Configuring and Using PKI in Your Microsoft Network
Introduction

Public Key Infrastructure (PKI) has become an essential service to implement and maintain proper security on many networks. The PKI service that Microsoft offers meets that objective by providing a set of tools to guarantee the information stays in a secure state. By definition, a secure state is part of the CIA triad: confidentiality (data remains seen by authorized entities), integrity (data cannot be tempered with), and availability (data remains online and can be accessed). As discussed in my previous white paper (Fundamentals of the PKI Infrastructure), PKI reinforces this idea by applying the first two types of security, but also provides some additional protection, as we will soon discover.

In this white paper, we will explore the practical use of the PKI and its configuration. We have seen many organizations using this technology, but some require an appropriate setup and additional configuration to take advantage of the numerous possibilities PKI has to offer. We will be looking at configuring the core components of the PKI role, as well as how it is possible to secure different types of servers using this technology.

This white paper is part of a two-part series on the PKI. The first white paper, Fundamentals of the PKI Infrastructure, can be found within the cybersecurity section of the white papers.

Conceptual Overview of the PKI

PKI is the practical application of a set of security technologies to help data protection.

PKI relies on two types of encryption: symmetric and asymmetric. These types of encryption work together by scrambling the data using an algorithm that makes the information unreadable by unauthorized entities, providing secrecy.

Symmetric encryption encrypts the data itself, while asymmetric encryption makes sure the transfer of session keys generated by symmetric encryption is kept secret. Different keys are generated at that point, and are available to cipher and decipher the information.

These two types of encryption, along with keys, work with certificates—the piece of information that can be used to validate users and computers and perform secure data transfer. Certificates can be obtained from specific servers, such as Certificate Authorities (CA), which are individual servers performing maintenance and operations of the entire PKI infrastructure.

CAs are active in issuing certificates to computers, users, or others CAs in hierarchies. Requests can be processed automatically, or by manual approval, depending on properties of the certificates and authentication methods to Active Directory, which can be actively involved in the process of certificate issue.

When a certificate is issued, it has a validity period, and can be revoked and then found in a certificate revocation list (CRL) that is published within the organization. Although certificates and the PKI seem quite complex to understand, creating a PKI in your company is all about establishing trust between computers within your
organization. It makes sure computers know each other when exchanging information. Thus, PKI prevents man-in-the-middle attacks, session hijacking, data tampering, and integrity attacks, and performs non-repudiation as well as authentication between peers.

Before you begin with the configuration with the PKI, here is a graphical overview from my first PKI white paper, outlining the most common components in that structure:

Planning the Microsoft PKI
The setup of the PKI can be straightforward, but will mainly depend on the plan for the configuration of that role. You have to know answers to the below questions before you start with setup.

Does your organization require a PKI?
Despite its very important security role, not all organizations require PKI. Installing a security service such as PKI will require active maintenance and configuration. There are scenarios when such an infrastructure is not necessary. For example, if your objective is to secure a web server that is mainly going to be accessed by the
outside world (for example a vendor portal), then it is not practical to issue a web server certificate from a private CA. If you’re a small company, getting a certificate from an external CA is just fine. There is a fee for this operation, but don’t forget the CA also costs time and internal resources when it comes to its maintenance and configuration.

**How many CAs do you need?**

To answer this question, you need to know the number of users or computers that are going to request certificates and the different types of certificates potentially given to these entities (web, VPN, secure email, etc.). CAs are very scalable, and often the number is going to be dependent on geographical locations, if you are utilizing a decentralized approach.

**Are your CAs going to be part of a hierarchy?**

A hierarchy can be good to have, but involves a certain level of complexity with root and subordinate certification authorities. If you want to spread the load over multiple CAs, and better protect your root, then the hierarchy is required. However, you are effectively increasing the time needed for the configuration and maintenance of your CAs.

**Are your CAs going to be part of a cross-certification structure?**

If your plan is to create trust between multiple organizations, additional configurations such as the import and export of root certificates from each of the companies involved in the trust relationship will have to be performed.

**Who will be the administrators of the PKI?**

In larger organizations, there are very dedicated security roles. One team will be needed to exclusively manage CAs in these companies, and can be potentially associated with other security duties. It is surely a role for which you need a dedicated resource with the appropriate skills and background in security.

**Is Active Directory integration required?**

Many organizations are configuring Microsoft Active Directory to allow a certain level of automation when issuing certificates. If the objective is to achieve certificates auto enrollment, you must integrate your CAs with Active Directory.

**Does the certificate validity need to be verified from outside the organization?**

If so, you will need to publish the CRL and make it available on the Internet, through reverse proxying, for example. You can instruct your certificate to not be checked for revocation, however, this will decrease the level of security, thus it is not recommended and not very scalable.

**What will be the characteristics of certificates?**

Items include lifetime, database locations, and crypto providers as well as key sizes. This can greatly affect the number of certificates a CA can deliver, as well as latency, depending on the length of keys as well as algorithms used to encrypt and decrypt data.

**How does the backup work for CAs?**

Performing backups becomes a critical piece of administration you need to follow, given how important it is to have a strategy to avoid problems in case of CA failures. Backups should be part of the maintenance strategy of every CA, as these will allow you to recover from a potential failure without losing information about certificates and keys issued to PKI clients.
Core Configuration of the PKI

Installation
Once planning PKI objectives is complete, you can proceed with the installation of your first certificate authority. You will see the CA as a role in Windows Server operating systems:

The installation configuration requires the following information:

- Role services: These are subcomponents that complete the PKI role. Standard role services that are installed are the CA itself, CA Web Enrollment (IIS server configured with one website where users can access certificates and make requests), as well as the web service (for non-domain-joined computers).

Setup Type of CA
It is critical to define whether your CA will be connected to Active Directory. In doing so, more certificate request options will be available. For example, auto-approval works only when the CA is connected to Active Directory. When a computer or user requests a certificate, the CA needs to validate its identity. If you select the Enterprise option, validation can be processed through AD, without manually having to specify the identity properties. As a common scenario, subordinates are Enterprise to allow for auto enrollment features, while root CAs are typically Standalone, since no client is going to directly perform certificate requests at the root level. Selecting Standalone does not mean the computer is not a member of the domain, but simply that CA and AD will not interact with that server.
Type of CA
The first CA server will be root; all subsequent servers will be subordinates to achieve a hierarchy. The advantage of creating a hierarchy is the ability to divide CAs by functions, entity types or geographical locations. In case of a root setup within the hierarchy, that server should not be holding any other role used within the organization, as it typically becomes connected once new subordinates are created or when these get renewed, periodically.

- Private Key creation: This key will allow the CA to issue certificates it is able to sign for validity verification.
- Cryptographic options: Here, you specify the options related to the master key, such as its integrity algorithm as well as its length.
- Name of CA: This name does not have to be the computer name; the CA Name is going to be used in all instances where lookups are performed to find CA servers.
- Validity period: Specifies the validity period of the master key, generated by the CA itself.
- Certificate location: Specifies data store location for the entire CA configuration and certificate metadata.

Initial Configuration
Once the installation is completed, a new snap-in within the MMC will be available, from which most of the configuration can be performed:

Permissions
One of the first operations to perform after installation is the configuration of proper administrative delegation for the CA. Typically, a specific team will administer the server, and thus you need to provide them access. However, since multiple access levels are available, other users with limited privileges can be part of the CA administrative groups. The configuration can be found in the CA properties—Security Tab. The following permissions are available:

- Read: Read-only configuration of the CA console
- Issue and manage certificates: Certificate handling with no permission to administer the core CA
- Manage CA: Full control over the CA environment
- Request certificates: Entities allowed to connect to certificate request services

A good recommendation is to have the fewest possible administrators or delegated users having access to the CA functionality.

CRL Publishing
It is critical the CRL is available to all users. A revoked certificate can be found in the CRL, letting users know there is a problem with the information, and preventing them from providing sensitive information.
By default, the CRL is published in Active Directory through LDAP, however, it might be required to publish the CRL through web services and make it available from virtually anywhere. This configuration is specifically required when accessing CAs through VPN tunnels, because if the CRL is not accessible, some CA functionality will not work.

The location specifies URL or the domain name where the CRL will be found and accessed from, as well as where the CRL will be physically published, such as a file server share. Variables involve properties included within the CRL to allow incremental upgrades to the CRL, as well as other characteristics found within the download location of the CRL.

**Key Archival**

Key archival is a feature defined at the global CA level that can be implemented to allow key recovery agents to restore certificates in case of users or computers losing access to their keys. This feature is good to have, as your certificates key can be found in an archive and recovered by specific entities: Key Recovery Agents (KRAs).

The feature is enabled within the properties of the CA – Recovery Agents tab:

Upon enabling the functionality, it is necessary to provide the system with KRA certificates of authorized individuals.
Administration and Maintenance of the CA
Managing the CA is an ongoing process. Once the initial parameters are defined, you will have to continuously make changes within the CA for new request types.

Certificate Templates
One of the most critical components to configure is the certificate templates. Templates define the types of certificates that can be issued, as well as their properties, such as security, name, versions, purposes, etc. It is the certificate template that defines most of the properties of an X.509 certificate once it is issued to a computer or user.

A certificate template defines what the user or computer will be able to perform with the certificate on the network. For example: authenticate to resources, provide an SSL tunnel, configure secure email transfers, and so on.

Note certificate templates are available only in Enterprise types of CAs, as they work closely with Active Directory to deliver auto enrollment and certificate customization features:

If that node is not present within the interface, it is a good indication that you are dealing with a standalone setup.

It is uncommon to use the default list of published certificate templates. Rather, you want to customize some of them and insert new features. For that, you can see all templates available by right-clicking on the above figure items and by selecting Manage:

At that point, it becomes possible to duplicate a template, and adjust it for your needs.
For example, you want to configure a template allowing some users to become KRA (being able to recover certificates from the archive), and at the same time to be able to use that certificate for encrypting files on disk. In that case, you can choose to duplicate the existing KRA (by right-clicking the existing template), and make it fully editable (many features will be grayed out if you choose to not duplicate):

Adding a purpose to a template and indirectly to certificate upon issuance is done through Application Policies, from the Extensions tab. In this example, you need to add EFS, as it is not included in the default KRA template we have duplicated.

In this case, security settings will also have to be altered, as there will be a need to permit users to request certificates based on that template, and potentially auto-enroll the affected users:
Once the new template is created, you will have to make it available to users and computers for requesting certificates based on this template. Template availability can be accomplished by adding it to the list of published templates on the CA:

And, finally selecting your recently created template:

Note that this is only one example of multiple custom templates you can create. For instance, third-party vendors often integrate specific templates to use with their certificates.

For another example, you might be tasked to configure IKEv2 (Microsoft’s new “VPNReconnect” VPN type). This service requires a custom template to be built. Ultimately, the CA does not have the template for that service, so a similar process would be necessary to create one that matches the requirements of the encryption server. As part of that process, you can use custom Application Policies and make use of their object identifiers (OIDs) to further customize the creation of your template:

The options and potential here are enormous and lots of possibilities exist, but don’t worry, as technical documentation exists for most custom types of templates you have to build for a specific server.
Revoking and Controlling Issued Certificates

By using the CA MMC, it is possible to manage revoked certificates, as well as the ones in use:

This is a convenient place allowing you to see all characteristics of certificates this CA has been dealing with. Note a certificate cannot be restored if it is revoked for some reason (key corruption, lost certificate, terminated employee), except when the reason is “Certificate hold.”

CA Backups

It is possible to back up the CA database. It is a fairly simple process and should be done regularly. The option is found by right-clicking on the CA itself and selecting All Tasks – Back up CA.

Requesting Certificates

Computers and users can utilize certificates for many purposes, as you saw above when working with certificate templates. It is basically used often with a super username/password combination, meaning much more security than the traditional login.

The next question is: How do you get a computer or user to request a certificate?

There are three ways to perform this action, assuming you are using Enterprise CAs.

Active Directory Integration and Auto Enrollment

If you decide to use this configuration, you are fully automating the deployment of certificates based on templates for your entities. What happens here is the following:

1. You need to publish a certificate template with the auto enrollment enabled in its security (seen previously).
2. Configuration of group policies is required to pull information from the template, and make it possible to generate a certificate automatically by the corresponding CA:

Notice the Update certificate that use certificate templates option is checked.

You can perform auto enrollment at two levels: computer or user. Depending on the level you choose, certificates are going to be deployed at different locations in the certificate stores. For example, you can configure IPsec authentication between computers. The certificate based on that template will be stored in the computer certificate store. However, if you choose to generate a certificate based on a KRA and EFS usage (configured previously through templates), then it is going to be stored in the user certificate store.

3. Group policy settings need to be refreshed on the computer or user level. If everything is set up properly, entities will get their certificate deployed to them automatically.

This is the most common scenario for client authentication. When a user connects on the network for the first time, a certificate is pushed to that user’s store. Later, when the user accesses resources, such as a VPN, that generated certificate from the store will be used transparently, and will allow the user to connect remotely.

MMC Console
It is also possible to request a certificate through the MMC Certificates snap-in (manual process). You can request certificates based on user or computer level, depending on which store you browse (more on this in the next topic):
All certificate templates that can be used to generate a certificate will be displayed, assuming the user has the permission to enroll for a specific template type:

In the above example, the user is allowed to generate a certificate from two templates, Basic EFS and User. The first provides EFS capability, while the second allows user authentication, EFS, as well as secure email (SMIME). Application issuance information can be obtained by browsing to the details of the template.

This method of acquiring certificates can be interesting for some advanced users, if auto enrollment is not used. This requires manual intervention and some setup on the end-user side, which is not always the most convenient way to perform an operation.

**Web Service**

It is possible to request a certificate through a web-based enrollment tool, typically installed on CAs. This becomes the only way to request a certificate if using a Standalone CA configuration. The certificate request page can be reached by specifying a URL in the below format:

http://CA-COMPUTERNAME/certsrv
https://CA-COMPUTERNAME/certsrv

The exact URL depends on whether the website has already been secured, by using a temporary certificate by the name of “self-signed.” Note that this is common practice, as it is not possible to request a certificate if HTTPS is not used.

Once the service is accessed, there will be two ways to request a certificate:

- Simple: Outlines the use of a client certificate (EFS, client authentication, SMIME)
- Advanced: Outlines the use of any other certificate template the user has access to

Select the certificate type:
- User Certificate

Or, submit an advanced certificate request.
By submitting an advanced certificate request, you can choose to create a new request, or use one that you already have generated. The most common scenario in generating requests in advance is with web servers. You can pre-create a BASE64 encoded file, and copy its content directly:

<table>
<thead>
<tr>
<th>Microsoft Active Directory Certificate Services -- Adatum LON-DC1-CA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Advanced Certificate Request</td>
</tr>
<tr>
<td>The policy of the CA determines the types of certificates you can request. Click one of the following options to:</td>
</tr>
<tr>
<td>Create and submit a request to this CA.</td>
</tr>
<tr>
<td>Submit a certificate request by using a base-64-encoded CMC or PKCS #10 file, or submit a renewal request by using a base-64-encoded PKCS #7 file.</td>
</tr>
</tbody>
</table>

For either option you choose, your request will be approved:

Why is it approved automatically? Because you are using integrated authentication for accessing the website, so you are already logged in as a specific user. When using Enterprise CAs, it is possible to use ACLs with Active Directory to approve your requests, based on your identity.

**Storage of User and Computer Certificates**

Certificates at the computer or user level need to be stored at some location. It is important to know where they can be found, which will allow us to browse properties, but also understand how individuals can back up their own certificates from those locations.

**User Certificates**

User certificates can be stored either on a smart card or directly in the user profile.

What becomes a potential problem is that the loss of a user profile will trigger deletion of any certificates found in the store and the default location.

Thus, it would be better to use a smart card and store keys on a separate piece of hardware, if possible. By browsing to the certificate store at the user level (MMC and certificates snap-in), we can see the issued certificates for that profile:

![Certificate store](image)

Notice the certificates requested by a user will be placed in the Personal/Certificates container. You can browse through the properties of all generated certificates here.
Once these certificates appear in the store, their use will become effective in applications and services that need that certificate. For example, when using EFS, the application will trigger the use of the appropriate certificate from the store. You can also be asked to select which certificate you would like to use for that specific task:

One good recommendation is to store a copy of the certificate in Active Directory. It will help in case certificates are lost from the store (or a user profile is accidentally cleared). It will also prevent creating new certificates in case the user is roaming and using multiple computers. Once the option is turned on, AD will have a copy of the user certificate and will load it upon opening the session. This feature is an enhancement that came with the Windows Server 2008 operating system. To enable this feature, check the “Publish certificate in Active Directory” option found in the general tab of your certificate template.

Computer Certificates

Computer certificates function the same way the user ones do, however these are much more static. In fact, they are attached to one single computer, on which they are stored. The store hosting the computer certificates is very similar to the other variant; however, it cannot be accessed by non-administrators of the system. It has the same structure, the only difference is the computer certificates are not used when performing connection through applications, but rather through activities performed between computers that does not involve user interaction. For example, two computers can authenticate transparently through IPsec. The computer store has one interesting container:
Trusted Root Certification Authorities
Here, you will find which CAs your computer trusts. Since PKI is all about trust between computers, users, and servers, a good recommendation would be to publish the root certificate of your internal CA here, which can be found on the main web service page:

![Microsoft Active Directory Certificate Services](image)

To trust certificates issued from this certification authority, install this CA certificate chain.
To download a CA certificate, certificate chain, or CRL, select the certificate and encoding method.

CA certificate:
- [Current (Adatum-LON-DC1-CA)]

Encoding method:
- DER
- Base 64

Download CA certificate

Although the same container exists at the user level, it is typically configured at the computer store level:

Certificate Backups
User and computer certificates are the same when it comes to performing backups. In fact, it is possible to export certificates with both keys, or just with the public key. If you are exporting both keys, this assumes a real backup, as you will be able to restore the certificate if it gets corrupted or deleted, and needs to be re-imported. If you back up the public key only, it assumes you are transmitting that key to one of your peers to proceed with trust relationships or EFS/SMIME configuration, for example:
Common Certificate Scenario Requests
Different real-world scenarios involve requesting certificates for different practical purposes. Below, you will essentially learn which certificate templates are the most used.

Web Servers
Securing web servers is very common (using the web server certificate template). What you are doing exactly is activating SSL through the HTTPS protocol that is enabled when a proper certificate is configured. For web servers, the situation is more specific, as there is some extra configuration to perform when requesting a certificate. More ways exist to perform this transaction, and these differ from the traditional three methods we have outlined in the section above.

When you use a certificate on a web server, certificates are mostly used to maintain a secure channel between the user computer and the server. Upon creation of a secure transmission channel by verifying CRLs, web server certificates and their properties, a secure exchange of symmetric keys (or session keys) occurs, allowing data to be encrypted on the fly, and transferred from one end to the other.

The process of configuring certificates (on IIS servers) is available through a few options:

Create a Certificate Request
This allows you to create a BASE64 text file to later proceed with certificate requests by connecting to a web interface of a CA and pasting the text within the request itself on the CA.

Create Domain Certificate
This option allows connectivity to an internal CA and to perform a certificate request directly to that CA. If the CA is not available, you can also save a text file, but it assumes the process is going to be completed internally.

Create a Self-Signed Certificate
This one does not involve communicating with a CA for a certificate request, but rather deploying a local type of certificate to the server. This option should be used with caution, as it is not very scalable and secure, since any user browsing the “secure” website has no way of authenticating to the other end. Thus man-in-the-middle attacks can occur. It is best to use self-signed certificates only for testing purposes.

Complete Certificate Request
This option by itself is not so much to create a request, but rather to use the certificate provided by your CA to install it properly on the web server and complete that request.
Web Interface
Requesting the certificate for web servers through the interface is possible, as outlined in an earlier scenario. In that case, you can provide the BASE64 text file generated from the certificate request creation wizard, or create a new request and import the certificate into the computer store once approved and generated.

Once a certificate is acquired, it is critical to create a new binding for the website (using SSL) and to select the appropriate certificate you would like to use for that website.

Client Authentication
This type of certificate involves a certificate template with application policies designed to provide use of certificate to open web pages, connect to VPN tunnels, use certificates upon accessing resources, etc.

Data Encryption (Basic EFS)
It is possible to use a certificate template designed to allow you to encrypt and decrypt files and folders on disk, based on the Microsoft’s EFS technology.

IPsec/Computer Certificates
It is possible to configure authentication between computers by using IPsec certificates at the machine level.

EFS Recovery Agents (DRA) and Key Recovery Agents (KRA)
When file and folder level encryption is enabled, it is good practice to have defined a DRA (data recovery agent), which is authorized to unencrypt files in case users are not able to utilize their own certificate to decrypt the files. While, KRAs are allowed to restore keys from archive and recreate lost certificates.

Enrollment Agents
These certificate types allow entities to configure smart cards for users within an organization.

Kerberos Authentication
This type of certificate is generated to protect the authentication phase at the user and computer level, as sessions are created on the domain to prevent eavesdropping on user credentials.

Conclusion
As you have seen in this white paper, the Public Key Infrastructure (PKI) can become a fairly complex environment and deployment, as it involves multiple components that all need to work together smoothly. Understanding the theory of how it all works will help in building a solid foundation for this complex technology in your organization. Although there is always something new to learn in PKI, I hope my white paper has given you a great overview of the core configuration of your Microsoft CAs.
Learn More
Learn more about how you can improve productivity, enhance efficiency, and sharpen your competitive edge through training.

- Configuring Advanced Windows Server 2012 Services (M20412)
- Administering the Web Server (IIS) Role of Windows Server (M10972)
- Cybersecurity Foundations

Visit www.globalknowledge.com or call 1-800-COURSES (1-800-268-7737) to speak with a Global Knowledge training advisor.

About the Author
Boris Gigovic (MCTS, MCITP, CCNA) is a Global Knowledge trainer focused on Microsoft, Citrix, and Windows technologies. With over 10 years of experience in the field, Boris is in high demand as an IT consultant conducting corporate security and network audits in the Montreal area.