PKI Security Considerations for AMI, Smart Grid, and ICS Networks

Seth Bromberger
National Electric Sector Cybersecurity Organization

Stan Pietrowicz
Applied Communication Sciences

Acknowledgement / Disclaimer

This material is based upon work supported by the Department of Energy under Award Number DE-OE0000516.

The authors would like to thank the following individuals and organizations for their valuable contributions to this paper:

- Kurt Grutzmacher, Cisco Systems
- Clifford Maraschino, Southern California Edison
- Craig Rosen, Pacific Gas and Electric Company
- Mark Ward, CyDesign Labs, Inc.

Document Number	WP2012-01-01
Revision	1.4
Publication Date	12 January 2012

The National Electric Sector Cybersecurity Organization (NESCO) is operated by EnergySec with funding assistance from the U.S. Department of Energy.

ABSTRACT

The recent high-profile breaches of root certificate authorities[1, 2, 3, 4] have resulted in widespread discussion around the presumed impenetrability of public key infrastructure (PKI), secure communication, and key-based cryptography. Richard Clarke, former Special Adviser to the President on Cybersecurity, has weighed in: "My take is that you can't trust digital certificates...and it's a turning point not just for digital certificates."[5] While these breaches were limited in scope, they underscore a fundamental complexity in securing PKI components: protecting the root and intermediate certificates and the private keys that guarantee their authenticity.

This complexity is not limited just to public root certificate authorities: any organization that issues or uses certificates and other PKI components for authentication or session integrity – including those that elect to use "self-signed" root certificates – must ensure that the private keys associated with these certificates are never disclosed to nor used by unauthorized individuals or systems, and that the protection scheme used to secure these certificates is strong. Many organizations that issue certificates and other PKI credentials do not count PKI management among their core business processes. Furthermore, PKI operation is typically not identified as a critical process in business risk assessment and therefore organizations do not allocate the resources to securing cryptographic keys. While the cryptographic strength designed into PKI specifications provides a strong defense against communications interception (assuming the system design keeps pace with computing technology), governance, protection, and storage of the key material remain the weak links in the chain. Recovery from PKI root breaches, as an example, is a non-trivial exercise: once the primary basis for trust is compromised, there are no good means to re-establish this trust without the ability to rely on already-secure supply chain, communications, and provisioning processes, which is made even more difficult once equipment that is designed to trust the now-compromised root certificate has been deployed in the field.

This paper describes common problems associated with large-scale deployment of PKI technology in two sets of emerging and legacy technologies used within the electric sector: Smart Grid / AMI and ICS networks. Common weaknesses in the provisioning of PKI within these environments are described. Recommended mitigations and questions that are intended to

---

[1] http://news.softpedia.com/news/New-Stuxnet-Related-Malware-Signed-Using-Certificate-from-JMicron-148213.shtml, retrieved 1 December 2011

[2] http://arstechnica.com/security/news/2011/03/how-the-comodo-certificate-fraud-calls-ca-trust-into-question.ars, retrieved 1 December 2011

[3] http://www.coriolis-systems.com/blog/2011/08/diginotar-certificate-security.php, retrieved 1 December 2011

[4] http://www.zdnet.com/blog/london/-8216hacked-server-claims-another-certificate-authority-casualty/596, retrieved 1 December 2011

[5] http://news.techworld.com/security/3304648/dont-trust-ssl-certificates-says-us-cybersecurity-adviser/, retrieved 1 December 2011

guide further discussions among electric sector asset owners, operators, equipment manufacturers, and vendors are provided at the end of this paper.

1. INTRODUCTION AND SCOPE

Public Key Infrastructure (PKI) is a common technological foundation for assuring secure electronic transactions. It provides an overall trust framework and enables methods for exchanging credentials, proving authenticity, and providing integrity protection for user and device authentication and authorization, and the establishment of secure (encrypted) communications channels. PKI is usually described as having two primary components: a digitally-signed *certificate* issued by a trusted authority that provides proof of a claimed identity; and a public/private *key pair* that, when used securely, can establish cryptographic trust (e.g., encryption, digital signatures).

While a full description of PKI is outside the scope of this paper (the reader is referred to the "Additional Resources / Further Reading" section at the end of the document), an overview of PKI functionality and trust models, with a focus on the methods by which PKI implements cryptographically strong identification, will be useful for later discussion.

An entity (user, device, system organization) wishing to make a claim of identity to a peer provides a *certificate* that minimally includes the entity's name and public key. This certificate is signed by an issuing *certificate authority* (CA), providing a declaration by the CA that the identity presented is valid, and thus providing a *certificate chain* that shows (via a cryptographically-valid process) that the certificate presented is itself derived from a series of other implicitly or explicitly-trusted organizations. The peer therefore can determine whether or not it trusts one of the organizations attesting to the validity of any certificate in the chain, with the implication that if the peer trusts an organization higher in the certificate chain, it will trust all assertions of some set of capabilities made by any of the sub-organizations, including the original entity which desires communication.

There exists, therefore, a hierarchy of trust in PKI (see figure 1): on every device, there is a list of trusted *root certificates* which, when included correctly in the certificate chain of any entity, will allow the peer to accept the certificate of the (heretofore unknown) entity with a specific level of trust (encryption, identity assurance, etc.). The root certificates have no higher 'issuer' and therefore are signed by the root organization, called a *root certificate authority* (root CA). These root certificates are stored in each end device in a *trusted root store*. The process of signing a root certificate involves using the private key of the root authority. All certificates issued by the root authority are signed using this key. Hence, this one key, if compromised, could result in a breach of trust for all certificates underneath it.

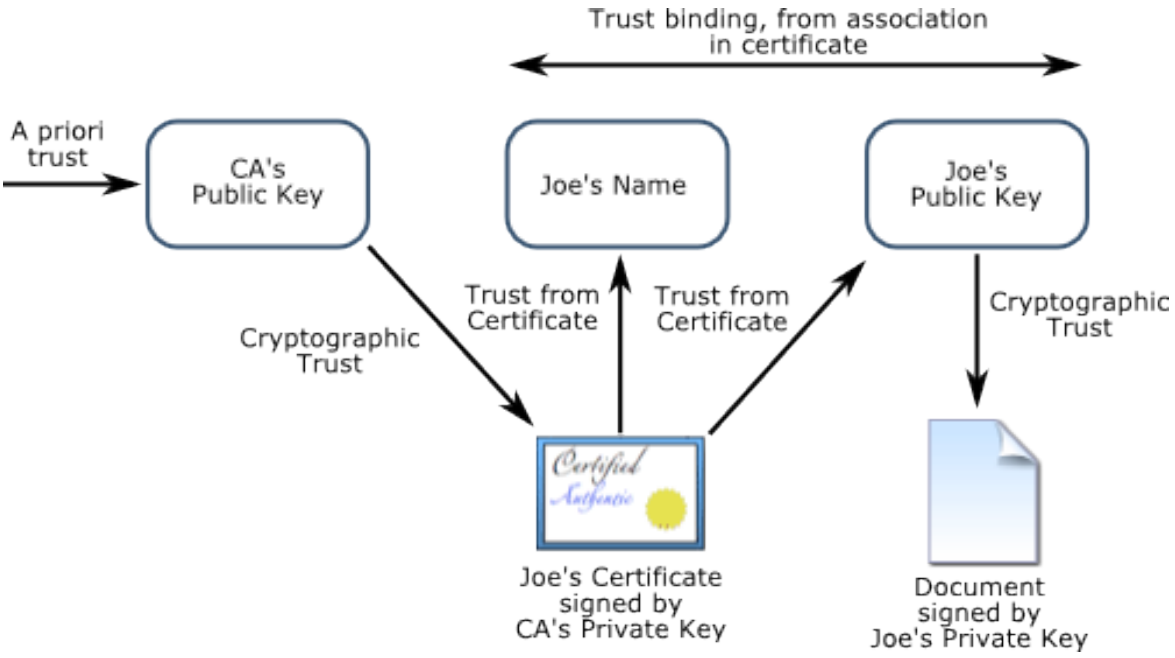

Figure 1. Trust bindings within PKI. Image courtesy of Isode Ltd. Used with permission.

2. IMPLICATIONS OF TRUST

Because the issuing CA's private key is the sole credential used to sign an organization's certificate, it is the basis of the trust relationship between an organization and an unknown peer. Compromise of this private key allows trivial impersonation of any organization whose certificate has been issued by the compromised organization; there is no way for an entity who has never communicated with a specific organization to know whether or not its issuer's private key (and thus, its signing authority) has been compromised without validating against a list of known compromised certificates. This was the basis of the recent DigiNotar and Comodo attacks, where malicious actors obtained the ability to issue arbitrary certificates signed with the CA's credentials. To mitigate the risk associated with fraudulently-issued certificates, PKI specifies a certificate revocation function that allows entities to determine the validity of an identity claim.

While the concept of certificate revocation is sound in theory, there are several weaknesses in practice.[6] First, all devices must be configured to retrieve the *certificate revocation lists* (CRLs) periodically, and most importantly before any trusted communication with a previously-unknown peer (who might present a fraudulently-issued certificate). This requires connectivity to a CRL server, which is not always feasible – especially in closed or low-bandwidth communications networks, such as AMI or control systems networks. For this reason, CRL checking is not always implemented within embedded PKI implementations. The notion of

[6] Variations of the traditional certificate revocation model, such as OSCP, are not specifically discussed, but have similar, if not identical, weaknesses in large-scale implementation as the CRL model described herein.

using a PKI-based security system therefore does not imply that all aspects of PKI have been implemented. (The trust issues involved with accepting a CRL are outside the scope of this paper.)
Second, CRLs are only updated when a revocation request has been made; that is, after a breach has been detected. This necessitates monitors that will provide real-time updates to some set of CRLs. This in turn requires that a breach of an organization's private key is detected rapidly, added to the CRL, and that this CRL is checked before any trusted communication with that organization – or any entity whose certificates chain through that organization – begins. When a CA does not promptly and publicly disclose the breach of its PKI – as was the case in the DigiNotar breach – the consequences can be severe, as attacks against organizations who have no explicit reason to distrust the now-compromised credentials are possible without detection.

Certificates also typically include expiration dates that invalidate the credentials after a certain time without its entry on the CRL. This, of course, puts the burden on the device that is validating authenticity to inspect the expiry. One implementation weakness with certificate expiration is that organizations frequently issue certificates that are valid for extremely long periods of time: it is not unusual to see certificates with lifespans measured in decades, when the equipment requiring the certificate has no such usable lifespan. In effect, the certificate issuer is saying the certificate will not expire for the life of the device. This is often done to reduce the overhead of reissuing certificates and in some cases deal with an implementation limitation that the credentials cannot be reissued. A second implementation weakness is that the expiration date and other fields are sometimes not checked by the device as part of the certificate validation. In both cases, the failure of an entity to thoroughly check the validity of a certificate allows a malicious actor extended, if not unlimited, time to use compromised credentials.

A breach of a CA's private key – or the issuance of fraudulent certificates signed by the CA – should result in the invalidation of any other chained certificates for which it vouches. Because certificates can be cross-signed by multiple organizations[7], validation of an entity's cross-signed certificate is more complex: even though the invalidation of one of a series of signing certificates in a cross-signed situation does not, in most systems, invalidate the cross-signed certificate, it is up to an organization to determine whether the revocation reduces the trust it should give to the cross-signed certificate. Bugs in the certificate revocation checking process occur as well – some popular systems, for example, fail to check the integrity of certificates when an intermediate wildcard certificate is part of the certificate chain.[8]

A breach of a root CA's private key cannot be mitigated by a CRL update: the entire PKI chain must be assumed to be compromised. A breach involving a root CA would force a redeployment of the trusted root stores through software or firmware updates, which would introduce

---

[7] http://redmondmag.com/Articles/2003/11/01/Cross-Certification-Trusts.aspx?Page=1&p=1, retrieved 10 September 2011

[8] http://www.macworld.com/article/162086/2011/08/mac_os_x_cant_properly_revoke_dodgy_digital_certificates.html, retrieved 10 September 2011

significant delays on low-bandwidth mesh networks such as those found in AMI implementations. The redeployment must also be performed across secure communications channels, which by definition precludes the use of the now-compromised credentials.

Because provisioning of certificates and their related key material must occur using trusted communications channels, the supply chain of the cryptographic key material is of additional concern. If the initial provisioning and deployment of certificates and key pairs is not performed using secure facilities, the chain of trust is compromised from the initial deployment. This is especially significant in AMI deployments, since the provisioning of large numbers of certificates and associated credentials frequently takes place in manufacturing plants and other less-secure facilities, and sometimes occurs in foreign countries by unvetted workers.

3. REAL-WORLD IMPLEMENTATION ISSUES

PKI is provably secure in theory. Implementation, however, is a different matter entirely. Because PKI is complex and requires the correct implementation and execution of multiple services in order to function as designed, it is easy to overlook or deliberately exclude functionality that is required for proper, secure operation of PKI. Embedded devices and heavily-resource constrained networks, in particular, are prone to "short-cuts" and "optimized" implementations of PKI that do not implement PKI components fully or at all. Common examples of PKI component omissions and deleterious optimizations include a lack of CRL checking and mismanagement or insecure storage of private keys or the issuing chain. Separately, each of these implementation flaws erodes the trust of the system as a whole; taken together, the consequences can be severe. Two examples in the electric sector highlight the risks associated with incomplete or insecure deployment of PKI.

3.1. AMI and Smart Grid devices

A common implementation of PKI is in advanced metering infrastructure (AMI), which requires security and privacy to protect utilities and their customers. In many implementations, "smart" meters communicate among each other via a low-bandwidth mesh wireless network, and accept and relay commands from a meter management system (MMS). While the system establishes several certificates and trust relationships, we will concern ourselves with the trust between a meter, the MMS, and the source of firmware. There are at least five attacks against PKI in common AMI implementations that deserve exploration.

Trust between a meter and the MMS is bidirectional: that is, the meter claims an identity to the MMS by producing a signed certificate (with public key) that is provisioned in non-volatile memory. The certificates are signed by the meter or network interface manufacturer; in some cases, these certificates may be chained to a well-established root CA; in other cases, the meter manufacturer implements the root CA for this closed trust system. In either case, the manufacturer acts as a CA for the issuance of meter and MMS certificates; its private key is used as the basis of trust for the identity assertions made via presentation of the certificates.

Because of the cryptographic trust required for the proper functioning of an AMI network (the establishment of secure and private communications channels, for example, to include the application of nonrepudiation and authentication), the integrity of the CA that issues the meter and MMS certificates is a critical function. This, in turn, requires strong protection of the private key used to sign certificates and the systems involved in certificate issuance. This is a non-trivial application of security controls: it is difficult for companies whose core business it is to issue certificates; it is even more difficult for companies, such as meter/network interface manufacturers, to provide these safeguards as operation of a CA is not their core business and may not be well understood.

The effect of a compromise of the manufacturer's private key or signing process can result in the attacker being able to impersonate components of the AMI system. This may include the MMS or the meters entrusted to relay messages within the AMI mesh; in either case, confidentiality, nonrepudiation, and authentication functions can be compromised, resulting in disclosure of customer data or the application of fraudulent control commands, such as remote premise disconnection. An attacker who succeeds in obtaining a fraudulent MMS certificate may now impersonate the MMS to every meter configured to accept configuration, remote commands, and firmware updates from that system.

The second attack against AMI systems involves the compromise of insecure private keys and root certificates used by the meters to establish secure communications channels. In this scenario, the attacker does not need to impersonate a meter – he already has access to it – but he may now subvert the meter's private key material to intercept and/or change communications protected by that key. In common implementations, this private key material is often stored in ordinary non-volatile memory as clear-text. A scan of this memory can reveal the private key to someone who knows what to look for. An attacker with physical access to the device (or one who can extract the contents of memory remotely) may now impersonate the meter. This allows the attacker to spoof the identity of the meter for all functions for which trust is required: both meter-to-meter communications requiring digital signatures, as well as meter-to-MMS communications, may be intercepted and issued by a rogue device. A variety of techniques, such as obfuscating or encrypting the private key in memory are also used. However, obfuscation is not provably secure, and encrypting the key only moves the burden of secure storage to the key used to encrypt it. In other implementations, special hardware is used to control access to the device credentials and destroy the key if indications of tampering are detected. This type of security is typically associated with Smart Cards, which are used in some instances for electronic payment transactions. Systems using these safeguards are typically involved in high-value transactions and are highly targeted because of the potential benefit of a successful compromise. Consequently, they are carefully designed and tested with many security defenses. The primary argument against deploying secure hardware within AMI and Smart Grid networks is the cost of the secure components and the need to keep unit production costs low: when a device manufacturer calculates the total cost to product millions of devices with this

extra cost, the economics may suggest using less secure methods, including attempts to design custom, unproven secure hardware.

A related attack exploits weaknesses in the processes that use device credentials. Secure storage satisfies many security needs, but the private key still needs to be used in cryptographic operations. In insecure implementations, a securely stored private key may be decoded and then stored in RAM for convenience or performance optimization. A malicious actor who is able to read the device memory (through any one of a number of attack vectors, including debugging interface attacks and programmatically) will be able to retrieve the key. Yet another insecurity occurs when the key material is passed outside the secure hardware over insecure ports and buses. A logic analyzer or port sniffer can be used to recover the unprotected key. Any interface that device credentials pass through needs to be secure.

A fourth class of attack against PKI security credentials in devices is known as a *side-channel attack*. Side channels refer to indirect ways to observe behavior and extract information through unintended leaks. Two primary means by which side channel attacks are performed are by monitoring the devices power system and electromagnetic emissions. Every time a processor executes an instruction, it uses different elements: in one case, it may be memory; in another, an arithmetic logic unit. Each of these elements uses power and creates tiny fluctuations in the power drawn by the system. Using a technique known as *differential power analysis* (DPA), an malicious actor can recover device credentials by monitoring fluctuations in power use unless special hardware counter measures are not built into the system. Such countermeasures are typically not incorporated in consumer grade electronic components. DPA, in most cases, requires physical access to the device. However, because customer AMI devices are typically not physically well protected, access to a device (such as a residential meter) is usually not difficult.

A more insidious side channel attack involves monitoring the electromagnetic emissions created by the processor while it is performing a cryptographic operation. Private keys can be recovered successfully using this method without direct physical access to a device.

A fifth form of attack is to re-align device trust. Similar to its private key, the root certificates on a meter (within the trusted root store) must be protected. It is to a root certificate that the device ultimately chains all authenticity validation checks. If a malicious actor were able to change the root certificate in a device, the device would no longer have a trust chain to its intended root CA. Instead, the trust chain could be manipulated to force the device to trust the entity installed by the malicious actor.

Finally, we examine the process of certificate revocation, which is the primary method to mitigate the risk of compromised key material. In several embedded device implementations (including some AMI systems), certificate revocation is not performed properly, if at all. In some cases, there is no way for a meter to determine whether or not a certificate being presented to it is still valid; it will check the signature via a public key stored in FLASH (or, in some cases, in

the firmware image itself) but has no timely way (if one exists at all) to check for revocation. Since many certificates' lifespans are measured in decades, the lack of an effective certificate revocation service means that a compromise of a certificate will go undetected by the devices. If a device does process CRLs, a related attack is to modify the CRL to remove some or all certificates so that it will respond to compromised certificates as if they were valid.

More significantly, if the AMI system uses a self-signed root certificate issued by the vendor or manufacturer, there is no practical way to recover from a breach of this root CA: the self-signed root certificates that are located in a device's trusted root store must be removed manually. This typically requires a firmware update or a software rebuild of the device. Since firmware updates are typically signed by the vendor or manufacturer, the organization implementing the AMI network will find itself in a catch-22 situation: it must rely on the claim of identity provided by an entity that is known to have been compromised.

3.2. Control Systems and SCADA Networks

Networks of industrial control systems (ICS) are typically isolated from other corporate networks and the Internet. This isolation provides a good layer of defense against the propagation of malicious code; however, in many cases it also hinders the timely application of patches and updates which are often necessary to change system-wide trust relationships. From a PKI perspective there are several issues to consider.

PKI attacks against ICS can leverage the fact that the chain of trust cannot be guaranteed in an isolated environment when that chain relies on a root CA whose CRLs are not on the local network. That is, systems have no easy way of checking an updated CRL in a network that has no external connectivity. Updates to the CRLs typically come via a patching process, which in ICS networks is sometimes delayed due to the extended testing of patches required to ensure that their application does not disrupt operations of critical systems. It is therefore likely that a malicious actor using a compromised issuing certificate / private key from a CA that is no longer publicly trusted could, even months after the disclosure of the breaches, forge credentials that would be accepted by many ICS systems, merely because they have yet to receive the patches that would invalidate the breached companies' certificates.

The same general concerns relating to the compromise of a self-signed root certificate apply to ICS networks: there is no easy way to recover from a breach of a root CA. In the case of control systems networks, which are typically much smaller than AMI networks, the trusted root stores, located in firmware and software, might be updatable by a technician. For a vendor with thousands of customers worldwide, however, such a scenario is not practical and approaches the level of complexity that is described for AMI networks.

In general, industrial control systems are vulnerable to the similar types of attacks on PKI and implementation weaknesses as Smart Grid AMI systems. Industrial control systems may have the added benefits of better protected physically and their electronics built to a higher total unit

cost than mass-produced AMI devices; this implies that the addition of hardened security modules and associated hardware will have less influence on the overall manufacturing cost of the ICS hardware relative to AMI installations.

4. MITIGATION PROCESSES

Technical mitigations for deficiencies in PKI implementation fall to the manufacturers and maintainers of the equipment. In most cases, the end users of the equipment (utilities and other operators of this infrastructure) are unable to implement the missing or incorrect PKI functionality in the products without violating warranties or risking operational disruption.

Utilities, vendors, and others responsible for operating AMI and control systems must understand the role of certificates in the AMI implementation, and must be aware of what systems and what individuals have access to the private keys used to establish trusted identities among the critical AMI or control systems components. These organizations must be cognizant of the protections afforded the PKI cryptographic key material, and of the protections available to the systems and personnel involved in the generation of certificates that are used as the basis of trust for AMI and control systems network communications. As an example of policy requirements surrounding the protection of PKI components, the Federal Bridge Certification Authority Product Interoperability Guidelines, a document specifying the requirements for CAs to participate in the US Government PKI, mandates the use of FIPS 140-1 level 3 hardware protection for CA private keys.[9, 10]

A common method to mitigate the effect of PKI compromises is to segment the PKI system. While this method involves increased administrative cost – multiple root keys and certificates must be separately and securely managed – the keys can be independently protected and a compromise of one does not necessarily imply the compromise of the remaining credentials / devices. This is particularly applicable in AMI environments, where a large utility may have several million endpoints. A single PKI hierarchy implies that different utilities' trust chains eventually lead to a common CA at the manufacturer or vendor. A compromise at the manufacturer or higher up the chain affects all utilities, whereas a segmented system may limit the impact to a subset of customers. Furthermore, there is the problem posed when an equipment manufacturer or vendor operating part of a CA chain ceases operation or otherwise goes out of business. In such an event, the difficult situation arises of determining which surviving organization(s) will be responsible for a CA that supports multiple customers. In the segmented model, each customer can decide to redirect its own part of the CA chain that was previously chained to the now-defunct organization.
Protection of the root CA credentials is vitally important. Securing this material is a complex process that requires extreme attention to detail in technology, policy, and personnel decisions.

---

[9] http://www.idmanagement.gov/fpkima/documents/product_guidance_rev9-26.pdf, retrieved 20 November 2011

[10] http://www.idmanagement.gov/fpkipa/documents/CommonPolicy.pdf, retrieved 20 November 2011

For example, no single person should have access to these credentials. A number of published methods for the secure storage of CA credentials exist; there are different approaches that can balance confidentiality and availability, such as the "M of N" model.[11]

5. CONCLUSION

PKI is difficult to implement securely given the complexity of its interrelated functions, but it can be done by organizations demonstrating a commitment to providing and maintaining a secure infrastructure. A wealth of documentation describing proper PKI implementation and best practices exists in the public domain; please see the "Additional Resources / Further Reading" section below for links.

Operators of AMI and control systems infrastructure should ask the suppliers of the equipment to describe, in detail, how specific pieces of the PKI system are implemented. Suggested questions include the following (questions marked with asterisks denote analyses that must be performed for every type of device that stores private key or certificate material, not just for select systems. For example, an analysis that focuses on the certificates stored at a Meter Management System would be incomplete in the absence of any analysis of the AMI meters themselves.):

Operations

- Has the CA published its Certification Practices Statements (CPS) and its Certificate Policy (CP) detailing how it operates its CA?
- Has the CA been externally certified and vetted by a third party?

Key Material

- Where (geographically) are the keys provisioned?*
- Who provisions the key material?* Is there a vetting / personnel assurance process for these individuals? To whom are they accountable?
- How is private key material protected, physically and logically?*
- What policies exist surrounding the creation, storage, and maintenance of the keys?*
- Which keys are common across multiple customers?*

Certificate chaining and handling

- Is there an identity proofing process used to establish the root of trust prior to credential issuance?
- Who issues the certificates?
- What is the designated lifespan of the certificates?*
- What organization maintains the root CA?
- What policies exist surrounding the operation and maintenance of the certificate authorities (both intermediate and root)?
- Which certificates are common across multiple customers?*

---

[11] http://www.certiguide.com/secplus/cg_sp_4571MofNControl.htm, retrieved 2 December 2011

- What certificate extensions, if any, are in use?
- What certificate attributes are used for authentication and/or authorization?*

Breach / Incident Response

- In the event of business shutdown, how will the keys be maintained or distributed to the customers?
- How will customers be notified of a breach of key material or a compromise of an issuing CA?
- How is certificate revocation performed?* Who is responsible for updating CRLs if they are used?
- What is the process for updating the trusted root stores if a breach of the root CA occurs?*
- How are compromised / invalid keys reissued?*
- What is the impact of the compromise of private key material?*
- Does the PKI architecture support Online Certificate Status Protocol (OCSP)?

6. ADDITIONAL RESOURCES / FURTHER READING

In addition to the documents referenced in the footnotes, the following is a short list of recommended reading:

Isode Ltd.: "A Short Tutorial on Distributed PKI", via http://www.isode.com/whitepapers/dist-pki-tutorial.html, retrieved 20 November 2011

Gutmann, Peter: "Everything you Never Wanted to Know about PKI but were Forced to Find Out", via http://www.cs.auckland.ac.nz/~pgut001/pubs/pkitutorial.pdf, retrieved 2 December 2011

ArcticSoft Technologies Ltd: "What is PKI (Public Key Infrastructure)?", via http://www.articsoft.com/public_key_infrastructure.htm, retrieved 29 November 2011

Kuhn, D. Richard, *et al.*, National Institute of Standards and Technology, "SP 800-32: Introduction to Public Key Technology and the Federal PKI Infrastructure", via http://www.gpo.gov/pdfs/authentication/sp800-32.pdf, retrieved 30 October 2011

SANS, Inc: "End User Encryption Key Protection Policy", via http://www.sans.edu/student-files/projects/200908_02.pdf, retrieved 2 December 2011