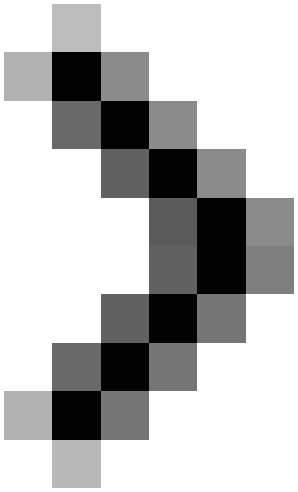
PKI Tutorial - Part 2: Design

In the first part of this PKI tutorial series, we have an overview of how to prepare and plan your PKI. In this second part, we will continue the introduction with a little more technique. Ch & uac



Part 1: Planning

Martin Kiaer

In the first part of this PKI tutorial series, we have an overview of how to prepare and plan your PKI. In this second part, we will continue the introduction with a little more technique. We will look at some PKI design issues. Throughout this article, we will introduce you to avoid the most common mistakes in the design process.

Design a PKI

When designing a PKI, there are many things that you need to consider:

1. What will the CA architecture look like (eg number of CAs and what roles will be available)
2. How do you want to protect as CA private keys
3. Where do you want to create publication points?

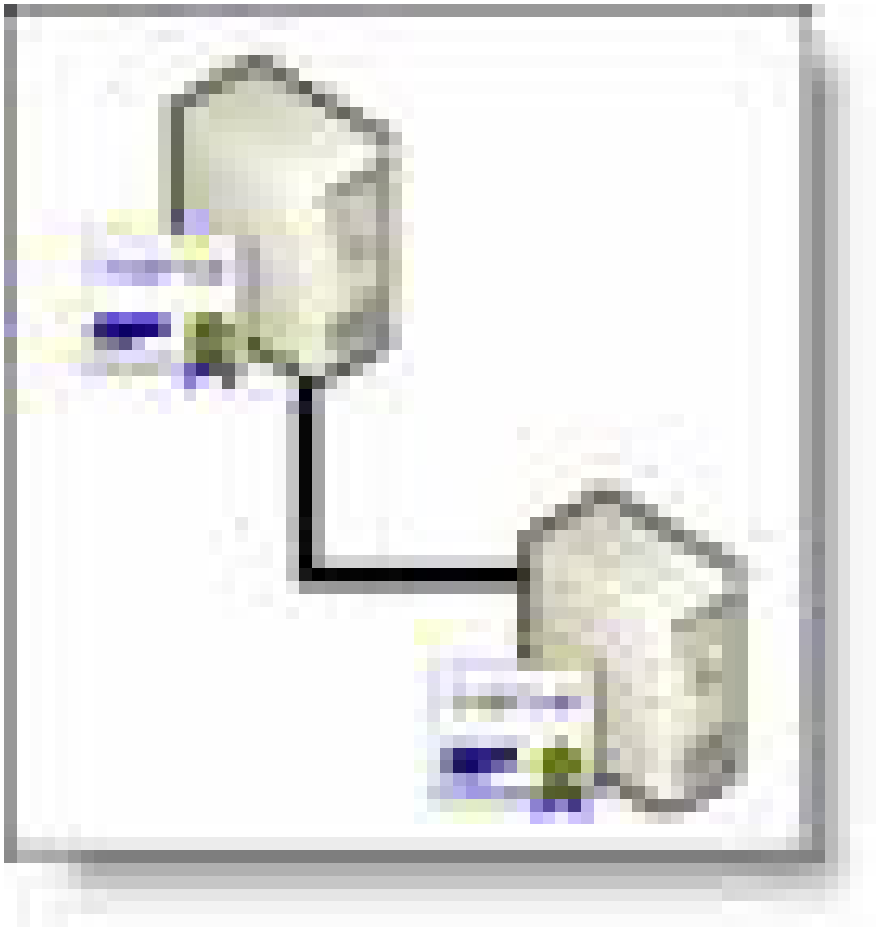
Let's take a closer look at these issues

Create a CA architecture with good scalability

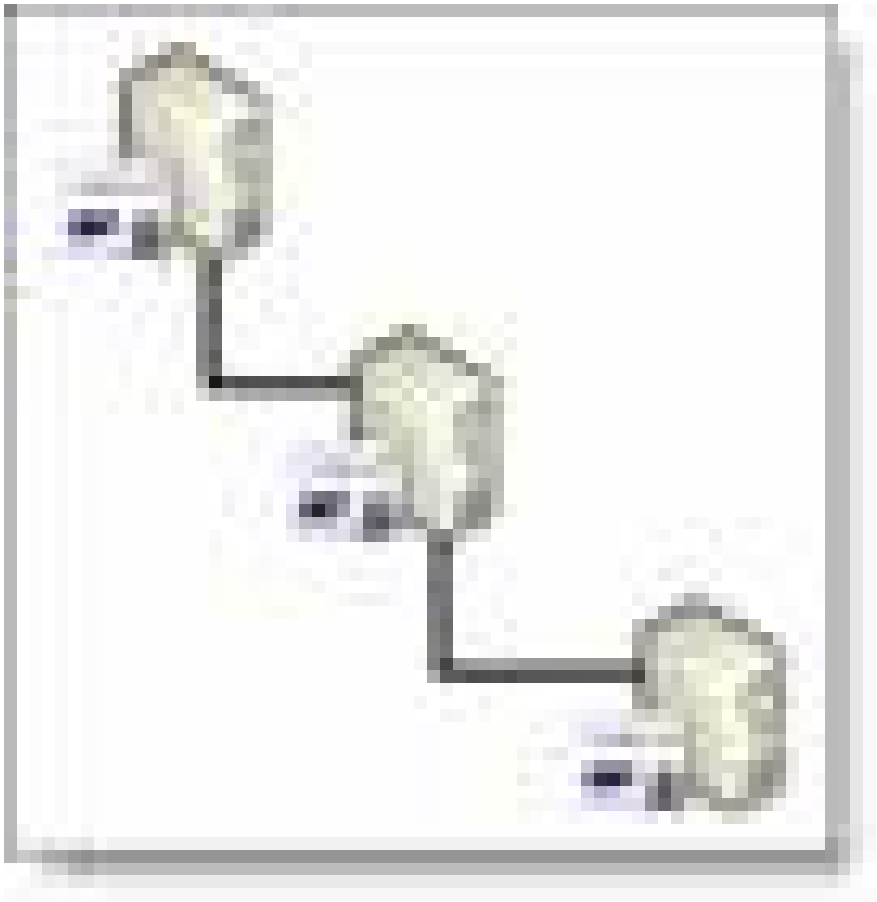
The number and levels of CA you should base are based on your security and availability requirements. You should try to organize your architecture according to what is needed. There is really no best experience in choosing how many CA levels you need, but it's rare for anyone to need more than four levels; however, there is a rule we listed in table 1 below that helps you stay on track:

CA level Caption

- Low security
- The requirements for lower CA security are lower
- Includes a root CA
- Small number of certificate requirements



- Average security
- Includes an offline root CA and subordinate CAs online
- Offline root CA is removed from the network
- Offer online CAs
- There are two or more CAs issued with each certificate



- High security
- Includes offline root CA and offline policy
- One or more secondary CAs online
- Suitable for large geographic areas and high security organizations

Table 1 : Number of necessary CA levels

As an additional note, you may recall from Part 1 of this series, there is something called a certificate policy that describes how and who issues / distributes certificates to a topic. (eg topics about existing users, computers, devices, .). If you ever thought you needed more than one certificate policy because of validity, geography, organization, or usage of a certificate, you need to define a three-level architecture that will need up to 2 or more policy CAs at level 2 (policy CAs).

When executing a PKI, you will always have to start with a root CA, which is completely unimportant when we deal with a level 1, level 2, or level 3 PKI architecture. When the root CA is always the root CA and Usually executed by itself, you need to protect the original CA's private key as best as possible. This should always be taken into account, regardless of how many levels of your PKI architecture. If your PKI architecture consists of two or more levels, the root CA needs a minimum number of accesses, since only the new level CAs require access to the root CA. However, as the distance from the root CA increases (ie there are many additional levels), security requirements will decrease and access is increased for secondary-level CAs. This will be an important factor when we start installing CAs, which we'll cover later.

CA private keys

Before installing a CA, you should have a plan on how much the size of the CA private key will be and how it is protected. Let's take a look at the key size, which is very important for security and compatibility reasons. Table 2 below lists recommended key sizes:

CA role Root CA 4096 **Policy CA** 4096 **Issuing CA** 2048 **Policy CA** 2048

Table 2: Some CA key sizes

Typically, key size 4096 is recommended to use to increase security, especially for root CAs. However, this can generate incompatible issues, for example with Cisco network products (depending on which version of Cisco IOS is being used). This is because many products of the third production group have problems managing key sizes larger than 2048. And when network devices can be integrated in solutions like 802.1x for valid problems. and in principle, the key size will be important. So you have to make sure that you know what equipment will be used and what restrictions might be imposed on certificate management before you start executing a PKI.

Once you have defined the size of the CA key you want to use, we will look at how the CA's private key will be protected.

Secure private key CA

By default, CA uses Microsoft's Cryptographic Service Provider (CSP) and protects its private key with the help of Data Protection API (DPAPI). This causes a problem, since all members of the Administrators group have access to the private key of CA and any member of this group can export the CA private key, possibly Create a new fake CA and it comes with fake certificates. Other security challenges also exist as buffer overflow attacks from malicious software.

So what should you do? You need to balance security requirements with cost and usability related to CA's private key protection. In Table 3 below, we have listed some of the most common methods to protect CA's private key. We will let you evaluate how to best protect your CA private key. Remember that this may be the most important component to protect your PKI.

Protection method Strengths Weaknesses Save local certificates - Easy to execute (default)
- Low cost - Low security
- CSP is only FIPS 140-1 compliant **Chip-based authentication**
(Smart card or USB Token) - Easy to execute
- Low cost
- FIPS 140-2 compliant - Low physical security because smart cards can be easily stolen or dropped.
- Requires physical presence when certificate services are started
- CSP requirements are particularly FIPS 140-2 compliant and Microsoft certificate support services. **Virtual machines are encrypted** - Easy to deploy
- Low cost
- No hardware
- Dependency
- FIPS 140-2 compliant - Average security
- Vulnerable by analog attacks because virtual computers include easy-to-lose hard drive or DVD or steal **hardware security module (HSM)** - Very high security level
- FIPS 140-2 level 2 and 3 compliant
- Can be PCI or LAN

- Can be used as helpers for SSL. - High cost (depending on configuration)
- Requires careful and careful planning

Table 3: Some common methods to protect private keys CA.

In addition to the methods listed in Table 3, you also want to increase the security of CA by ensuring all CAs, except for issuing CAs that are kept offline. This way they will appear less on the network and only connected to the network when the CRL and have issued certificates for the CA when there are new changes needed with your PKI. Most of the original and policy CAs are turned off completely, but this will depend on how good your security level is and how well the private keys of the CA are well protected. How hard is it to trust.

Place to create publication points

The last area that we will focus on before starting the PKI implementation is the location of the Certificate Revocation Lists (CRL) and the public keys of CA. This can be understood as distribution points for Certificate Distribution Points (CDP) certificates. There are different protocols we can use to define CDP and they are listed below:

1. HTTP
2. LDAP (in the normal Microsoft world means Active Directory)
3. FTP
4. File share (SMB)

In Figure 1 below, we have demonstrated where the CRL is published and the CA public keys along with the order of the recommended protocols to use with it.

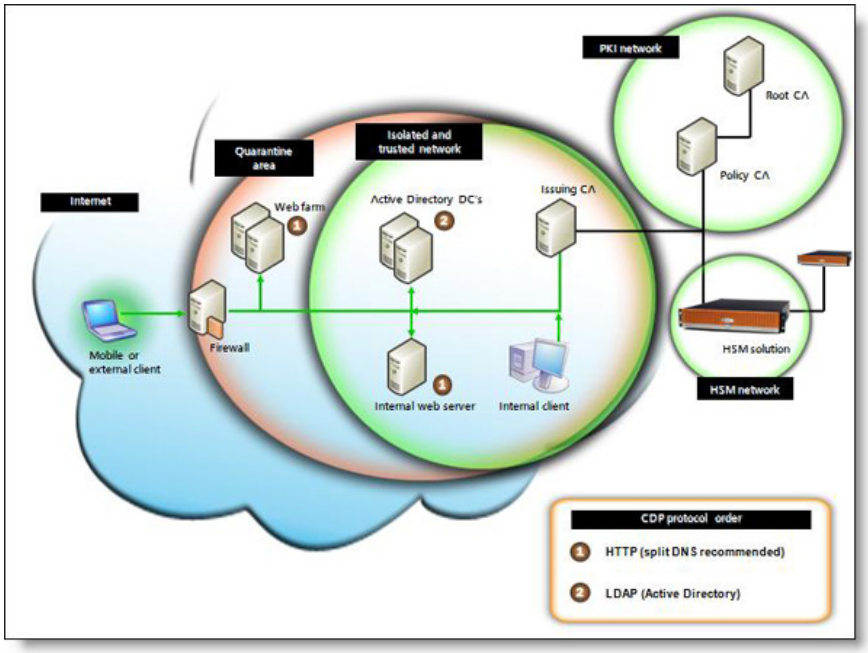


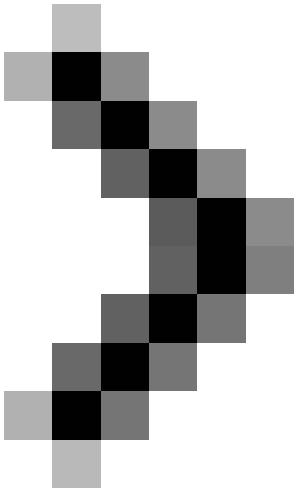
Figure 1: Recommended CDP protocol order

As you can see, the main protocol recommended here is HTTP and there is also a reason for using this protocol: HTTP is the best protocol for both internal and external publishing points. It is perfect if you need to issue certificates for both internal and external users at the same time. Especially external usage is important to consider, because you will have to make sure that the certificates used for VPN, NAQ or Wi-Fi access are checked for cancellation before the user is allowed. Access to the internal network. You should also note that if a CDP is not available for the protocol that is already issued, it will time-out (usually after 30 seconds) and move to the next protocol on the list. So by having the right configuration from the beginning is an important and necessary thing for user perceptions, because CRL can be checked from internal and external locations without time-out issues. and there is no loss of your network security settings. However, if you have a reason to choose the default protocol as LDAP, there is nothing to worry about especially if your PKI will only be used for internal purposes. You also need to be careful that if you use an Active Directory that integrates PKI and issues certificates to external users, they may need to execute LDAP queries against your Active Directory (assuming that You use Active Directory as a CDP LDAP repository.

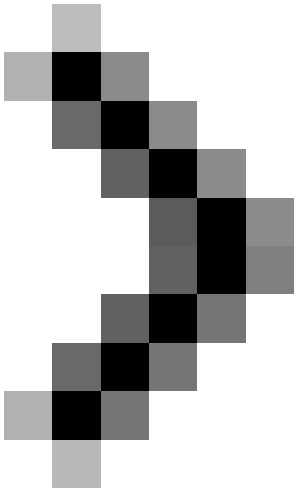
However, you should make sure you have a backup web server installation, using DNS round-robin settings when your preferred protocol is HTTP. If you want to use LDAP, you need to have a backup installation when there are two or more controllers in your domain.

Conclude

In this part 2, I have shown you some useful tutorials and tips on how to best design a PKI with both offline and online CAs. You should study the tables and figures in this section because they contain lots of details needed for practice. In the next part of this series (Part 3), we will use the best practices for designing from this part two and show you how to install PKI step by step.



Part 3: Installation



Part 4: Troubleshooting

You finished reading the article "**PKI Tutorial - Part 2: Design**" edited by the **TipsMake** team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.
