Offline Certification Authority Best Practices

White Paper

David Wozny

March 2011

Candidate architectural design solutions and best practices for deploying Offline Certification Authorities secured with Thales nShield Hardware Security Modules

SPONSORED BY

THALES

Table of Contents

Introduction

Anchoring Trust with Root Certification Authorities

Root Certification Authorities (CAs) are deployed as trust anchors in Public Key Infrastructure (PKI) solutions used to issue digital certificates in an organization. Root CAs in a simple PKI hierarchy certify sub-ordinate CAs (often referred to as Issuing CAs) which then issue digital certificates to "end entities" such as Web servers, workstations or individuals. These entities use digital certificates, and the public / private keys associated with them, to identify themselves on the network or perform other functions such as encrypt and/or sign information or messages.

The physical separation of trust anchor (Root CA) and Issuing CAs in the PKI hierarchy is designed to minimize the risk of compromise of the Root CA and therefore of the entire PKI. As part of the risk mitigation of the Root CA it is protected using both physical and logical controls.

The Root CA is unlike any other Information Technology asset deployed within an enterprise; implementing a Root CA demands critical security considerations first and foremost. To ensure suitable physical isolation, a Root CA is ordinarily deployed offline (not connected to any networks). The private key material[1] which constitutes the primary security enforcing component of a Root CA (it is used to sign certificates issued by the CA) must be protected within a tamper proof environment for cryptographic isolation – generally established by the use of a Hardware Security Module (HSM). Furthermore, the HSM should provide suitable controls to ensure correct authorization of access to the protected key material.

HSMs are cryptographic devices that are connected to a system to provide extremely strong protection of cryptographic key material and to overcome the inherent weaknesses of performing cryptographic operations in software. HSMs provide a physically tamper-proof security envelope within which key material can be stored and used securely subject to multi-user operational controls that enforce separation of duties and authorization policies. Despite a CA being deployed offline, it is still essential to employ HSMs to protect signing keys.

It is essential to always be mindful that the PKI is often used to issue certificates where ultimate reliance is placed in the trustworthiness of the credential - whether that is to assert identity to high value systems (authentication), encrypt security sensitive data (confidentiality) or impart confidence that security sensitive data has not been modified after a signing operation has been performed on it (integrity). Loss of confidence in the integrity of a Root CA could force some or, quite often, all issued credentials to be revoked and reissued which implies a high level of business disruption for potentially a prolonged period of time incurring significant direct and indirect costs.

It should be stated that nowhere is it written in *PKI lore* that Root CAs must be offline[2] – it's a design approach influenced by the assurance required of the trust anchor, largely derived from the potential value of the operations outlined previously. Whether a Root CA is implemented online or offline in no way structurally affects the logical PKI design - such as the chain of trust from a leaf certificate to a Root CA. Storage of Root CA keys in an appropriately rated (e.g. FIPS[3] 140-2 Level 3) HSM adds a further level of physical protection to the logical protection of the Root CA concept.

[1] Compromise of a CA's signing key would make it relatively easy for a hostile party to impersonate the CA and issue bogus certificates to untrusted entities

[2] This paper only considers the offline CA scenario - with the CA's private key protected by an nShield HSM

[3] Federal Information Processing Standard (FIPS) 140-2 is a U.S. government computer security standard used to accredit cryptographic modules; see http://csrc.nist.gov/groups/STM/cmvp/standards.html#02

Combined, these two tactics yield what has come to be regarded as an industry standard for commercial strength Root CA deployment.

Being deployed "offline" ensures there's absolutely no opportunity for network based attacks directly on the Root CA. It's worth noting however that while Root CAs are deployed offline, they periodically publish a CA certificate and Certificate Revocation List (CRL) which must be distributed to online repositories[4] and retrievable by Relying Parties[5]. Unavailability of this material can be a potential denial of service threat against parties relying upon certificates issued by the PKI.

This document focuses on describing the architectural choices and considerations in deploying a CA hierarchy rather than pure PKI security elements such as key lengths or CRL distribution points – the exception being the short discussion regarding the applicability of a "three-tier PKI" in the next section. The purpose of this document is to articulate trade-offs in cost, utility, and security involved in offline CA architectural design.

[4] Such as HTTP servers or LDAP directories

[5] A Relying Party is an entity that must make a decision upon whether to trust a certificate presented to it

The Case for Three-Tier PKI

Before getting into the nuts and bolts of architecting offline CAs, it's worthwhile considering whether there is justification for a two or three-tier solution. As a basic premise, a PKI design should start with a two-tier solution - consisting of a single Root CA and however many Issuing CAs are deemed necessary[6]. Ability to cross-certify with other organizations, ability to enforce strict certificate policy constraints either intra-organization or via the cross-certification can all be achieved using a two-tier approach as illustrated Figure 1.

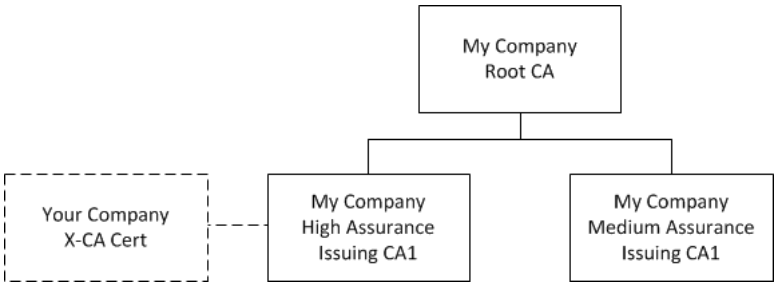

Figure 1: Two-Tier PKI

There *are* situations where an intermediate tier of Policy CAs is justifiable, for example you may have an extensive infrastructure of high assurance CAs which needs to be trusted via a cross-certificate[7] as shown in Figure 2; in this circumstance it may be sensible to cross-certify at a Policy CA rather than have multiple cross certification "instances".

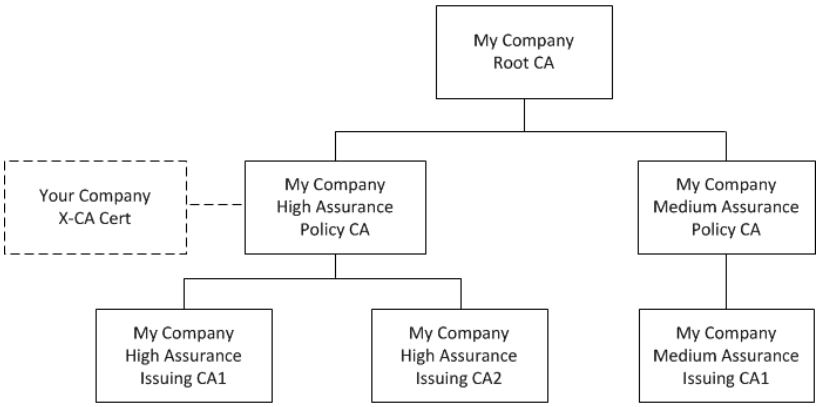

Figure 2: Three-Tier PKI

Many people consider it a best practice to implement a three-tier PKI based and a cursory read of older Microsoft PKI documentation on the topic would lead one towards three-tier architecture. However, this excerpt from the latest Microsoft *Implementation Planning and Design Guide for Active Directory Certificate Services*[8], clearly illustrates that there are trade-offs and the designer should consider both approaches.

"Designing a three-tier hierarchy with intermediate CAs increases the complexity of the environment. Requirements to implement different policies can be implemented in a two-tier hierarchy with additional Issuing CAs. The Windows Server product group states that there are no scale limitations that require a middle tier, so avoid using intermediate CAs unless there is a compelling business reason for doing so."

[6] The number of Issuing CAs is driven by location, volume, response time and other considerations too numerous to detail in this document

[7] Another good use of cross-certification is to migrate from legacy PKI to a new PKI instantiation

[8] See http://technet.microsoft.com/en-us/library/ff630887.aspx

One final thought on the subject of tiers… if you are simply implementing a PKI for a short-term tactical objective or for a low assurance situation such as a dedicated CA for issuing "health certificates" to support Network Access Protection (NAP), etc. it may not even be necessary to go beyond a single tier. There's nothing mandating that you must have Root CAs purely as trust anchors; in tactical situations it may be advantageous to issue end entity certificates from a "Root Issuing CA" – not a scenario which this paper considers.

Offline CA Solution Design Approaches

When considering the best approach for designing your offline CA solution, a number of basic elements should be considered:

- Server hardware or portable hardware

  There's certainly two schools of thought here… one aligned with deploying offline CA services on enterprise class server hardware and the other minded to deploy offline CA services on a portable, small form-factor platform such as a laptop

- Virtualization or "physical server" platforms

  While virtualized server platforms (such as Microsoft Hyper-V or VMWare solutions) have obvious benefits in the IT estate, we look at how applicable this approach is in the use case of offline CAs

- HSM form-factor

  The Thales nShield HSM product line essentially consists of three different form-factors[9]: USB attached (nShield Edge), PCI card (nShield Solo) and network attached appliance (nShield Connect). The newest of these products, the nShield Edge has hugely enriched the opportunity to simplify offline CA deployments

- Planned obsolescence

  It should be borne in mind that during a Root CA's lifetime (typically twenty years) the server hardware platform, HSM hardware module, operating system and CA application software will all have become *obsolete* several times over

Finally, and by no means least, consideration should be made to ensure the commissioning processes and activities of deploying the PKI in what is often referred to as a Key Signing Ceremony[10] are fully documented. Indeed, there is almost a mythology around deploying offline CAs with the rigorous processes and strict accountability – unnecessarily complex solutions can have serious ramifications for ensuring the integrity of the initial instantiation as well as on-going maintenance and potential recovery scenarios. CA implementations that are rushed and / or poorly documented, no matter how much expense is lavished on technology, are likely to be brittle in times of emergency (e.g. system restore under pressure) or to demonstrate integrity to an auditor.

From a practical point of view the fact that offline CA(s) have no requirement to integrate with network services, in some ways, reduces the complexity of the system. It can be argued however, that the balance of complexity essentially transfers to suitable physical access controls to provide the requisite assurance.

[9] The various features and specifications that differentiate members of the Thales nShield family of HSMs are outlined in the Appendix of this document

[10] A Key Signing Ceremony is an approach used when commissioning systems utilising sensitive cryptographic key material (such as with a CA) whereby extremely detailed installation documentation is required and every action is precisely executed and observed

So, although it can be recognized that offline CA design is perhaps fundamentally less technically complex than that of network connected CAs – it does not mean deploying them is necessarily straightforward. The bulk of this paper is intended to inform the reader of the type of decisions needing to be made when considering physical deployment of offline CAs and give insight into potential best practice approaches.

The remainder of this document considers the following three offline CA deployment options:

- Server with Thales nShield Solo HSM
- Laptop / desktop with Thales nShield Edge HSM
- Virtualized server with Thales nShield Connect HSM

Server Based Offline CAs

Overview

Traditionally, the platform of choice for deployment of an offline CA has been a dedicated server; with resilient hardware, enhanced supportability and high performance processing. Whilst there has clearly been a significant shift in the market towards consolidating servers using virtualization technology, there are still many situations (and an offline CA is one) where dedicated "tin" is still a very legitimate approach to provide the requisite security, longevity and manageability platform for an offline CA.

Architecture

The nShield Solo is an HSM with the form-factor of a PCI[11] card and is generally the preferred member of the nShield family when an enterprise server chassis approach is taken to deploying offline CA capability. The use of a server chassis with easily pluggable disk drives makes viable the deployment of each offline CA (assuming there are Policy CAs as well as a Root CA) on independent disks which would be bootable in their own right and never present in the server chassis simultaneously.

Deploying offline CAs on a server chassis means that there may be physical access challenges to be overcome which may necessitate establishing dedicated server racks etc., with suitable controls to ensure that the high assurance benefits of implementing the CAs offline are not negated.

The server chassis approach for implementing offline CAs is illustrated in Figure 3; although a rack form-factor server chassis is illustrated here, there's no reason why a tower server could not be used.

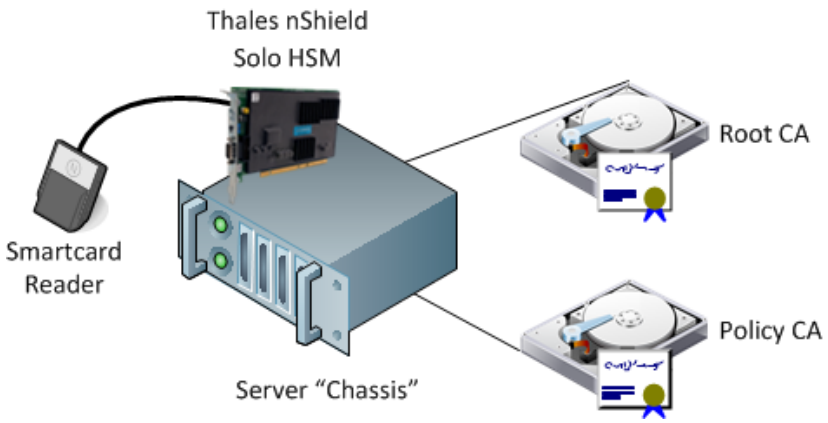

Figure 3: Server Chassis Hosted Offline CAs

Discussion

As far as simplicity of server build / demarcation between offline CAs are concerned, the server chassis approach has great merit and is entirely valid. For many years the server centric approach combined with the nShield Solo HSM cards has been the mainstay of offline CA deployments

[11] There are both PCI and PCIe variants available

leveraging Thales nShield HSMs. Nevertheless, when considering issues of longevity the primary architectural challenge experienced has been that bus architecture within servers inevitably moves through generations relatively quickly and therefore a PCI card form factor HSM deployed in a server, say five years ago, would be unlikely to fit in a modern generation server which are generally only equipped for PCIe based peripherals.

A further consideration when evaluating how to design longevity into your offline CA solution is the backup / recovery methodology employed. One positive school of thought is to make it relatively simple to recover / move / migrate your offline CA from one platform to another – termed a component based recovery, it is described in the following section.

Component Based Offline CA Recovery[12]

Introduction

It's likely that during its twenty year lifetime (typically), the Root CA will go through the following events:

- Three or more hardware refreshes (especially so for laptops)
- A hardware failure
- Two or more operating system platform (e.g. Windows Server version) cycles[13]

For these reasons, it's imperative that a solution is in place for a component-by-component recovery / migration installation of a CA to complement any "full system" type restore methods employed. Placing suitable emphasis on perfecting an efficient recovery process which is available in the circumstance of a hardware failure, operating system corruption / migration, lost administrator password, etc. means that the awkward decisions around whether to use expensive tape based backup hardware and third party backup software can often be avoided entirely by backing up critical files to removable (typically DVD) media.

While this paper is Certification Authority vendor agnostic, it is still instructive to illustrate what is meant here by a component-by-component based recovery by using the example of an Active Directory Certificate Services (AD CS) implementation on Windows Server 2008 R2. Other Certification Authority platforms would require a different backup regime than that described.

Material to Backup

In the case of AD CS, the solution should be designed with a component based recovery approach in mind and this is made easier by separating the components to be recovered onto a second logical volume, e.g. "D". This second drive doesn't need to be on a separate physical disk and the principal reason for its use is solely to make the backup / recovery process clearer; you'd store the following material on "D":

- Thales Key Management Data (HSM configuration files and key blobs[14])
- CA database (and logs)
- CA database backups (run every time the offline CA is operated)
- CA certificate and CRL (by configuring file based AIA and CDP publication paths)

[12] The recovery process described here is applicable to all offline CA deployment strategies described in this paper

[13] While it's true that upgrades could be done cumulatively (i.e. in place upgrades) - there would almost certainly come a time when it's desirable to have a *clean build*

[14] In this context, key blobs are the encrypted application keys (encrypted by high assurance internal HSM keys instantiated in the Security World) at rest - awaiting loading into an HSM for decryption, authorization and activation

When the aforementioned components are established, backup essentially consists of copying a few megabytes of data in folders on your "D" drive to a CD / DVD.

Recovery Process

Recovery is straightforward; the process outlined below can recover / redeploy / migrate an offline CA in a matter of minutes once the Windows server platform is re-established.

- Install Windows and configure the platform as before (hostname, logical drives, etc.)
- Copy the files from the component backup (CD / DVD) to "D"
- Install Thales middleware, then connect to the HSM
- Re-associate the CA certificate with its HSM protected private key (CSP / KSP repair)
- Install AD CS specifying the existing certificate and private key
- Replay the AD CS registry configuration[15] then restore the CA database

While the sequence may seem more complex than simply "insert a tape and press the restore button", there's total confidence in the outcome as the anxiety of waiting for a restored system to come back up without errors is avoided. Furthermore, the dilemma of backup tape / drive hardware decisions over the lifetime of the Root CA are avoided as it is also reasonably safe to assume that during the twenty year life of the Root CA that backup solutions will have iterated two or three generations - bringing their own headaches!

[15] Best practice is to configure CA registry settings programmatically (e.g. with batch files) rather than manually editing the registry or using the Certification Authority management snap-in

Laptop Based Offline CAs

Overview

The offline Root CA scenario has strong requirements for physical isolation of its computer hardware. Recent improvements in laptop technology and new form-factor HSMs have opened up the possibility of deploying a Root CA on a laptop which can be stored in a safe and only brought out when needed to be operated (periodically). Given the cost and logistical advantages of this relatively new model it is an approach worthy of serious consideration.

Architecture

While the network attached appliance based nShield Connect could legitimately be used in combination with a laptop, the clearest and most practicable HSM form-factor is the USB attached nShield Edge unit. The nShield Edge provides all the capability required to support an offline Root CA and shares the same level of FIPS certification and software platform as other members of the nShield HSM family. The primary concession of the nShield Edge over its most immediate sibling, the nShield Solo (PCI card) is its signing performance, which in the context of a Root CA is a negligible factor. The architectural simplicity of connecting a laptop to an nShield Edge HSM is illustrated in Figure 4.

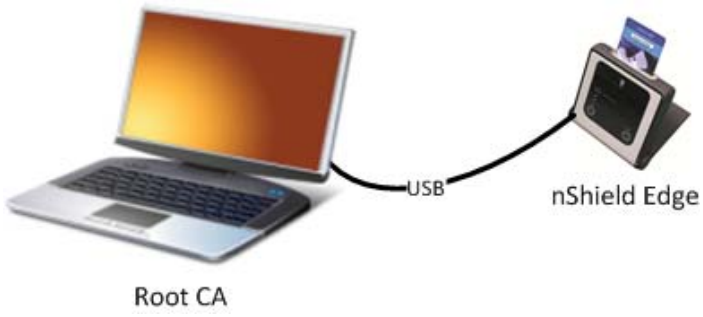

Figure 4: Laptop Native Root CA

The next consideration is whether to implement the server operating system hosting the Root CA natively on the laptop, e.g. install Windows Server 2008 R2 on the laptop or to choose a virtualization path. Attention must be paid to whether the Windows Server OS is supported on the laptop hardware of choice. There are multiple laptop offerings from the major equipment vendors that are certified interoperable with MS Windows Server 2008 R2, any of which would be a suitable platform for an offline CA.

Another variation of the laptop architecture is for situations where you have multiple offline CAs, as in the circumstance of implementing Policy CAs as well as a Root CA. In this situation, you might choose to implement your Root CA on a dedicated hard disk caddy[16] which would then be removed and replaced with a separate hard disk caddy upon which the Policy CA is installed. In this scenario, the laptop is essentially a stateless host, the nShield Edge would be shared between the two offline CAs with no re-configuration whatsoever, assuming both the Root CA and Policy CA are in the same Security World[17].

[16] As solid state drive technology matures this would likely be a more elegant solution

[17] A Security World is an administrative security boundary; it enables multiple nShield HSMs to participate in a single management regime and protect common keys while simultaneously allowing operational separation of the HSMs

Another approach while still considering the laptop architecture is to install Windows 7, for example, on the laptop, then implement virtualization platforms such as Microsoft Virtual PC or VMWare Workstation[18] as shown conceptually in Figure 5. The rationale for choosing Virtual PC or VMWare Workstation is that they support connecting devices to guest operating systems via the host's USB bus – which is something that enterprise virtualization platforms generally don't do.

To provide suitable "separation" between the Root CA and Policy CAs, it might be appropriate to use a variation of the approach described earlier for multiple offline CAs whereby each virtual guest *image* is stored on a discrete SSD / hard disk caddy.

Figure 5: Laptop Virtualized Offline CAs

Discussion

Prior to the availability of the nShield Edge, the laptop architecture approach was limited to use with an nShield Connect. The nShield Edge opens up a significant opportunity to implement "better-sized" and more cost effective solutions for offline CAs where high cryptographic performance is not required. In fact, regardless of the offline CA host's form-factor (server / desktop / PC), the nShield Edge is generally an excellent fit as a direct attached HSM given its form-factor and price point.

The laptop approach clearly gives significant flexibility with respect to purposing offline CA servers while using a small footprint host – although this can also be said of many of the newer small foot print desktop and "luggable" computers available in today's PC marketplace. As servers become smaller one could just as easily deploy a Root CA on a small form-factor PC as a laptop.

Of all the candidate solutions presented in this paper, the lifetime of laptop hardware probably has the fastest obsolescence profile and the greatest vulnerability to hardware failure. The shortcomings of laptops as a hardware platform might steer you down the virtualization path (to abstract the offline CA from the laptop hardware). Alternatively, implementing the component based recovery process described earlier ensures that it's relatively straightforward to move the offline CA from one platform to another – whether the circumstance is demanded due to a hardware failure, hardware refresh, operating system upgrade, etc.

[18] These virtualization platforms may not have the robustness or longevity of enterprise class virtualization platforms

Virtualization of Offline CAs

Overview

Virtualization has obviously made massive inroads into IT infrastructure and it's not the intent here to repeat all of the potential benefits which can be realized in a general context, however, there's purpose in examining the value proposition of virtualization for "server based" offline CAs.

Architecture

Firstly, it's worth noting which HSM form-factors are available in the context of server virtualization. For example, PCI card based HSMs are not supported by either VMWare vSphere / ESX or Microsoft Hyper-V. It's a similar story with USB based HSMs since USB devices cannot be attached to guests on either of the two aforementioned virtualization platforms[19].

So, the most appropriate HSM form-factor when choosing to go down the virtualization path is the network-attached nShield Connect. Taking the basis of a single offline CA (Root CA), Figure 6 illustrates the principal components required to support deploying the aforementioned CA in a VMWare ESX solution. One thing's quite clear - this approach has a relatively high "platform overhead" in the context that a single Root CA depends upon:

- A VMWare ESX host
- A (Windows) management console for the ESX host[20]
- A Remote File System (RFS) guest Operating System (OS) on the ESX host – this is a requirement when deploying an nShield Connect HSM
- The Root CA guest OS itself

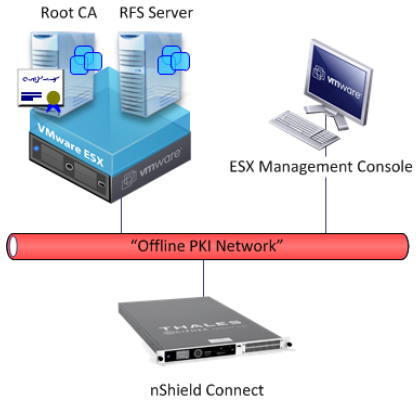

Figure 6: nShield Connect for Offline CA Virtualization

There may also be a network switch required to plumb the components together; generally speaking the cost and complexity of this type of solution to satisfy instantiation of a single Root CA might be deemed unjustifiable.

[19] VMWare Workstation, VMWare Server and Microsoft Virtual PC do support attachment of USB devices to a guest OS and are considered in the "laptop" candidate solution

[20] For a Hyper-V solution there is no requirement for a separate management console

To realize greater value from the nShield Connect deployment, customers often state a desire to leverage the nShield Connect by utilizing the same unit for their online CA infrastructure as well, an example of which is illustrated in Figure 7.

There are a number of ways of implementing the leveraged nShield Connect; it is possible to take advantage of the fact that it has a second (non-routable) network interface which can be connected to the *offline network*[21] while the first interface is connected to the *online network* (as in Figure 7). Another option would be to deploy the nShield Connect in a DMZ off a firewall which straddles the two "networks" and configure appropriate rules to isolate the offline network from the online network.

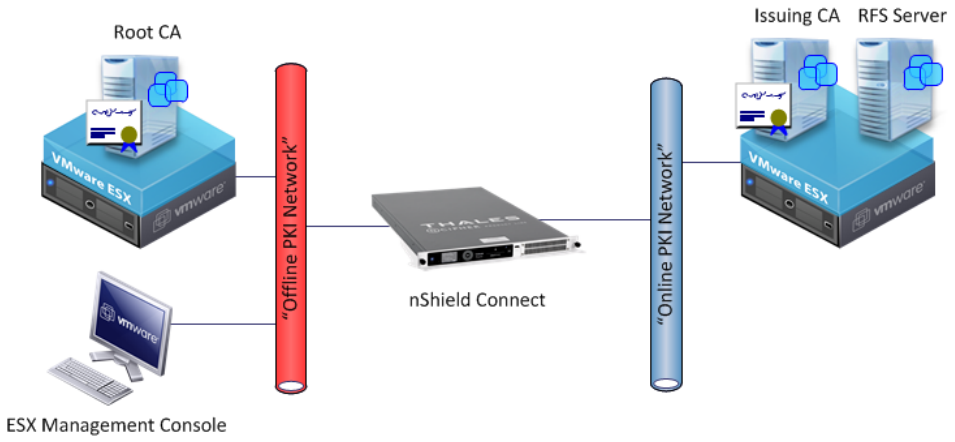

Figure 7: Leveraging nShield Connect for Offline and Online CA Virtualization

So, could deploying a suitably rated and configured firewall between the Root CA and extended network suffice? Typically no - PKI is governed by Certificate Policy (CP) which for anything higher than a low assurance scenario will ordinarily demand that proper separation (air-gapping) is rigidly enforced. There are no hard and fast rules around this and it may be that for a PKI being deployed solely for "internal use" that a firewall break is sufficient; however, if future requirements demand the PKI extends via cross-certification to a wider base, partners may not be satisfied with the firewall approach and their certificate policy may disallow the aforementioned cross-certification.

Discussion

It's important to ensure evaluation of the relative merits of architectural solutions are not simply validated by their technical viability, but more so by their applicability in the context of an extremely demanding operational and management PKI regime. The solutions described in this section are often passionately defended by "techies" and they certainly have much merit when building demonstrators or development rigs to flesh out the bones of a solution. However, put yourself in the position of developing a key signing ceremony around the aforementioned approaches and it puts a whole different slant on proceedings.

Let's look at the primary perceived advantages which are mooted for the virtualization approach for server based offline CAs; these are more conservative than those benefits which would be identified for online CAs where virtualization provides huge opportunities for extracting value.

[21] In this context, the term "offline network" is used to describe a De-Militarised Zone (DMZ) which provides a degree of isolation of the Root CA; the approach is to leverage the nShield Connect for the Root CA as well as the Online CA while protecting the Root CA from potential threats in the local network

- Leveraging nShield Connect

  Can the Root CA truly be described as offline if there is clear evidence of connectivity to wider networks as shown in Figure 7? Furthermore, it is harder to apply additional levels of physical security in this scenario, leading to an increased exposure to tamper attempts. To mitigate this, the virtualization platform hosting the Root CA could legitimately be powered off and removed to a secure location and only re-introduced at such time that it needs to be operated.

- Hardware abstraction

  Having the Root CA virtualized makes moving it from one server to another a relatively straightforward task. But, it's also likely that during its typically twenty year lifetime, the Root CA will go through two or three operating system platform cycles and also the virtualization platform will go through a similar sequence. A component based recovery solution (described earlier) offers greater flexibility to mitigate the potential for hardware / software failures / planned upgrades at the Root CA.

Conclusions

Best Practices

Hopefully this paper has given food for thought regarding different options available for deploying offline CAs in combination with Thales nShield HSMs and perhaps shot down a few myths. Clearly, this paper can't address every possible approach that can be taken, but does provide best practice guidance and candidate solutions which will help you deploy offline CAs with Thales nShield HSMs for you and your customers.

All three presented options are legitimate approaches and given suitable parameters can be justified; some of the influencing parameters which may drive you to take a certain approach are included here:

- **Small form-factor host**

  If the over-riding requirement is to use a laptop for hosting the offline CA, then the nShield Edge HSM is generally the best choice

- **Virtualization technology**

  In this situation you would ordinarily use an nShield Edge HSM if you have a virtualization platform that supports attaching USB devices to guests (typically the laptop virtualization scenario); otherwise an nShield Connect HSM is the realistic choice

- **Performance**

  Ordinarily performance comes far down the list of requirements for an offline CA, however, if this was ever the case then the nShield Solo HSM or nShield Connect HSM provide a high cryptographic performance platform

- **Isolation**

  In situations where "embedding" of the HSM into the server unit is required, the nShield Solo provides unambiguous coupling between the HSM and host

- **Leverage nShield Connect Unit(s)**

  Customers are generally keen to get maximum utilization out of their HSM investment; in this circumstance it may be advantageous to connect an offline CA to the non-routable network interface of the nShield Connect unit

A couple of further important points to bear in mind when making your design choices:

- Consider the applicability of your solution in the context of a key signing ceremony / on-going rigorous management regime. Server virtualization may appear to be a technically adept solution, but the additional complexity introduced may negate this benefit – this is always difficult to portray to anyone who has never been responsible for authoring / directing a key ceremony or managing a tightly controlled and operated PKI estate.
- Use a *neutral backup format* where possible, such as DVD (which must be stored securely); in combination with a component based recovery process this affords significant flexibility for maintaining the offline CA over the course of its lifetime.

Regardless of the offline CA platform architecture – it's clear that the Thales nShield family of HSMs provides an extensive portfolio that can fit your solution.

Appendix – Thales nShield HSM Product Matrix

nShield Edge	nShield Solo	nShield Connect
		

FEATURE	nShield Edge	nShield Solo	nShield Connect
Interface	USB	PCI, PCIe	2x GBit
Shareable	No	No	Yes
Power Supplies	N/A	N/A	2
Speed Variants	10	PCI: 500, 2000, 4000 PCIe: 500, 6000	500, 1500, 6000
CA Software Support	Entrust, Microsoft, Red Hat, RSA, and others		
Certifications	FIPS 140-2 L2 &L3	FIPS 140-2 L2 &L3 CC EAL4+	FIPS 140-2 L2 &L3 CC EAL4+

About the Author

David Wozny is a freelance security consultant specializing in Public Key Infrastructure and associated technologies such as smart card management systems. David, who is based in the UK, has led the delivery of some of that country's largest PKIs in various capacities from solution architect down to detailed engineering, as well as being author and director of many PKI Key Ceremonies.

Two of David's principal areas of technical expertise are the Active Directory Certificate Services (Windows Server PKI) platform and Thales nShield Hardware Security Modules. David is reachable by email using the address: david@wozny.org.

About Thales e-Security

Thales is one of the world leaders in the provision of information and communication systems security solutions for government, defense, critical infrastructure operators, enterprises, and the finance industry. Thales' unique position in the market is due to its end-to-end security offering spanning the entire value chain in the security domain. The comprehensive offering includes architecture design, security and encryption product development, evaluation and certification preparation, and through-life management services.

Thales has an unrivalled 40-year track record of protecting information ranging from "sensitive but unclassified" up to "top secret," as well as a comprehensive portfolio of security products and services, including network security products, application security products, and secured telephony products. To learn more, please visit http://www.thales-esecurity.com.