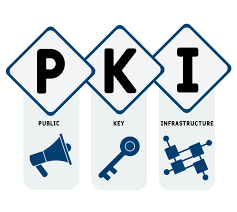
**PKI Infrastructure**



In this project, we will be creating a virtual environment that implements a Public Key Infrastructure (PKI) infrastructure. This will allow for secure login to network endpoints using tokens. The implementation of PKI infrastructure ensures the confidentiality, integrity, and authenticity of information exchanged between different network endpoints.

For the environment we will need windows servers and a workstation to log into. We will need the Domain controller role and the Certificate Authority role, implemented on our windows servers.

**PKI IMPLIMENTATION**

We will first add the Domain Controller and Certificate Authority Roles to our Windows Servers. This can be seen in the screenshot below.

In the screenshot below we can see the Certificate Authority Role implemented on our windows server.

Text

Description automatically generated

In the screenshot below we can see the Active Directory Role implemented.

Graphical user interface, text

Description automatically generated

Once our infrastructure is in place, we went to the PIV website and downloaded the required software, the first being the Admin kit. This can be seen in the screenshots below.

Graphical user interface, text, application, email

Description automatically generated

It will download the following software. This can be seen in the screenshot below.

Graphical user interface, application

Description automatically generated

**Enrollment Process**

We first created an Active Directory User called “Crypto G’ and made sure that it had the correct properties for the Certificate Authority.

The screenshots below demostrate the different properties that have to be manipulated for the User. First we will assign the Crypto G user an email.

Graphical user interface, application

Description automatically generated

Next, we will create a log on and make sure that” Smartcard is required for interactive login” is checked.

Graphical user interface, application

Description automatically generated

Then, we went to the Certificate Authority Role and implemented the “Smart Card User” Certificate template.

Graphical user interface, text, application

Description automatically generated

Following that, we set up the certification for the private key and implemented the Smartcard User template. This configuration can be seen in the 2 screenshots below.

Graphical user interface, application

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

At this point in the process, it is important to note that we set the correct permissions for the security roles.

Graphical user interface

Description automatically generated with low confidence

We also disabled email from the subject name, so we do not have to be verified through email, this was done for simplicity.

Graphical user interface, text, application, email

Description automatically generated

Next, we enabled our new Certificate. This can be seen in the screenshot below.

Graphical user interface, text, application, email

Description automatically generated

At this point in the process, we will be working a lot in the mmc.exe application. A screenshot of this program can be seen below.



Within the mmc console we went to add a Snap in. This can be found in the “File” tree. Seen in the screenshot below.

Graphical user interface, text, application, table, email

Description automatically generated

Below you can see the Snap in being added.

Graphical user interface, text, application

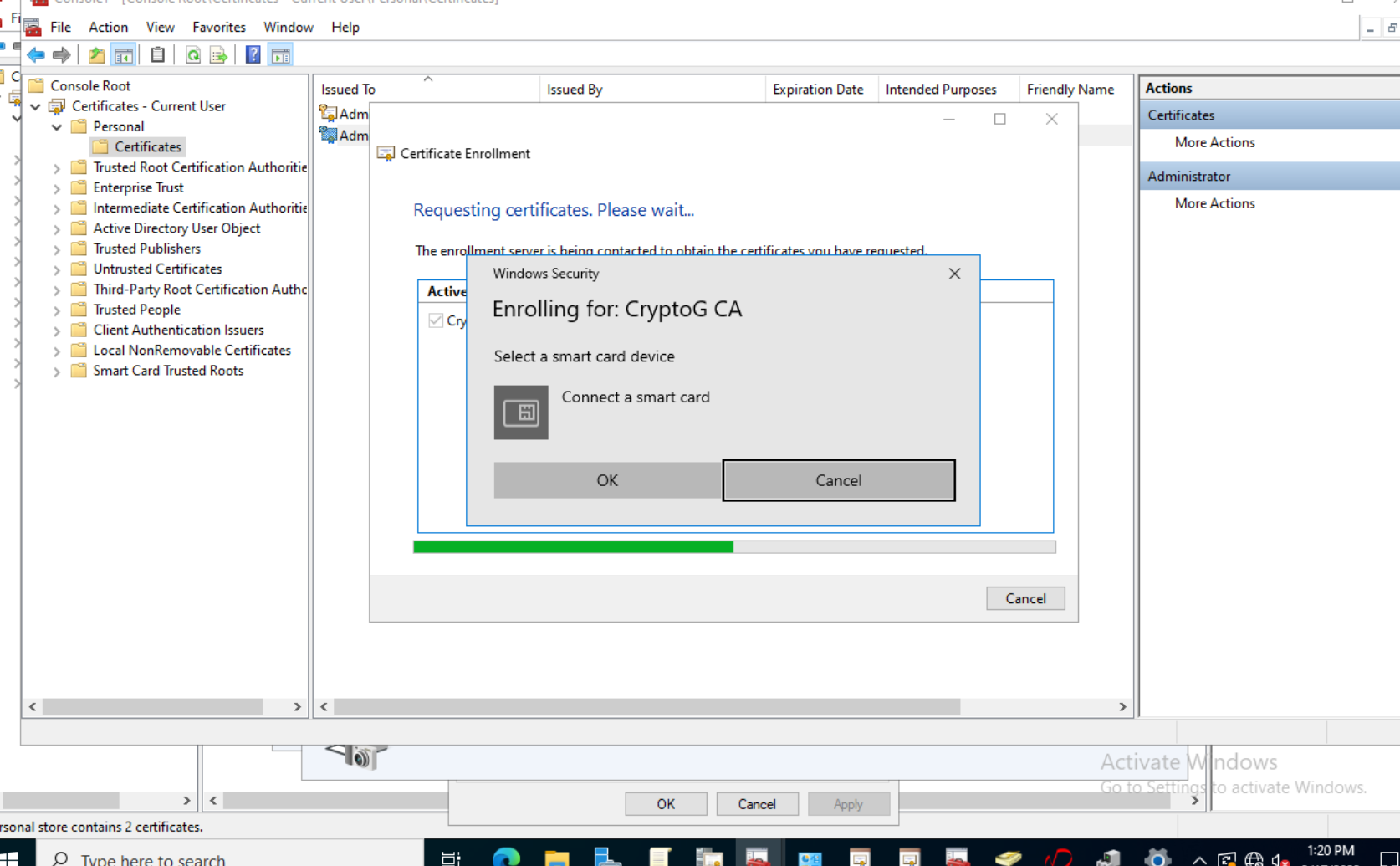
Description automatically generated

In the screenshot below you can see that the “STATUS” for the crypto G CA is “avalable”. We checked it and then enrolled the CA. This can be seen below.

Graphical user interface, application

Description automatically generated

We then are prompted to connect a smartcard when clicking “ok” for the enrollment. However, before we can do this, we will need a pin. The screenshot below shows the enrollment prompt.



You can see here that the smart card reader has been added to the targeted device. It is important to remember to update the drivers of the smart card.

Graphical user interface, application

Description automatically generated

Now we will go back to our PivKey software and reset the pin. This can be seen in the screenshot below.

Graphical user interface

Description automatically generated

Now that our pin is set, we are able to enroll the CryptoG CA with that pin.

Graphical user interface, application, Word

Description automatically generated

If it successfully installs you will see that it succeeded in the Certificate Installation results. This can be seen in our screenshot below.

Graphical user interface, application

Description automatically generated

Referencing back to our PivKey software, we should be able to see the certificate in the app, this particular one is for the administrator.

Graphical user interface

Description automatically generated

Next, we will be creating a Certificate Authority for the users. Much of the same process will be repeated. We need to go into the Windows server and create the template and configure the Crypto G user’s properties. In the screenshot below we are duplicating another template for the CryptoG users.

Graphical user interface, application

Description automatically generated

Then we will be configuring the Active Directory properties. In the screenshots below we are enabling the Base Smart Card Crypto Provider and making sure that we prompt the user during enrollment

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

We then made sure that we were building it from Active Directory, with the email and UPN checked.

Graphical user interface, text, application

Description automatically generated

For the screenshot below, check read, write and enroll for Authenticated User, and domain users and domain admins, (Because they are using the same token)

Graphical user interface, text, application

Description automatically generated

The finally we make sure again that we have “Prompt the user during enrollment” checked.

Graphical user interface, text, application, email

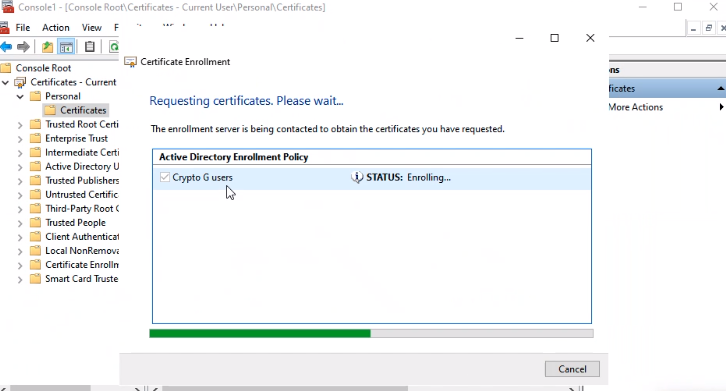
Description automatically generated

Next is enrollment agent. (We are making sure to rename it as to not confuse other students.)

Table

Description automatically generated with medium confidence

The we renamed it to Crypto G enrollment agent, then we go to the MMC and enroll our agent. This can be seen below.



Next, we need to configure the CryptoG Enrollment Agent properties. We will start in the “Request Handling” tab and make sure the “Purpose” is signature and that we prompt the user during enrollment with a required a private key.

Graphical user interface, text, application

Description automatically generated

Then we change the permissions for domain admins and authenticated users. These permission changes can be seen in the screenshot below.

Graphical user interface, application

Description automatically generated

Then we make sure UPN is checked again. See screenshot below.

Graphical user interface, text, application

Description automatically generated

Now we are heading back to the MMC console so we can enroll the user agent.

Graphical user interface, text, application

Description automatically generated

Once the agent us successfully enrolled , we went to the Pivkey application, this is where you will see the signature.

Graphical user interface, text, application

Description automatically generated

Next go back to MMC and we will enroll on behalf of.

Graphical user interface, application

Description automatically generated

This is the certificate we created.

Graphical user interface, application, Word

Description automatically generated

Now we will be using the certificate and enrolling the users, so they can authenticate with the smartcard. This can be seen below.

Graphical user interface, application

Description automatically generated

Once we start that process, we then select our user. This can be seen in the two screenshot below.

Graphical user interface, application, Word

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Then it will be enrolled in the system as CryptoG, this will be our sig. You can see this in the screenshot below.

Graphical user interface, text, application

Description automatically generated

Now we will be able to log into both our Users and our Administrator with the Smart Card functionality. The two screenshots below demonstrate the successful log-ons for both accounts.

Graphical user interface

Description automatically generated

Sign with ADMIN

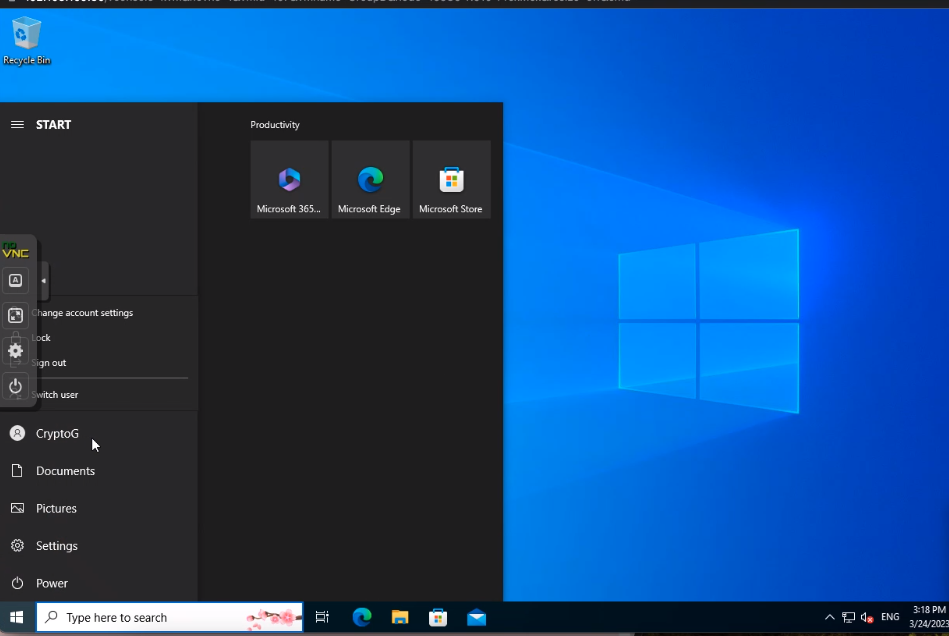
Graphical user interface

Description automatically generated

A screenshot of a computer

Description automatically generated with medium confidence

You know that it is working because you have multiple sign-in options between the user and the Administrator.



**Conclusion**

To conclude, our implementation of a Public Key Infrastructure (PKI) infrastructure using Windows servers and a workstation is crucial for secure communication between network endpoints. By implementing the Domain controller and Certificate Authority roles, we have created a secure PKI infrastructure that ensures confidentiality, integrity, and authenticity of information exchanged over the network. This project showcases the importance of PKI infrastructure in enhancing network security and protecting sensitive data.