'm going to try to cover a few items that often cause confusion with the Proxy component.

1. The server does not have to be domain joined. It can be and often is a standalone server in the perimeter network. A typical setup would be to have the Proxy in the DMZ and a firewall rule which allows communication over 443 between the Proxy and the Federation server only.
2. The matching certificate subject names on the Federation Server and the Federation Server Proxy also cause confusion. The reason for this is that the ADFS server can only have a single endpoint URL. The web servers and partner federation servers can only be configured with a single URL for federation services. In my example it is adfsaccount.adatum.com. My Federation

Server has an IP address of 192.168.0.170 and my Federation Proxy Server has an IP address of 192.168.0.119 (normally this would be a public IP since it would be in the DMZ). My internal DNS server has an A record for adfsaccount.adatum.com → 192.168.0.170, but the internet DNS servers would have an A record for adfsaccount.adatum.com → 192.168.0.119

If we think about this – if the client is internal to the network, it will point to internal DNS for name resolution and will resolve the name to the .170 address and never visit the Proxy Server. This will result in a single sign on experience as the client has already entered username/password to authenticate with a DC on the internal network.

If the client is at home or at a public place on the internet, they will be pointed to some ISP DNS server for name resolution. This will resolve the name to the .119 address and the user will get a Forms Based Authentication experience because we assume they have not authenticated with a DC on the internal network.

Thru the use of a host file on the client machine, we can simulate resolving the name to different IP addresses quickly. The client is pointed to internal DNS so it resolves the name to .170, but a host file entry with the adfsaccount.adatum.com to the .119 address will bypass DNS and simulate a different DNS server with the .119 for that name.

I hope this is clear and I'm not over explaining. Please feel free to comment to this post if it isn't clear or if you have a better way to explain.