Global Information Assurance Certification Paper

Analyzing Enterprise PKI Deployments

GIAC (GSNA) Gold Certification

Author: Walter Goulet, wgoulet@gmail.com

Advisor: Rob VandenBrink

Accepted: August 4th 2009

Abstract

PKI solutions are frequently deployed in enterprise environments to solve various security problems (improving WLAN security with 802.1x, securing internal & external websites, signing code and sensitive documents etc). Due to the inherent complexity in PKI solutions, many enterprises have difficulty using and deploying them correctly. As such, any security assessment targeted to an enterprise environment should include an analysis of PKI solutions in use. This paper will describe how PKI solutions can be assessed as part of an overall security assessment program.

1. Introduction

Public Key Infrastructure (PKI) has long been used as a solution to provide the following assurances to enterprises (Kuhn, Hu, Polk, & Chang, 2001):

- The person or process identified as sending the transaction is actually the originator.
- The person or process receiving the transaction is the intended recipient.
- Data integrity has not been compromised.

The ability to provide these assurances to enterprise owners has resulted in an explosion in the popularity of PKI as an authentication solution for many different systems that are deployed in a typical enterprise network environment. However, correct deployment of PKI solutions is crucial to ensuring that the security assurances are in fact in place. An article published in the Computer Security Journal identifies 10 risks that an organization may be exposed to due to inadequate deployments of PKI solutions, with perhaps the most important risk being "Risk #1: "Who do we trust, and for what?" (Ellison & Schneier, 2000). The significance of this risk is key, as one of the most fundamental jobs of a PKI solution is to provide postive assurance that both parties in a communication channel have accurately identified each other. Therefore, it is critical that any security assessment that is undertaken by an enterprise pay serious attention to any PKI deployments that are used as part of the enterprise's business operations.

Assessing an enterprise PKI deployment involves a combination of assessing policies that govern the operation of a PKI such as a Certificate Policy (CP) and a Certificate Practice Statement (CPS) as well as observing that the policies/practices are being effectively implemented. An assessment should also evaluate the X.509 certificates that are issued by a Certificate Authority operating a PKI to ensure that best practice guidelines are being followed. One of the most authoritative methods for assessing the correctness of a PKI deployment is the PKI Assessment Guidelines, published by the American Bar Association which defines a complete framework for assessing a PKI

deployment for both technical and operational correctness (Information Security Committee, American Bar Association, 2001).

This paper will introduce the motivation for public key infrastructure, briefly introduce PKI concepts, identify common use cases for PKI deployments, define a set of X.509 digital certificate profiles as well as appropriate certificate policy/certificate practice statement outlines for each use case, and present a lightweight assessment technique partially adopted from the PKI Assessment Guidelines (Information Security Committee, American Bar Association, 2001). In addition to the assessment technique, a set of automated checklists and a script to perform the assessment will be presented.

2. Why Public Key Infrastructure?

In many enterprise environments, sensitive information is exchanged between users within the enterprise. This information may include sensitive data such as company intellectual property, employee personal information, and financial data. In order for this information to be exchanged safely, the communication system must provide the following services:

- Provide assurance that users within an enterprise are communicating only with intended recipients.
- Provide assurance that messages exchanged between users cannot be observed by other users.
- Provide assurance that messages between users are not be modified while in transit.

In addition to secure communications, enterprises may also have requirements to be able to certify that electronic documents are authentic and have not been modified since they were originally created.

Finally, enterprise network environments typically provide access using wireless and remote access technologies. These types of environments do not provide the same level of security and control that a wired network environment provides. Therefore, users

that connect to the enterprise need some assurances that they are connecting to the correct network while the network needs some assurance that only valid users are permitted access to the network.

Many different security solutions exist to solve the problems above. However, all of these security services require users to be assigned a credential or key. In a typical enterprise environment where there may be thousands of employees, it is not feasible to exchange keys with all other enterprise users. Public key infrastructure provides a solution to this problem by enabling the enterprise to designate a single entity that is responsible for issuing credentials to users that can be used for a variety of different security services.

3. PKI Overview

In order to understand the use cases for enterprise PKI deployments and how to assess them, it is necessary to understand the basics of how PKI solutions work and the reasons for enterprises to deploy them. This section will cover the basics of public key cryptography, the problems solved by PKI deployments, and a technical description of the components of a PKI deployment.

For a more thorough discussion of X.509 digital certificates and PKI in general, the user is encouraged to refer to RFC3280 as well as (Kuhn, Hu, Polk, & Chang, 2001).

3.1. Cryptography Basics

Cryptography is the art of securely exchanging information that is to be kept secret over an open, public communication channel. Cryptographic systems provide the following services to users: confidentiality (to keep the exchanged information secret), authentication (to ensure that the users of the system are who they claim to be), and integrity (to verify that the information exchanged between users has not been modified in transmission). These systems make use of several cryptographic algorithms which are combined to provide these services.

Encryption algorithms are reversible functions which take as input data that is to be encrypted as well as a secret key. The output of an encryption function is undecipherable data that appears to bear no relation to the input data. Encryption algorithms are reversible in that the output data and the secret key can be run through the encryption function again to recover the original input data.

Hash functions are irreversible functions which take as input data that is to be protected and output a fixed size chunk of data that uniquely represents the input data. A key property of hash functions is that two different sets of input data must have different outputs. Another key property of hash functions is that any change to the input data, even a single character, should yield a completely different hash output.

Encryption and hash functions can be combined to provide additional security services such as integrity checking. For example, say two users wish to send a message to each other that doesn't need to be kept secret, but that cannot be modified while it is in transit. Further assume that both users have access to an encryption key and are using the same hash and encryption algorithms. User A can input the message into a hash function to generate a small block of text. User A then uses an encryption algorithm and the secret key to encrypt the hash output. User A then sends the message along with the encrypted hash output to User B. User B first inputs the message into the hash function to obtain a hash of the message. User B then uses the secret key and encryption algorithm to decrypt the hash output he received from User A. The decrypted hash output is then compared to the hash output he obtained earlier. If the outputs are identical, User B knows that the message was not modified while it was in transit. This particular use of hash and encryption algorithms is referred to as a digital signature algorithm.

As noted above, encryption algorithms make use of a secret key to encrypt and decrypt information. There are two different types of encryption algorithms; algorithms which use the same key to encrypt and decrypt information (symmetric encryption) and algorithms which use different keys to encryption and decrypt information (asymmetric encryption). There are important advantages and disadvantages to both types of algorithms as discussed in the next section.

Walter Goulet, wgoulet@gmail.com

3.2. Public Key Cryptography

Cryptosystems which make use of symmetric encryption algorithms require that all users have access to the same secret key. In some scenarios such as tightly controlled environments where the network is controlled and all users are known, this is not a problem because it is relatively easy to distribute the secret key to all users. However, in other environments such as public networks where the users cannot exchange the key securely, this is a significant problem. Since there is only 1 secret key used, if the key were to be captured by an attacker, the attacker would be able to encrypt and decrypt all information that is exchanged in the network.

Public key cryptography solves this problem by making use of asymmetric encryption algorithms combined with message authentication code (MAC) functions. In this type of system, one of the keys is kept secret and never disclosed by the owner (private key). The other key is shared with other users (public key). Information that has been encrypted with the public key can only be decrypted with the corresponding private key. This relationship between the keys means that the public key has little value to an attacker since the public key cannot be used to decrypt messages that are exchanged between users. Therefore, the public key can be distributed to other users over any network.

3.3. X.509 Digital Certificates

X.509 digital certificates are the most prevalent public key cryptography solution in modern use. X.509 digital certificates contain a user's public key along with identifiers that indicate which public key cryptography algorithm and digital signature algorithms were used to generate the certificate. Users that are issued X.509 digital certificates have a private key stored securely on their system that corresponds to the public key contained in their X.509 digital certificate.

In addition to containing a user's public key, a X.509 digital certificate contains additional identifying information about the user. This information varies depending on the type of entity that the certificate belongs to. For example, a certificate which is used

to secure email messages would contain the email address of the certificate owner along with the name of the organization that issued the certificate. Recipients of the user's X.509 digital certificate can then use this identifying information to associate the message with the user.

Finally, a recipient of a X.509 digital certificate must be certain that the certificate contents have not been modified after the certificate was generated. This assurance is provided by a digital signature that is included in the X.509 digital certificate. The digital signature is generated by a digital signature algorithm which takes as input the contents of the X.509 digital certificate (identifying information, user public key and other fields) along with a private key. The digital signature is then appended to a X.509 certificate. This signature can be used to validate the contents of the certificate as long as the recipient of the message has the public key that corresponds to the private key used to generate the digital signature.

3.3.1. Self Signed vs. Non Self Signed X.509 Certificates

As mentioned in the previous section, recipients of a X.509 digital certificate use the digital signature embedded in the certificate to verify that the certificate contents have not been modified. Recall that the digital signature is created using a private key. If this private key belongs to the user that owns the X.509 digital certificate, the certificate is known as a self-signed digital certificate. If the private key used to create the digital signature belongs to another user, the certificate is considered to be a signed certificate.

Note that this distinction has serious ramifications on the overall trustworthiness of the X.509 digital certificate. If the certificate is self-signed, the recipient of the certificate has to trust the certificate contents at 'face value' since the entity which owns the certificate is vouching that it is a valid certificate. If the certificate is signed by another party, the recipient has assurance that another entity has signed the certificate and is essentially vouching for the contents of the certificate. The entity that vouches for the contents of a certificate by signing it is known as a Certificate Authority (CA). The process of verifying the identity of a certificate owner and digitally signing the certificate is known as issuing a certificate.

3.3.2. Anatomy of a X.509 Certificate

As discussed earlier, a X.509 digital certificate contains information that identifies the certificate owner as well as information that can be used to verify that the certificate is valid. Certificates are split into 3 sections

- Certificate fields which contain information that identifies the certificate owner, the issuer of the certificate, and other information used to validate the certificate.
- Certificate extensions which contain additional information typically used to indicate what purposes the certificate is valid for. Additional identifying information and revocation information may also be contained in the extensions.
- A signature which is a digital signature of the certificate contents. This signature is used to validate that the certificate has not been modified since it was generated and, in the case of signed certificates, can verify that the certificate is issued by a trusted CA.

The following table explains the contents of a digital certificate in more detail. This table is not exhaustive and shows only the most significant portions of a X.509 digital certificate; refer to RFC3280 for more details.

Certificate Section	Name	Notes
Certificate Field	startDate	This field, along with the endDate field, is used to indicate the time period that the certificate is valid for.
	endDate	
	signatureAlgorithm	This field indicates which digital signature algorithm was used by the entity which digitally signed (issued) the certificate.
	Version	There are 3 versions of the X.509 standard. This field indicates which version the certificate conforms to.

	Subject	This field contains a collection of information that identifies the owner of the certificate. The field contains a distinguished name which is further split into sub fields that contain details like the actual name of the entity, an email address, organization, address etc.
	Issuer	This field contains a collection of information that identifies the entity that signed the certificate. If this field is identical to the subject field, the certificate is self-signed. If the field is not identical to the subject field, then a third party has signed the certificate.
	Public Key Info	This field contains the subject's public key as well as the name of the public key algorithm that was used to generate the key.
Certificate Extension	[multiple extension names may be listed]	The certificate extension field contains a collection of extensions which, among other uses, indicate the valid uses for the certificate (e.g. a digitalSignature extension means the certificate can be used to digitally sign objects). Other common extension fields are used to publish additional information about the certificate owner and certificate issuer.
Signature	[contains binary encoded digital signature]	The signature section contains the digital signature created as a result of the certificate issuer signing the certificate. The signature indicates that the certificate fields and

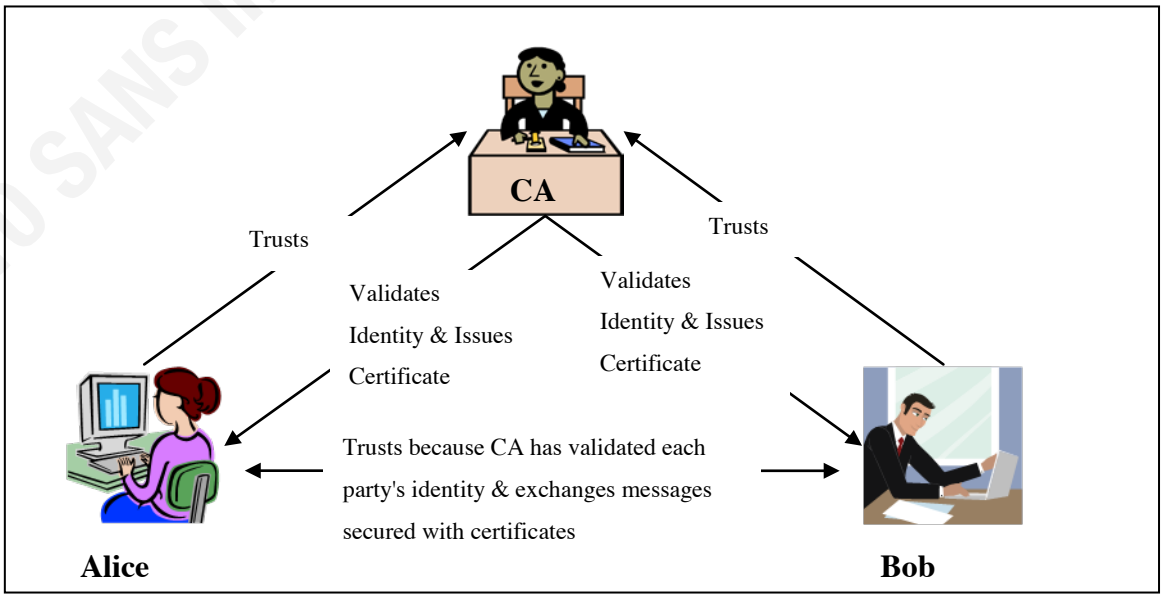
		extensions have not been modified since the certificate was generated. From a trust perspective, the act of a certificate issuer signing a digital certificate indicates that the issuer has verified the certificate owner's identity and has verified that the certificate owner has a private key corresponding to the public key included in the certificate Public Key Info field .

3.4. Introduction to Public Key Infrastructure

As discussed in the previous section, in order for X.509 digital certificates to be generally useful it is necessary to establish a CA. The CA will have responsibility for validating the identity of users that wish to obtain X.509 digital certificates as well as issuing the certificates to users.

The supporting infrastructure that provides this functionality is known as a Public Key Infrastructure (PKI). The following figure shows a basic conceptual model of a PKI.

Figure 1 Basic PKI Model



In this model, the CA is universally trusted by all users of the PKI. Since the CA is a known trusted entity, end users will trust certificates that have been issued by the CA.

3.4.1. Establishing Trust

As noted in Section 3.3, the CA issues certificates by signing certificate requests with its private key. In order for users to indicate that they 'trust' the CA, all end users of the PKI must obtain a copy of the CA's public key. The CA's public key is known as a root certificate. The root certificate is self-signed by necessity (there is no other entity that can independently validate the CA), therefore it is distributed to users by other entities that they trust (operating systems and web browsers are pre-configured with a list of trusted root certificates, for enterprise PKI deployments the root certificate is typically installed by the enterprise IT department).

3.4.2. Validating Identity

In order for the CA to actually be able to vouch for the identity of the users who wish to obtain digital certificates, the CA must perform validation of the user identity. Typically this is done by verifying that the user requesting the certificate is an authorized user of the PKI (for an enterprise PKI, this would likely consist of verifying that the user is a current employee of the organization).

3.4.3. Issuing & Revoking Certificates

Once the CA has verified the identity of the user requesting a certificate, the CA uses the root certificate private key to sign the certificate request which produces a X.509 digital certificate. The digital certificate is then sent to the user at which point it can be stored in the user's PKI client. Additionally, the CA may publish the user's X.509 digital certificates in a central repository for other PKI users to retrieve them as necessary.

In some cases, it is necessary for a CA to revoke a digital certificate. This could be because the user of a digital certificate has lost their private key or it has otherwise been compromised. In this case, the CA publishes a list of certificates that have been revoked in a certificate revocation list (CRL). These CRLs are periodically distributed to users to ensure that they are notified as soon as possible when a certificate is compromised.

3.5. Services Provided by a PKI

Fundamentally, a PKI provides its users with a validated credential along with a set of keys that can be used to assert the identity of other end users. Additional cryptographic algorithms can take advantage of the key material to provide even more advanced security functions. For example, key negotiation algorithms can use the material in X.509 digital certificates to create a shared secret key that can be used for other cryptographic operations. Another common use of X.509 digital certificates is to create digital signatures for electronic documents, code objects, and other artifacts.

When choosing whether or not to deploy a PKI, an organization should consider the following questions which may indicate that a PKI is an appropriate solution:

- Does the organization have a requirement to securely exchange messages and electronic documents with other organizations?
- Does the organization permit guest users to access network resources or does it wish to deploy wireless networks?
- Does the organization need to provide access to sensitive applications and / or data to certain users of the organization?

If the answer to any of the above questions is 'yes', then a PKI deployment can help provide these services. While this list is not exhaustive, it illustrates just a few of the services that can be provided by a PKI.

3.6. PKI Technical Model

A PKI implementation consists of several components connected in a client/server scheme over a network. The components are:

- A CA server - this system issues X.509 certificates to clients and generates CRLs. The CA also hosts the root certificate and private key for the PKI.
- PKI clients - software on systems that make use of X.509 certificates to provide security services to users. PKI clients are embedded in web browsers, email

clients, WLAN supplicants, and network infrastructure such as routers/switches to name a few examples.

- A revocation server - this system hosts CRLs issued by the CA or optionally acts as an online certificate status protocol (OCSP) server. PKI clients use CRLs and OCSP requests/responses to validate that a X.509 certificate has not been revoked by the CA.
- A Registration Authority (RA) server - this system accepts requests for new X.509 certificates from PKI clients and validates the identity of the requestor before submitting the request to the CA server. Note that the RA/CA may be collocated on the same server

For a more complete description of the technical PKI model, readers are encouraged to refer to Leslie Peckham's analysis of enterprise PKI deployments (Peckham, 2003).

In addition to the technical components of the PKI, there are supporting policies and procedures which govern the operation of the PKI.

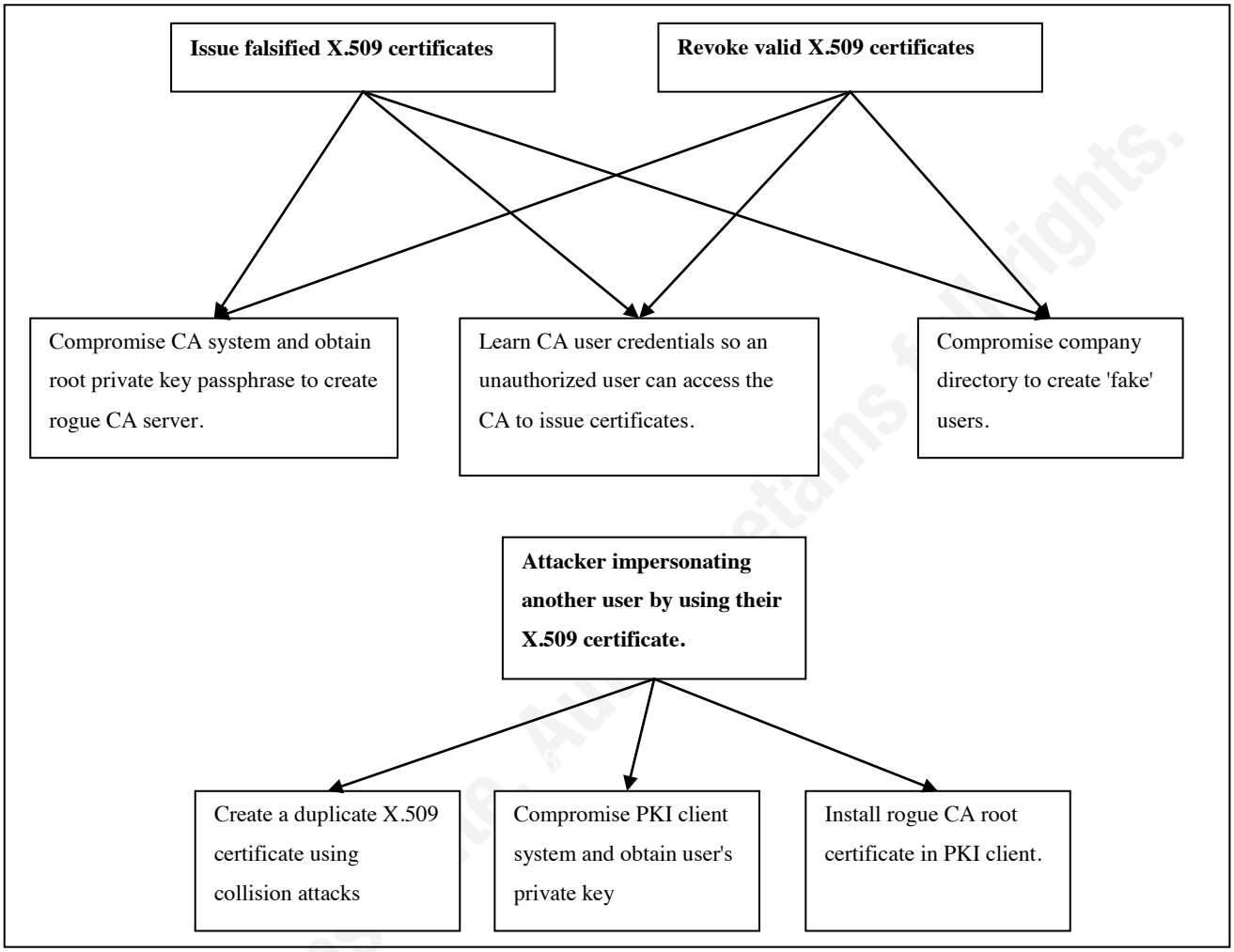
- Certificate Policy - This policy document describes the overall purpose of the PKI and specifies the requirements that govern its operation.
- Certificate Practices Statement - This document specifies the technical, operational, and management practices that implement the requirements laid forth in the certificate policy.

3.7. PKI Best Practices

To define the best practices for operating a PKI, it is necessary to first define a basic threat model which identifies the most common security threats that are applicable

Figure 2 PKI Deployment Attack Trees

for an enterprise PKI deployment. The following attack tree illustrates these threats.



As the attack tree illustrates, the 3 major threats that need to be mitigated by the enterprise PKI deployment are: issuing false X.509 certificates, revoking valid X.509 digital certificates, and impersonating a valid user. The attacks that can be performed to realize these threats can be countered by ensuring that the CA components are protected in accordance with server security best practices, that the processes used by the CA administrator and RA are complete, and by ensuring that the algorithms and key lengths used in the issued X.509 certificates are sufficiently complex. The following table lists out PKI best practices that can effectively mitigate these attacks.

Attack Scenario	Mitigating Best Practice
Compromise CA system and obtain root private key passphrase to create rogue CA	Ensure the CA server operating system and physical environment is protected using security

server.	best practices and that a CP/CPS is developed and followed by the organization (can be validated via the process defined in 5.1.1)
Learn CA user credentials so an unauthorized user can access the CA to issue certificates.	Ensure the CA server is protected using security best practices and that a CP/CPS is developed and followed by the organization (can be validated via the process defined in 5.1.1)
Compromise company directory to create 'fake' users.	Ensure that the validation process used to validate the identity of users/entities that are issued certificates does not rely solely on published electronic directories (e.g. verify identity using phone calls etc) and that this process is documented in the CP/CPS.
Create a duplicate X.509 certificate using collision attacks	Ensure that sufficiently strong key lengths are used in end user certificates as well as CA root certificates (discussed in detail in the individual certificate profiles defined below).
Compromise PKI client system and obtain user's private key	Ensure that all client systems which use X.509 certificates are hardened in accordance with security best practices.
Install rogue CA root certificate in PKI client.	Ensure that the CP/CPS defines the process used to issue root certificates to end users and that this process is followed rigorously (can be validated via the process defined in 5.1.1)

The assessment process defined in this paper, the CP/CPS outline, and the various certificate profiles permit assessors to validate that a particular enterprise PKI satisfies these best practices.

4. Enterprise PKI Deployment Use Cases and Certificate Profiles

The security features of PKI are useful in several common use cases in enterprise network environments. The following sections examine each of the PKI use cases in detail as well as describe how certificates are used in each use case. Each section also

includes a certificate profile which identifies what the contents of certificates used in each type of PKI deployment should be.

4.1. S/MIME

Email is one of the original applications that enjoyed widespread use as the Internet evolved, both for personal use and for enterprise sue. An important requirement for some enterprise email users is the ability to be certain that email messages are authentic (ensuring that the author of an email is who they claim to be) and that they have not been modified in transit.

S/MIME (Secure/Multi-Purpose Internet Mail Extensions) solves this problem by providing a mechanism to digitally sign the contents of an email with a X.509 certificate that is issued to users by a trusted CA. The S/MIME RFC (RFC 3851) provides this service by defining several extensions that can be added to email messages by email clients which contain information such as digital signatures, identifying information about the author which can be used by clients to choose the proper certificate to authenticate digital signatures, and algorithms used in digital signatures. In addition to providing signatures, S/MIME also permits parties to encrypt email messages. The selection of which security service is to be used for messages exchanged between users is determined by the email application.

4.1.1. S/MIME Certificate Profile

RFC 4262 defines the X.509 certificate profile that is used by S/MIME clients. The RFC lists out which extensions *must* be supported by RFC 4262 compliant S/MIME clients as well as which certificates should be supported. Note that the RFC doesn't always present the rationale behind why a particular certificate field *should* be included in a certificate. For those fields for which no rationale is given, the field will only be included in the profile if the field improves the overall security of the S/MIME protected messages.

Note that the profile above is considered to be a minimum definition of a well formed S/MIME X.509 certificate. Practical experience with S/MIME certificates used

by different email clients has shown that other extensions that are specific to certain clients are frequently included in X.509 certificates. For the purposes of this paper, these extensions may be ignored.

Table 1 S/MIME Certificate Profile

	Name	Required Contents	Properties	Rationale
Certificate Field	startDate	UTCtime or Generalized Time stating when the certificate will become valid.		The validity field of the X.509 certificate is used by the CA to indicate when the certificate is considered 'valid' and may be used for its intended purpose. Auditors should be careful to understand circumstances under which it is permissible to issue certificates that are not yet or are no longer valid.
	endDate	UTCtime or Generalized Time stating when the certificate will no longer be considered valid.		
	signatureAlgorithm	sha1WithRSAEncryption		RFC 3850 specifies support for both md5WithRSAEncryption and md2WithRSAEncryption. However, the MD2 (The Cryptix Foundation Limited and David Hopwood, 2002) and MD5 (Dougherty, 2009) algorithms are subject to collision attacks which may permit an attacker to create a certificate that could be trusted by a S/MIME client. Therefore, only sha1WithRSAEncryption should be used as a signature algorithm. NIST SP800-57 part 1 provides guidance that after 2010, SHA1 should be deprecated in favor of the SHA2 variant of signature algorithms. However, this will not be a standard option in S/MIME clients until RFC 3370 is updated to support the stronger signature algorithms.
	Version	3 (0x2)		The X.509 v1 and v2 certificate formats are considered obsolete and should not be used in production systems.

	Subject	Non-blank value unless the subjectAltName field is populated with a valid email address		
	Issuer	Non-blank value that is different than the value in the subject field		The issuer field indicates which CA issued the certificate. As a best practice for S/MIME systems, self-signed certificates should not be used as they do not provide any useful security assurances to the users of the system (e.g. identity of the users cannot be authenticated using self-signed certificates).
	Public Key Algorithm	rsaEncryption id-dsa id-ecPublicKey	For the RSA public key algorithm, a key length of 1024 or greater bits is recommended.	From pp 158 – 164 of *Applied Cryptography* (Schneier, 1996), public key lengths of 1024 bits were predicted to be sufficient for protecting information against attacks by individuals. Current RSA recommendations for RSA public key algorithms are 1024 bits for corporate use (RSA Security, 2009). NIST SP800-57 part 1 indicates that for government applications, a key length of 1024 is sufficient through the year 2010, with key lengths of 2048 bits sufficient for use through the year 2030 (Barker, Barker, Burr, & Smid, 2007). Other algorithms such as DSA and ecPublicKey (elliptic curve public key algorithm) are also acceptable; although at the time of writing the ecPublicKey is not widely supported by S/MIME clients.
Certificate Extension	subjectAltName	Email address of the user sending a signed S/MIME message to a recipient	Must be marked as a Critical extension if the subjectAltName	If the 'From' field of a S/MIMEv3 signed message matches the subjectAltName extension in the certificate used to sign the message, the recipient of the message can be sure that the message is genuine (has been sent by the owner of the email address and has not been modified in transit).

	cRLDistributionPoints	A non-blank distribution point name field.	Not critical	RFC 3850 requires that all S/MIME clients perform certificate revocation checks. Since the main reason for using S/MIME when exchanging emails is to assure the recipient of a message that the message is genuine, it is imperative to ensure that the CA has the ability to revoke certificates so S/MIME clients can detect invalid certificates.
	keyUsage	nonRepudiation = true digitalSignature = true	Critical	These 2 certificate usage extensions indicate that the certificate is used to create digital signatures. If these certificates are NOT included in a S/MIME client certificate, it is not a violation of the RFC. However as a best practice these bits should always be included.
	extendedKeyUsageExtension	emailProtection OR anyExtendedKeyUsage		The emailProtection field MUST be included if the S/MIME client is not permitted to use the certificate to encrypt messages. If the S/MIME client is permitted to use the certificate to encrypt messages, then the anyExtendedKeyUsage OID must be included.

4.2. HTTPS protected Intranet Websites

Many applications in the enterprise environment are hosted via intranet websites. This is due to the explosion in the popularity of the web browser as an application platform. However, web applications do pose additional risks to end users. Web applications, unlike desktop applications, can easily be spoofed by simply recreating a website that is hosted on another web server. For example, a common Intranet hosted web application is web based email (typically deployed as part of the enterprise email system as a backup for desktop based email clients). If a malicious attacker wishes to obtain the email credentials for another user, an attacker could 'clone' the login screen for the web application and host this application on another website. Or, the user could attempt to spoof the intranet DNS server to point the web email domain name to another web server that has been setup for malicious purposes. Legitimate users would not be able to distinguish between the spoofed web email server and the actual web email server. However, if the real web email server is protected via a secure sockets layer (SSL) certificate, the attacker would not be able to spoof the real web email server as the user's web browser would display an error message while validating the spoofed web server's certificate (or, if no certificate is used on the spoofed web server, the user would notice that the browser 'lock' icon is not present).

SSL (or more correctly, transport layer security (TLS)), is the standard mechanism to secure websites by providing a way to assure the user that the website being visited is authentic and to encrypt traffic between the user and the website. These features are provided through the use of X.509 certificates which are installed in the web server and contain information that identifies the website, specifically the Fully Qualified Domain Name (FQDN), or a pattern that matches a FQDN. The certificate which is presented by the web server undergoes several rigorous checks by the end user's web browser before the user is permitted to connect to the web server.

4.2.1. HTTPS protected Intranet Website Certificate Profile

Unlike S/MIME, there is no single accepted standard that dictates what fields a web server certificate must contain. RFC 2818 states that the only required check for a web browser to perform when validating a connection to a HTTPS site is to verify that the hostname of the site matches the hostname contained in the certificate. In an attempt to identify best practices for a certificate used to secure Intranet websites with HTTPS, portions of the Extended Validation certificate profile (CA/Browser Forum, 2008) were used to build the profile.

Wildcard Certificates

A common practice of commercial CAs (see (Thawte), (Trustwave), and (Verisign)) is to offer wildcard certificates which match multiple FQDNs that end in a common domain name. For example, the wildcard pattern "*.testbank.com" would match "login.testbank.com" and "accountmgr.testbank.com". The primary use case for wildcard certificates is to enable a customer to purchase a single certificate which can be used to secure multiple websites. Wildcard certificates also provide flexibility for the customer to change the FQDN of the web server if the server's role changes in the future.

However, wildcard certificates do introduce additional security risks. By the very nature of public key cryptography, the wildcard certificate's public key component will correspond to a single private key. If the wildcard certificate is used in multiple web servers, the private key will need to be duplicated across all web servers. This reduces the overall security of the wildcard certificate to the security of the least secure web server which uses the wildcard certificate. NIST SP800-57 part 1 (Barker, Barker, Burr, & Smid, 2007) recommends that private keys are stored only in a cryptomodule on the target system and that transport of the keys external to such a module should be limited.

In addition, a new type of attack, called 'null-prefix attacks' can exploit certain types of wildcard certificates by embedding additional domain names before the root domain component of a common name field. These attacks are possible because of security vulnerabilities in common SSL/TLS stacks which match up the hostname of a website against the common name field of the certificate (Marlinspike, 2009).

Therefore, from a security best practice perspective it is not recommended to use wildcard certificates.

Table 2 HTTPS Server Certificate Profile

	Name	Required Contents	Properties	Rationale
Certificate Field	startDate	UTCtime or Generalized Time stating when the certificate will become valid.		The validity field of the X.509 certificate is used by the CA to indicate when the certificate is considered 'valid' and may be used for its intended purpose. Auditors should be careful to understand circumstances under which it is permissible to issue certificates that are not yet or are no longer valid.
	endDate	UTCtime or Generalized Time stating when the certificate will no longer be considered valid.		
	signatureAlgorithm	sha1WithRSAEncryption		NIST SP800-57 part 1 provides guidance that after 2010, SHA1 should be deprecated in favor of the SHA2 variant of signature algorithms. The SHA2 family of hash functions are defined for use in the RSA signature algorithm in RFC 4055. Note that older web browsers will likely not support the newer signature algorithms.
	Version	3 (0x2)		The X.509 v1 and v2 certificate formats are considered obsolete and should not be used in production systems.
	subject	commonName=FQDN of the web server		The only check that is reliably enforced by browsers (per RFC 2818) when validating this field is matching the contents of the CN field or a pattern contained in the CN field against the web server's FQDN. See the discussion below regarding the use of wildcard certificates (certificates which use a pattern instead of a FQDN in the CN field). Other values such as the

				organizationName, organizationalUnit etc. may be included in the subject field. However, including these fields in an Intranet HTTPS website certificate do not add any additional security to the website.
	Issuer	Non-blank value that is different than the value of the subject field.		As with S/MIME, it is not a security best practice to use self-signed certificates to secure intranet HTTPS websites. Unless the user's browser is pre-configured with the certificates from each HTTPS web server that is used in the enterprise environment, the user will be required to enter a security exception to add the certificate to their browser's trusted certificate list. This practice has the side effect of conditioning users to expect security exceptions as part of their normal browsing experience which defeats the purpose of security exception warnings.
	Public Key Algorithm	rsaEncryption id-dsa id-ecPublicKey	For the RSA public key algorithm, a key length of 1024 or greater bits is recommended.	From pp 158 – 164 of *Applied Cryptography* (Schneier, 1996), public key lengths of 1024 bits were predicted to be sufficient for protecting information against attacks by individuals. Current RSA recommendations for RSA public key algorithms are 1024 bits for corporate use (RSA Security, 2009). NIST SP800-57 part 1 indicates that for government applications, a key length of 1024 is sufficient through the year 2010, with key lengths of 2048 bits sufficient for use through the year 2030 (Barker, Barker, Burr, & Smid, 2007). Other algorithms such as DSA and ecPublicKey (elliptic curve public key algorithm) are also acceptable; although at the time of writing the ecPublicKey is not widely supported by web server software.

Certificate Extension	subjectAltName	FQDN of the web server		The subjectAltName is used in the same manner as the commonName field in the certificate subject.
	cRLDistributionPoints	A non-blank distribution point name field.	Not critical	The use of CRLs to revoke web server certificates allows enterprise administrators to recover from key compromises. If this extension is not present, then the authorityInfoAccess extension must be present with a valid OCSP responder URI.
	extendedKeyUsageExtension	id-kp-serverAuth	Not critical	The id-kp-serverAuth key usage value is used to indicate that the certificate is used for web server authentication as documented in RFC 3280.

4.3. 802.1x Authentication for WLAN Environments

Wireless networks have continued to grow more and more pervasive throughout enterprise networks. WLAN technology is an important enabler for many business processes besides just 'cutting the Ethernet cord', such as: inventory control for warehouses, queue busting operations for busy retail stores, and even voice over WLAN.

While WLANs offer much more flexibility than wired networks, the lack of a physical wire to connect to the network makes it much harder to control who is permitted to access the WLAN. The need to authenticate users prior to accessing the WLAN and to enforce policies such as which network resources WLAN users are permitted to access has led to a need for a centralized authentication scheme for WLAN access. The 802.1x authentication scheme has proven to be an ideal solution for WLAN authentication and access control.

Briefly, 802.1x is an authentication mechanism that can be used to authenticate devices (and optionally users of those devices) to wired and wireless networks. In addition to authenticating devices and networks to each other, 802.1x is used to authorize the device/user to access network resources. 802.1x supports multiple authentication methods, all of which are tunneled between a client device and an authentication server using Extensible Authentication Protocol (EAP). EAP is a protocol that is designed to relay credentials between a device and an authenticating server. It acts as a transport layer for other authentication methods, such as EAP-TLS and EAP Tunneled TLS (EAP-TTLS). Both EAP-TLS and EAP-TTLS use X.509 certificates either as authentication credentials or to provide an additional layer of security for additional authentication schemes.

4.3.1. WLAN 802.1x Authentication Certificate Profile

For WLAN 802.1x authentication, there are 2 types of certificates to consider, a server certificate which is installed on a RADIUS authentication server used to authenticate the server to wireless clients, and a client certificate that is configured on a WLAN client supplicant to authenticate the client to the server. These certificates are

used in the various EAP methods used in 802.1x authentication. For purposes of this paper, only EAP-TLS will be discussed.

As defined in RFC 5216, the EAP-TLS authentication mechanism requires that the authenticator (typically, the RADIUS server but sometimes the local AP acts as the authenticator) present a server certificate to the client. RFC 5216 states that while client certificate authentication is not mandatory, in practice EAP-TLS requires that the client be configured with a certificate as well. This practice stems from the fundamental requirement that WLAN users should authenticate themselves to the network prior to accessing network services. Since EAP-TLS does not provide any other way for a user to supply credentials to the network, a client certificate is the only way that a user can present a credential to the network. The EAP-TLS RFC only suggests that client authentication be disabled only for special cases such as emergency network access.

At the time of writing, there are no standard certificate profiles defined for certificates used for 802.1x authentication in a WLAN environment. RFC 4334 standardizes the use of additional certificate extensions that may be embedded in a WLAN client certificate which indicate which WLANs the certificate may be used to authenticate to. These extensions, the WLAN SSID Public Key Certificate Extension and the WLAN SSID Attribute Certificate Attribute, include a list of SSIDs that the certificates are mapped to. However, these extensions only serve to help the wireless client choose which certificate to present to the network.

Therefore, to create a useful WLAN 802.1x certificate profile, it is necessary to first determine what types of security risks can be incurred if the client and/or server certificate contents are not well formed. From a wireless client perspective, one such threat to consider is the threat of an attacker attempting to emulate the legitimate enterprise wireless network through the use of a rogue AP + RADIUS server. If an attacker is able to obtain a server certificate that is signed by a legitimate CA[1] and has

[1] Note that this attack is easier than one might expect; at the time of this writing the built in Microsoft XP WLAN supplicant only requires that the WLAN server certificate be configured with the

modified the rogue RADIUS server to not validate the client certificate, the attacker may be able to successfully cause enterprise users to associate to the rogue AP instead of a legitimate enterprise AP. The attacker may then attempt to host services that appear to be enterprise services such as webmail to attempt to harvest the credentials of enterprise users.

This threat can be partially mitigated by ensuring that server certificates are issued only by an enterprise specific PKI (self signed certificates or certificates issued by an enterprise owned PKI). Another mitigation is to include specific information in both the server certificate and the CA root certificate used to issue the server certificate to indicate that the network is operated by the enterprise. These countermeasures will permit the enterprise client to view and validate the server certificate prior to accepting the connection with the WLAN.

From an enterprise perspective, the threat posed to enterprise WLANs is the risk of an unauthorized client gaining access to the WLAN network. EAP-TLS is a very effective authentication scheme to mitigate this risk due to the fact that it requires the client to be configured with a certificate that is trusted by the server. From a best practices perspective, the client certificate must not be signed by any publicly operated CA (if the network administrator uses a public CA for client certificates, he will be unable to restrict access to unauthorized users who have gotten a client certificate from the public CA). This practice allows the network administrator to restrict access only to devices which have certificates that have been issued by the enterprise CA. Finally, the client certificate must contain unique information identifying either the user or the client device. This practice will help ensure that even if an attacker is able to gain access to a client certificate and private key, the enterprise network administrators will be able to trace the attacker to a particular user or device.

OID for TLS server authentication (Microsoft, 2007). This OID is present in any well formed HTTPS certificate, so a web server certificate could easily be used for this attack.

In addition, the RADIUS server used to authenticate clients must ensure that it regularly checks for revoked client certificates to enable administrators to quickly revoke certificates for compromised users.

Finally, it should be noted that the client and server certificate profiles could also be used in a wired 802.1x authentication system where wired Ethernet switches are performing port based authentication. However, wired 802.1x authentication is less commonly used due to deployment issues (all network resources including printers, servers, and client PCs must have 802.1x supplicants) and less security benefits than other wired network authentication schemes such as IPsec (wired 802.1x authentication only authenticates 802.1x clients when the client first communicates with the switch, subsequent packets sent by the clients are not authenticated which does not prevent an attacker from splicing another system into the same switch port using a hub).

Table 3 802.1x WLAN Server Certificate Profile

	Name	Required Contents	Properties	Rationale
Certificate Field	startDate	UTCtime or Generalized Time stating when the certificate will become valid.		The validity field of the X.509 certificate is used by the CA to indicate when the certificate is considered 'valid' and may be used for its intended purpose. Auditors should be careful to understand circumstances under which it is permissible to issue certificates that are not yet or are no longer valid.
	endDate	UTCtime or Generalized Time stating when the certificate will no longer be considered valid.		
	signatureAlgorithm	sha1WithRSAEncryption		NIST SP800-57 part 1 provides guidance that after 2010, SHA1 should be deprecated in favor of the SHA2 variant of signature algorithms. The SHA2 family of hash functions are defined for use in the RSA signature algorithm in RFC 4055. Note that older web browsers will likely not support the newer signature algorithms.
	version	3 (0x2)		The X.509 v1 and v2 certificate formats are considered obsolete and should not be used in production systems.
	subject	commonName=DNS name of the RADIUS server organizationName=Company name		The commonName and organizationName fields are usually not validated programmatically by WLAN supplicants. Therefore, these fields should contain information that, if presented to the user, helps them be certain that the server certificate belongs to their organization.

	issuer	commonName=DNS name of the RADIUS server organizationName =Company name OR commonName=Name of the enterprise CA organizationName=Company name		For 802.1x certificates, it is considered best practice to either use a self-signed certificate for WLAN server certificates or to use an internally operated CA. Under no circumstances should a server certificate be issued by a public CA, as this essentially permits your enterprise clients to associate with any WLANs that are configured with certificates issued by a public CA.
	Public Key Algorithm	rsaEncryption id-dsa id-ecPublicKey	For the RSA public key algorithm, a key length of 1024 or greater bits is recommended.	From pp 158 – 164 of *Applied Cryptography* (Schneier, 1996), public key lengths of 1024 bits were predicted to be sufficient for protecting information against attacks by individuals. Current RSA recommendations for RSA public key algorithms are 1024 bits for corporate use (RSA Security, 2009). NIST SP800-57 part 1 indicates that for government applications, a key length of 1024 is sufficient through the year 2010, with key lengths of 2048 bits sufficient for use through the year 2030 (Barker, Barker, Burr, & Smid, 2007). Other algorithms such as DSA and ecPublicKey (elliptic curve public key algorithm) are also acceptable; although at the time of writing the ecPublicKey is not widely supported by web server software.
Certificate Extension	subjectAltName	DNS name of the RADIUS server		The subjectAltName is used in the same manner as the commonName field in the certificate subject.

	cRLDistributionPoints	A non-blank distribution point name field.	Not critical	The server must be configured to check if client certificates have been revoked. This enables enterprise administrators to quickly revoke credentials for compromised users. Alternatively, the Authority Information Access extension may be used to enable OCSP for this purpose
	extendedKeyUsageExtension	id-kp-serverAuth Optional: id-pe-authorityInfoAccess id-ad-ocsp accessLocation=URL to OCSP server	Not critical	The id-kp-serverAuth key usage value is required by some WLAN clients (Windows XP). In the event that OCSP is used for revocation checks, the id-pe-authorityInfoAccess, id-ad-ocsp, and accessLocation fields must be present and populated.

Table 4 802.1x WLAN Client Certificate Profile

	Name	Required Contents	Properties	Rationale
Certificate Field	startDate	UTCtime or Generalized Time stating when the certificate will become valid.		The validity field of the X.509 certificate is used by the CA to indicate when the certificate is considered 'valid' and may be used for its intended purpose. Auditors should be careful to understand circumstances under which it is permissible to issue certificates that are not yet or are no longer valid.
	endDate	UTCtime or Generalized Time stating when the certificate will no longer be considered valid.		
	signatureAlgorithm	sha1WithRSAEncryption		NIST SP800-57 part 1 provides guidance that after 2010, SHA1 should be deprecated in favor of the SHA2 variant of signature algorithms. The SHA2 family of hash functions are defined for use in the RSA signature algorithm in RFC 4055. Note that older web browsers will likely not support the newer signature algorithms.
	version	3 (0x2)		The X.509 v1 and v2 certificate formats are considered obsolete and should not be used in production systems.
	subject	commonName=RFC822 name of the user ('user@domain') OR commonName=MAC		The commonName field should contain information that can be used by the RADIUS server to map the client certificate to a particular user (or system).

	address of the WLAN client (encoded as a printableString or utf8String)		
issuer	commonName=Name of the enterprise CA organizationName=Company name		For 802.1x certificates, it is considered best practice to either use a self-signed certificate for WLAN server certificates or to use an internally operated CA. Under no circumstances should a client certificate be issued by a public CA, as this may allow the RADIUS server to accept client certificates from clients which have not been authorized by the enterprise network administrator.
Public Key Algorithm	rsaEncryption id-dsa id-ecPublicKey	For the RSA public key algorithm, a key length of 1024 or greater bits is recommended.	From pp 158 – 164 of *Applied Cryptography* (Schneier, 1996), public key lengths of 1024 bits were predicted to be sufficient for protecting information against attacks by individuals. Current RSA recommendations for RSA public key algorithms are 1024 bits for corporate use (RSA Security, 2009). NIST SP800-57 part 1 indicates that for government applications, a key length of 1024 is sufficient through the year 2010, with key lengths of 2048 bits sufficient for use through the year 2030 (Barker, Barker, Burr, & Smid, 2007). Other algorithms such as DSA and ecPublicKey (elliptic curve public key algorithm) are also acceptable; although at the time of writing the ecPublicKey is not widely supported by web server software.

Certificate Extension	subjectAltName	RFC822 name of the user ('user@domain')		The subjectAltName is used in the same manner as the commonName field in the certificate subject. Note that unlike the CN field, there is no defined way to encode a MAC address into the subjectAltName field.
	cRLDistributionPoints	A non-blank distribution point name field.	Not critical	The client must be configured to validate that server certificates have not been revoked by the issuing certificate authority. Alternatively, the Authority Information Access extension may be used to enable OCSP for this purpose
	extendedKeyUsageExtension	id-kp-serverAuth Optional: id-pe-authorityInfoAccess id-ad-ocsp accessLocation=URL to OCSP server	Not critical	The id-kp-clientAuth key usage value is required by some RADIUS servers (specifically RADIUS services that are implemented by Microsoft IAS server) In the event that OCSP is used for revocation checks, the id-pe-authorityInfoAccess, id-ad-ocsp, and accessLocation fields must be present and populated.

4.4. IPSec

Another very common use case for enterprise environments is the requirement to extend the enterprise network to other locations (branch offices, employee homes) without using expensive dedicated leased lines. A common solution to this problem is to create a VPN over an open public network to virtually extend the enterprise network. Any VPN solution must ensure that the confidentiality and integrity of the information exchanged over the VPN is maintained.

IPSec is one of the most commonly used technologies to provide VPN services to enterprise environments. IPSec consists of multiple protocols: Authenticating Header (AH), Encapsulating Security Payload (ESP), Internet Key Exchange (IKE), and Internet Security Association and Key Management Protocol (ISAKMP). IPSec permits 2 IP endpoints to negotiate the use of either AH or ESP to protect IP packets which provide authentication and encryption services respectively. IPSec accomplishes this by using the ISAKMP protocol to authenticate and negotiate security associations, using keying material that is generated by the use of IKE. IKE supports the use of both pre-shared keys and X.509 certificates as credentials used to authenticate IPSec endpoints.

4.4.1. IPSec Certificate Profile

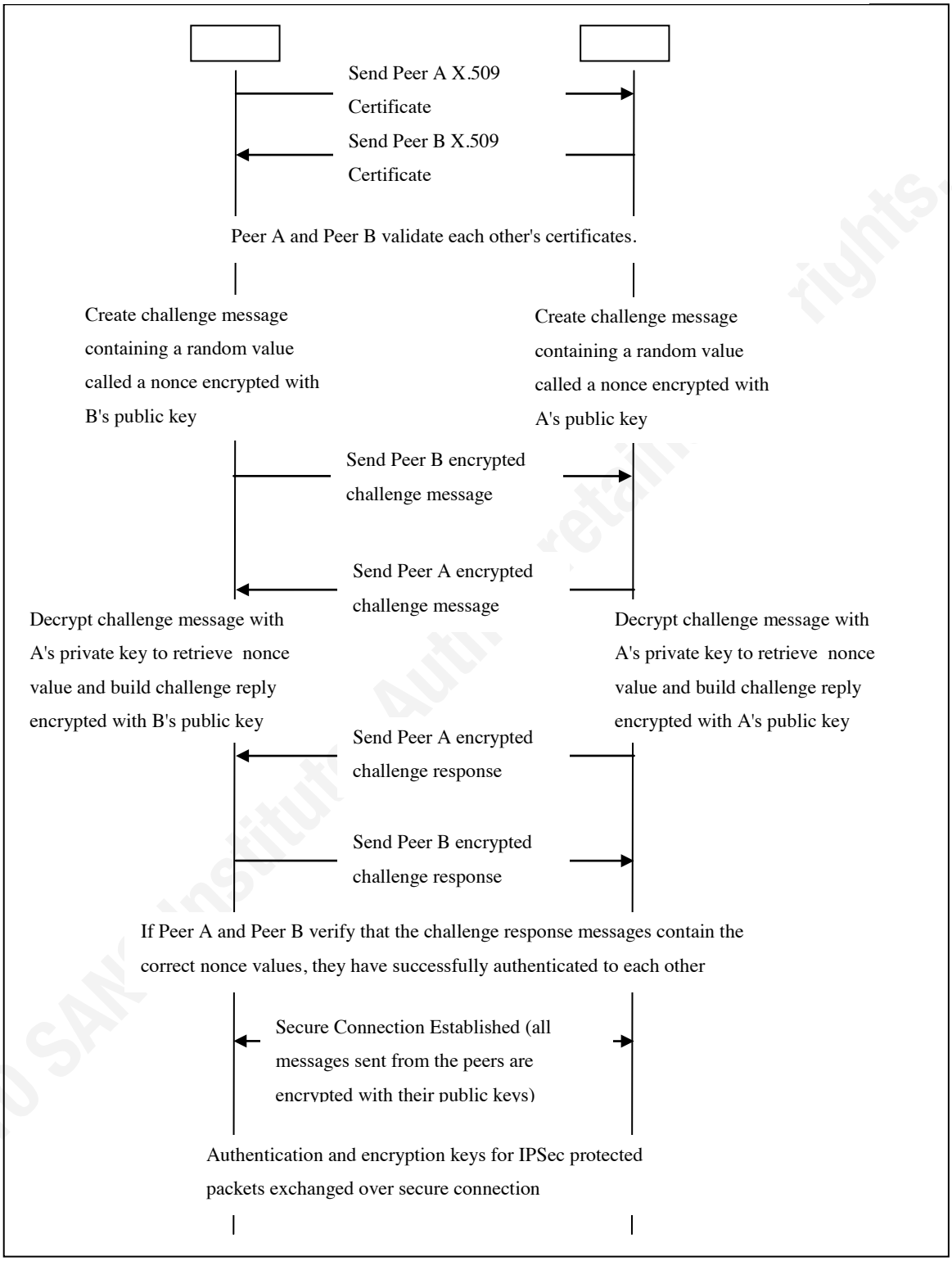
The IPSec working group has published an RFC (RFC 4945) which defines the contents of the certificates used by IPSec peers to authenticate each other. This RFC, unlike many of the other certificate usages described in this paper, is very complete in describing the content and requirements for the fields of the X.509 certificates. Therefore, the profile defined here will be entirely based on this RFC.

To understand the properties of a well-formed X.509 certificate for use in IPSec implementations, it is necessary first to understand how the X.509 certificate is used to setup an IPSec connection. IPSec connections are setup in 2 phases. In the first phase the endpoints authenticate each other by exchanging credentials. In the second phase, the endpoints exchange key material that will be used to authenticate and encrypt IP packets as well as other information needed for the IPSec connection to be established.

Walter Goulet, wgoulet@gmail.com

The protocol used to setup the IPSec connection is the Internet Key Exchange (IKE) protocol. The following figure illustrates the IKE process at a high level where both peers are using X.509 certificates for authentication.

Figure 3 IKE Key Exchange



As part of the certificate validation process, the IPSec clients on both endpoints perform additional validation of the certificate such as validating that the extensions contain the hostname or IP address of the peer and that an appropriate security policy for

the peer is configured on the endpoints. For example, VPN gateway A accepting a point to point connection from VPN gateway B could use information in the subject field to determine if VPN gateway B is permitted to initiate connections based on local security policy.

Note that the IKE protocol permits the use of pre-shared keys or X.509 certificates for the peer endpoints to authenticate with each other. X.509 certificates are generally used for scenarios where there are multiple hosts that must connect to a single endpoint (IPSec VPN gateway) and other scenarios where exchanging pre-shared keys is not feasible. In addition, X.509 certificates provide both endpoints more control in authenticating endpoints by taking advantage of PKI features such as certificate revocation.

Note that IPSec connections are generally host to host connections since they encrypt traffic at layer 3 of the network stack (the logic being that since user sessions are usually established at the application layer, user identity and credentials should only authenticate connections at the application layer). Therefore, the X.509 certificate usually contains information that identifies the endpoints that are establishing the connections and usually has little or no information about the user that is establishing the connection. Instead, the certificate contains information that is tied to the host such as IP address, DNS name, or other host specific identifiers. One exception to this case is if IPSec is used as part of a remote access VPN where end users use X.509 certificates to authenticate to a VPN gateway. In this case, the client certificate may contain additional information in the subject field that identifies the user.

Table 5 IPSec Certificate Profile

	Name	Required Contents	Properties	Rationale
Certificate Field	startDate	UTCtime or Generalized Time stating when the certificate will become valid.		The validity field of the X.509 certificate is used by the CA to indicate when the certificate is considered 'valid' and may be used for its intended purpose. Auditors should be careful to understand circumstances under which it is permissible to issue certificates that are not yet or are no longer valid.
	endDate	UTCtime or Generalized Time stating when the certificate will no longer be considered valid.		
	signatureAlgorithm	sha1WithRSAEncryption		NIST SP800-57 part 1 provides guidance that after 2010, SHA1 should be deprecated in favor of the SHA2 variant of signature algorithms. The SHA2 family of hash functions are defined for use in the RSA signature algorithm in RFC 4055. Note that older web browsers will likely not support the newer signature algorithms.
	version	3 (0x2)		Only X.509v3 certificates support the extensions that are mandated by RFC 4945.
	subject	commonName = [non blank value] Organization = [non blank value] Organizational Unit = [non blank value]	OR field is blank	The contents of the subject field are used by IPSec endpoints as identifiers for selecting security policies appropriate for the connection. RFC 4945 requires that this field either be blank (in which case the subjectAltName extension is used for policy selection), or that the commonName, Organization, Organizational Unit and Country fields be populated.

		Country = [non blank value]		
	issuer	commonName= [common name of the issuing CA]		Similar to the recommendations in Table 4, the CA used to issue certificates to IPSec endpoints should not be a public CA.
	Public Key Algorithm	rsaEncryption id-dsa id-ecPublicKey	For the RSA public key algorithm, a key length of 1024 or greater bits is recommended.	From pp 158 – 164 of Applied Cryptography (Schneier, 1996), public key lengths of 1024 bits were predicted to be sufficient for protecting information against attacks by individuals. Current RSA recommendations for RSA public key algorithms are 1024 bits for corporate use (RSA Security, 2009). NIST SP800-57 part 1 indicates that for government applications, a key length of 1024 is sufficient through the year 2010, with key lengths of 2048 bits sufficient for use through the year 2030 (Barker, Barker, Burr, & Smid, 2007). Other algorithms such as DSA and ecPublicKey (elliptic curve public key algorithm) are also acceptable; although at the time of writing the ecPublicKey is not widely supported by web server software.
Certificate Extension	subjectAltName	FQDN of the host OR IP address of the host OR RFC822 name of the user ('user@domain')		The subjectAltName is used in IPSec certificates to identify the peers of the IPSec connection.
	cRLDistributionPoints	A non-blank distribution point name field.	Not critical	The client must be configured to validate that server certificates have not been revoked by the issuing certificate authority. Alternatively, the Authority Information Access extension may be used

				to enable OCSP for this purpose
	keyUsage	digitalSignature nonRepudiation		RFC 4945 does not mandate that this extension be included in IPSec certificates. However, as the keyUsage extension is useful in helping restrict the use of the certificate, it should be included as a best practice. When it is included, the digitalSignature and nonRepudation usages should be specified.

4.5. Enterprise CA Root Certificate Profile

The final certificate profile to consider is the CA root certificate profile. The root certificate profile must be carefully considered to ensure that the CA can be fully trusted to issue certificates. In general, the CA root certificate must have the following properties: a large keysize (2048 or greater), unique identifying information about the enterprise and organization that is operating the CA in the subject/issuer fields, and extensions which limit the CA root certificate to only signing certificates and CRLs.

The CA root certificate must not be overloaded and used for other purposes (even though the X.509 standard permits this). The reason for this is that if the CA root certificate is used for other purposes such as digitally signing documents, additional users must be given access to the CA root private key which increases the risk of the CA root certificate private key being compromised. The following table specifies a best practice CA root certificate profile.

Table 6 Root CA Certificate Profile

	Name	Required Contents	Properties	Rationale
Certificate Field	startDate	UTCtime or Generalized Time stating when the certificate will become valid.		The validity field of the X.509 certificate is used to indicate when the CA certificate can be trusted to sign certificate requests and CRLs.
	endDate	UTCtime or Generalized Time stating when the certificate will no longer be considered valid.		
	signatureAlgorithm	sha1WithRSAEncryption		NIST SP800-57 part 1 provides guidance that after 2010, SHA1 should be deprecated in favor of the SHA2 variant of signature algorithms. The SHA2 family of hash functions are defined for use in the RSA signature algorithm in RFC 4055. Note that older web browsers will likely not support the newer signature algorithms.
	version	3 (0x2)		The X.509 v1 and v2 certificate formats are considered obsolete and should not be used in production systems.
	subject	[non-blank]		There are no general guidelines published that indicate what fields should be included in a CA root certificate subject. In general, security best practices dictate that the subject contain identifying information about the entity that issued the certificate so that end users can easily determine that the certificate was issued by an organization they trust. For example, the Organization field could be set to the name of the

				corporation that the CA belongs to. The Organizational Unit could contain the name of the division within the corporation that operates the PKI. Finally, the Common Name could summarize the purpose of the CA (e.g. 'Network CA', 'Digital Communcations CA' etc).
	issuer	identical to the subject field		CA root certificates are always self-signed; therefore the issuer field should be the same as the subject field.
	Public Key Algorithm	rsaEncryption id-dsa id-ecPublicKey	For the RSA public key algorithm, a key length of 2048 or greater bits is recommended.	For root CA certificates, security best practices dictate that the key length should be double or even quadruple the length used for issued certificates. This is due to the fact that the security requirements for the CA root certificate are much stricter than other certificates. 2048 bit key length is considered a minimum, with 4096 bit keys preferable.
Certificate Extension	keyUsage	keyCertSign cRLSign	Critical	The keyUsage bits keyCertSign and cRLSign are required as a minimum to be present in the CA certificate. Other key uses are possible, but the assessor should carefully evaluate whether or not each certificate key usage is appropriate for a root CA certificate. This mainly applies to the extendedKeyUsageExtension.
	basicConstraints	cA=1	Critical	This field must include the cA value which must be set to '1' to indicate that the certificate is a CA certificate. The pathLenConstraint may optionally be included to restrict the number of subCA certificates between the CA and end user certificates.

	certficicatePolicies	policyIdentifier = [non blank] policyQualifiers= [non blank]	Critical	While this field is not required by RFC 3280, as a security best practice this field should always be populated to point to a location where the user can access the CP/CPS. The makeup of this field is rather complex, but at a minimum it should include a URL where the user can access the CP/CPS. This field should be marked as 'Critical' as well to indicate that it should be processed by client software, although in practice many implementations will ignore this field or only present the user with a GUI to display the contents of this field.

5. Assessing Enterprise PKI Deployments

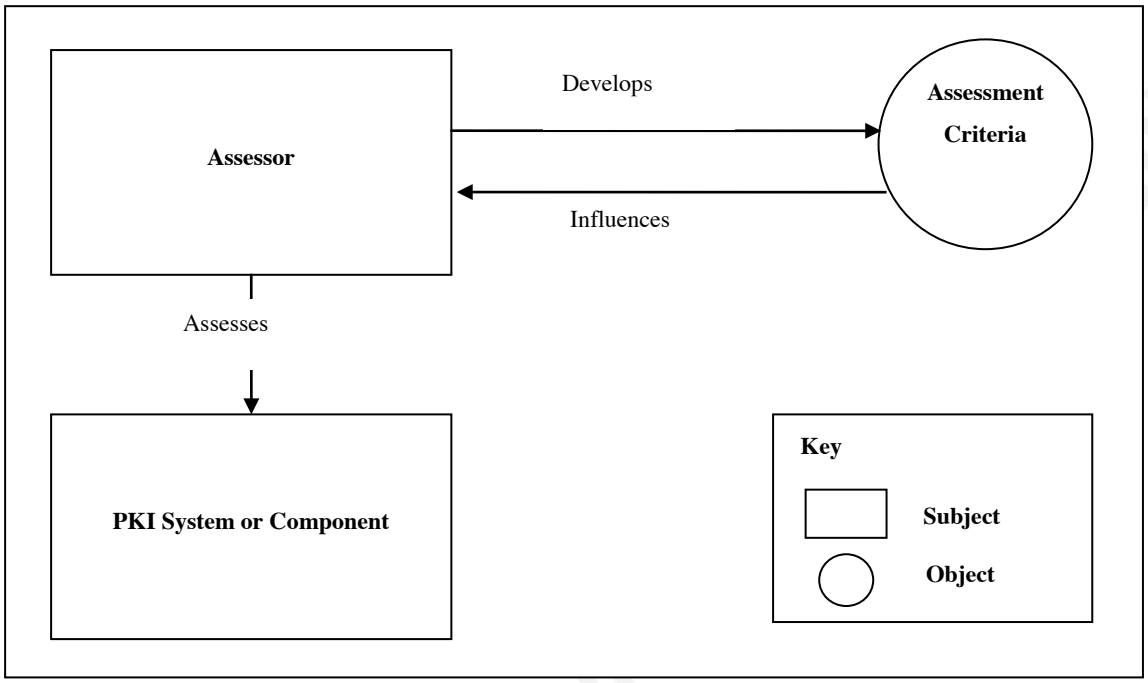
Assessing an enterprise PKI deployment consists of 2 basic steps: assessing the policies that govern operation of the PKI and assessing the technical controls that are in place in the PKI infrastructure. The ABA PKI Assessment Guidelines (Information Security Committee, American Bar Association, 2001) provide an approach to assessing a PKI deployment in a checklist fashion. This paper presents a slightly modified version of this checklist to make it appropriate for enterprise PKI deployments.

5.1. PKI Assessment Process

The ABA PKI assessment methodology identifies 3 primary actors involved in the PKI assessment: the assessors performing the assessment, the Policy Authority which develops the requirements for the PKI, and the party responsible for operating the PKI being assessed (PKI operator).

The following figure (adopted from the ABA PAG) depicts the PKI assessment process:

Figure 4 ABA PKI PAG Process



The Policy Authority is typically responsible for developing the Certificate Policy which dictates the requirements of the PKI. The PKI Operator is responsible for developing the Certificate Practices Statement which states how the PKI implementation meets the requirements specified in the CP.

The 3 subjects that are assessed in an enterprise PKI are the CP/CPS governing the operation of the PKI, the certificates that are issued by the PKI, and the physical infrastructure that hosts the PKI. This paper presents assessment criteria to evaluate the first 2 subjects of the PKI. An assessment criteria is not provided for the 3rd subject, but standards are provided which can be used as a baseline for assessing physical components of the PKI.

5.1.1. Policy Assessment

A PKI deployment generally should have a Certificate Policy and Certificate Practices Statement (CP & CPS) which state the requirements for the PKI and dictates how the PKI is to be operated. RFC 3647 lays out the basic framework for both the CP and CPS.

The assessment criteria for CP/CPS evaluation is presented as a checklist that is included in the appendix. This checklist lists the CP/CPS sections mandated by RFC 3647 along with assessment notes that are adopted from the ABA PAG. The spreadsheet is laid out as follows:

				S/MIME	HTTPS	802.1x	IPSec	Assessment Guidelines	Assessment Results
1	INTRODUCTION			M	M	M	M		
	1.1	Overview		M	M	M	M		
	1.2	Document name and identification		M	M	M	M	Section 1.3 should be specified in CPS for all PKI use cases since it essentially defines what the PKI is used for and who does what.	

The columns for each of the PKI enterprise use cases are marked with a M (Mandatory) or O (Optional) to indicate if the section of the CP/CPS must be in place for each type of PKI.

The assessment is performed as follows:

1. Gather the CP from the Policy Authority (if applicable)
2. Gather the CPS from the PKI system owner
3. Verify that the CPS has a section corresponding to each completed section of the CP (the requirement is that the CPS should have the same outline as the CP and should clearly indicate where it satisfies the requirements stated in the CP).
4. Step through the assessment spreadsheet and verify that each section that is marked as 'Mandatory' for the particular PKI use case is addressed in the CP/CPS.

Walter Goulet, wgoulet@gmail.com

5.2. Certificate Profile Assessment

Assessing the certificate profile used by an enterprise PKI is accomplished by determining whether or not the certificates issued by the enterprise PKI conform to the best practice certificate profiles presented in section 4.4.1. To assist with this effort, a companion spreadsheet and a script were developed that can help assessors quickly evaluate individual certificate profiles.

Current versions of the certificate spreadsheets and source for the accompanying script are posted in the accompanying google.code site for this paper: http://code.google.com/p/ assesspki/downloads/list. The script is also documented in Appendix B.

5.2.1. Certificate Profile Spreadsheet

The certificate profile spreadsheet outlines the overall structure of each of the X.509 certificates that are used by the four PKI use cases defined in this paper. In addition, a certificate profile spreadsheet is defined for a general purpose root CA certificate. The figure below show an example of a certificate field included in the spreadsheet.

Name	Required Contents	Properties	Rationale	Assessed Certificate Contents		Assessor Notes
startDate	UTCtime or Generalized Time stating when the certificate will become valid.		The validity field of the X.509 certificate is used by the CA to indicate when the certificate is considered 'valid' and may be used for its intended purpose. Auditors should be careful to understand circumstances under which it is permissible to issue certificates that are not yet or are no longer valid.	Dec 18 00:00:00 2009 GMT		

The fields are defined as follows:

- Name - Name of the certificate field
- Required Contents - the value that the field should be set to in each certificate profile.

Walter Goulet, wgoulet@gmail.com

- Properties - additional properties (ex. criticality value for extensions) that apply to the certificate field.
- Rationale - summary text explaining how the required contents value was arrived at.
- Assessed Certificate Contents - value extracted from the certificate being assessed.
- Assessor Notes - field for assessors to capture notes and results of the assessment.

The profile checklists make use of macros included in the spreadsheets to permit assessors to easily import certificates to be assessed directly into the spreadsheet. The filename of the profile checklist indicates which PKI use case it is used for.

5.2.2. Extracting Fields from Certificates

The OpenSSL command line utility can display certificate fields. However, the default output format is not straightforward to read for assessment purposes. To make the certificates easier to read and import into the certificate profile spreadsheets, a Perl script (included in Appendix B) was written to invoke the openssl utility, parse the certificate fields from the openssl output and create an XML output file with the certificate field. The script is invoked as follows:

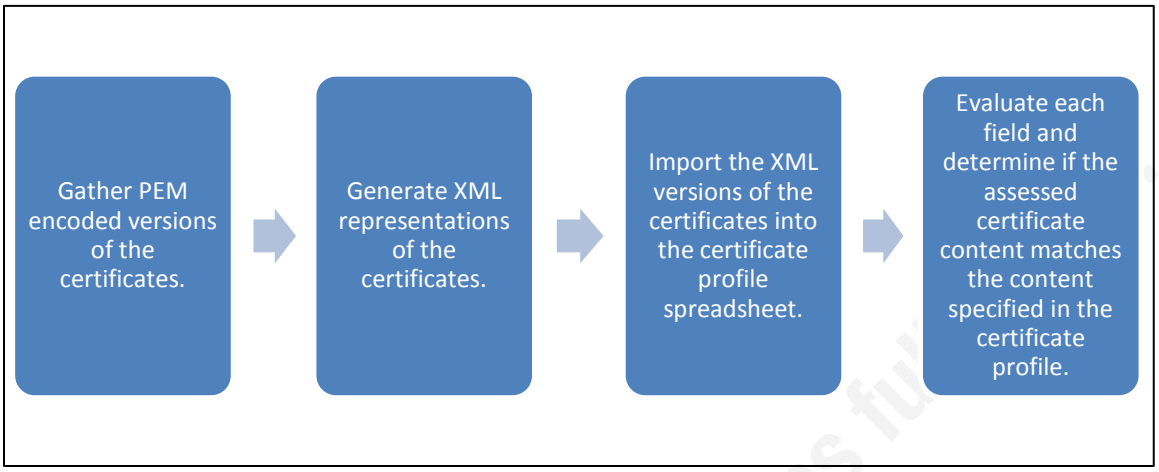
```
perl cert2xml.pl [filename of PEM formatted certificate] > cert.xml
```

The cert2xml.pl script requires that OpenSSL 0.9.8h or later is installed on the system. The script assumes that the openssl binary is in the current path. The script was tested on both Windows and Linux (for Windows, win32 prebuilt binaries of openssl and grep must be installed; they can be downloaded from http://gnuwin32.sourceforge.net/packages.html).

5.2.3. Certificate Profile Assessment Process

At a high level, the profile assessment process looks like this:

Figure 5 Certificate Assessment Process



1. The first step is to collect the certificates to be assessed in a common location and convert them to the Privacy Enhanced Mail (PEM) format. PEM formatted certificates are encoded as Base64 text. This conversion step will allow the certificate contents to be easily captured in assessment reports. The OpenSSL command-line utility can be used as follows to convert from other certificate formats to the PEM format. For example, the following example converts certificates from the Digital Encoding Rules (DER) format to the PEM format:

   ```
   openssl x509 -in cert.crt -inform DER -outform PEM -out cert.pem
   ```

   Note that many software utilities will give you the option of exporting certificates as DER encoded binary format or as PEM/Base64 format. If you choose the second option, this conversion step is not necessary.
2. Create the XML representations of the certificate contents using the cert2xml.pl perl script as follows:

   ```
   perl cert2xml.pl cert.pem > cert.xml
   ```
3. Open the certificate profile spreadsheet that matches the certificate being assessed. The spreadsheet filenames indicate the corresponding certificate profile.
4. Click the 'Import Certificate XML' button on the spreadsheet. You will be prompted for the filename of a certificate XML file generated in Step 2 above. Select the desired file and click OK. The Assessed Certificate Contents column will be populated with the certificate fields.

Walter Goulet, wgoulet@gmail.com

5. For each certificate field, determine if the value from the certificate matches the profile specification. Refer to the Rationale column for additional guidance and direction as certain certificate fields support multiple values. Fill in the Notes column with the assessment results.

5.3. PKI Infrastructure Assessment Standards

Assessing the physical components of the PKI is out of scope of this paper, however the following publications provide useful guidance to assessors to assist in evaluating PKI infrastructure:

- The NIST Guide to General Server Security (Karen Scarfone,Wayne Jansen,Miles Tracy, 2008) contains a generalized approach to securing servers. This guide is appropriate for supporting systems (management servers used to manage PKI components etc), but is not necessarily strict enough for the CA system itself (the system that actually signs certificate requests to generate certificates and hosts the root certificate private key).
- The US Federal Government's FPKIPA (Federal Public Key Infrastructure Policy Authority, http://www.idmanagement.gov/fpkipa/) publishes the Certificate Policy for the US Federal PKI (http://www.idmanagement.gov/fpkipa/documents/FBCA_CP_RFC3647.pdf). This Certificate Policy contains guidance in Sections 5 & 6 that list technical requirements that should be satisfied by the CA system. Vendor documentation and third party security benchmarks such as CIS can also be used to determine how to best configure the security controls on the CA system.

6. Conclusion

PKI deployments can provide many security services and benefits to enterprises. However, unless the PKI is deployed and operated in accordance with security best practices, the security benefits will not be realized as attackers can take advantage of

weaknesses in the deployment to forge certificates, gain access to the infrastructure and so on.

This paper has introduced the motivation and concepts behind PKI as well as a technical discussion of the components of a PKI. A set of certificate profiles were presented for four common PKI enterprise deployment use cases. Each profile provides guidance on what the certificate contents should be for each use case. Note that each of these profiles were developed with security best practices in mind and not based only on vendor recommendations. A CP/CPS outline was also developed that highlights which sections of a CP/CPS should be written for each enterprise PKI use case.

Finally, this paper presented a lightweight PKI assessment process to evaluate the CP/CPS as well as certificate profiles for each of the PKI use cases. Automated assessment checklists and a 'certificate to XML' script are presented to assist auditors in automating the assessment process. The automation built into the spreadsheets and the script will help minimize the amount of time required for auditors to manually populate checklists and reduce the overall effort to complete the assessment.

Auditors who are tasked with assessing the overall security posture of an enterprise will be able to use the assessment processes and tools presented in this paper to effectively and consistently audit enterprise PKI deployments.

7. Bibliography

Barker, E., Barker, W., Burr, W. P., & Smid, M. (2007, March). *NIST SP800-57 Recommendation for Key Management, Part 1: General*. Retrieved September 5, 2009, from NIST Computer Security Resource Center: http://csrc.nist.gov/publications/nistpubs/800-57/sp800-57-Part1-revised2_Mar08-2007.pdf

CA/Browser Forum. (2008, April). *Guidelines for the Issuance and Management of Extended Validation Certificates*. Retrieved September 13, 2009, from

CA/Browser Forum: http://www.cabforum.org/EV_Certificate_Guidelines_V11.pdf

Dougherty, C. R. (2009, 1 21). *Vulnerability Note VU#836068 MD5 vulnerable to collision attacks*. Retrieved August 26, 2009, from US CERT - United States Computer Emergency Response Team: http://www.kb.cert.org/vuls/id/836068

Ellison, C., & Schneier, B. (2000, November 1). *Ten Risks of PKI: What You're Not Being Told About Public Key Infrastructure*. Retrieved August 12, 2009, from Schneier on Security: http://www.schneier.com/paper-pki.pdf

Goffee, N., Kim, S. H., Smith, S., Taylor, P., Zhao, M., & Marchesini, J. (2004, April 12). *3rd Annual PKI R&D Workshop - Greenpass: Decentralized, PKI Authorization for Wireless LANs*. Retrieved August 12, 2009, from Internet 2 Middleware: http://middleware.internet2.edu/pki04/proceedings/greenpass-present.pdf

Guida, Rich; Johnson & Johnson. (2004, April 12 - 14). *3rd Annual PKI R&D Workshop - Johnson & Johnson Use of PKI*. Retrieved August 12, 2009, from Internet 2 Middleware: http://middleware.internet2.edu/pki04/proceedings/johnson_johnson.ppt

Hinchcliffe, D. (2009, May 15). *The year of the shift to Enterprise 2.0*. Retrieved August 17, 2009, from ZDNet: http://blogs.zdnet.com/Hinchcliffe/?p=382

Information Security Committee, American Bar Association. (2001, June 18). *PKI Assessment Guidelines*. Retrieved August 12, 2009, from American Bar Association: http://www.abanet.org/scitech/ec/isc/pagv30.pdf

Karen Scarfone,Wayne Jansen,Miles Tracy. (2008, July). *SP800-123 Guide to General Server Security*. Retrieved January 27, 2010, from NIST Computer Security Website: http://csrc.nist.gov/publications/nistpubs/800-123/SP800-123.pdf

Kuhn, R., Hu, V. C., Polk, W. T., & Chang, S.-J. (2001, Febuary 26). *NIST SP800-32 Introduction to Public Key Technology and the Federal PKI Infrastructure*. Retrieved August 12, 2009, from NIST Computer Security Resource Center: http://csrc.nist.gov/publications/nistpubs/800-32/sp800-32.pdf

Marlinspike, M. (2009, July 29). *Null Prefix Attacks against SSL/TLS Certificates*. Retrieved October 7, 2009, from Thoughtcrimes: http://www.thoughtcrime.org/papers/null-prefix-attacks.pdf

Microsoft. (2007, March 19). *Microsoft TechNet*. Retrieved November 3, 2009, from Troubleshooting IEEE 802.11 Wireless Access with Microsoft Windows: http://technet.microsoft.com/en-us/library/bb457017.aspx#EDAA

Peckham, L. (2003, October 21). *A Business Perspective on PKI: Why Many PKI Implementations Fail, and Success Factors To Consider*. Retrieved Febuary 2, 2010, from SANS InfoSec Reading Room - Encryption & VPNs: http://www.sans.org/reading_room/whitepapers/vpns/a_business_perspective_on_pki_why_many_pki_implementations_fail_and_success_factors_to_consider_728

RSA Security. (2009). *3.1.5 How large a key should be used in the RSA cryptosystem?* Retrieved August 31, 2009, from RSA Laboratories: http://www.rsa.com/rsalabs/node.asp?id=2218

S. Chokhani Orion Security Solutions, I., W. Ford VeriSign, I., LLP, R. S., C. Merrill McCarter & English, L., & S. Wu Infoliance, I. (2003, November). *RFC 3647 Internet X.509 Public Key Infrastructure Certificate Policy and Certification Practices Framework*. Retrieved December 29, 2009, from IETF: http://www.ietf.org/rfc/rfc3647.txt

Schneier, B. (1996). *Applied Cryptography*. John Wiley & Sons.

Thawte. (n.d.). Retrieved from Thawte Wildcard SSL Certificates: http://www.thawte.com/ssl/wildcard-ssl-certificates/index.html

The Cryptix Foundation Limited and David Hopwood. (2002, October 22). *Message Digest Algorithms*. Retrieved August 26, 2009, from Standard Cryptographic Algorithm Naming: http://www.users.zetnet.co.uk/hopwood/crypto/scan/md.html#MD2

Trustwave. (n.d.). Retrieved from Trustwave Wildcard SSL Certificates: https://www.trustwave.com/sslWildcard.php

Verisign. (n.d.). *Verisign Wildcard SSL Certificates*. Retrieved from http://www.verisign.com/ssl-certificates/wildcard-ssl-certificates/index.html

Walter Goulet, wgoulet@gmail.com

Appendix A. CP/CPS Checklist

The following table lists out a checklist for a Certificate Policy and Certificate Practices Statement that can be used to assess the CP/CPS for an enterprise PKI as described in Section 5.1.1.

				S/MIME	HTTPS	802.1x	IPSec	Assessment Guidelines
1	INTRODUCTION			M	M	M	M	
	1.1	Overview		M	M	M	M	
	1.2	Document name and identification		M	M	M	M	Section 1.3 should be specified in CPS for all PKI use cases since it essentially defines what the PKI is used for and who does what.
	1.3	PKI participants		M	M	M	M	
		1.3.1	Certification authorities	M	M	M	M	
		1.3.2	Registration authorities	M	M	M	M	
		1.3.3	Subscribers	M	M	M	M	
		1.3.4	Relying parties	M	M	M	M	
		1.3.5	Other participants	M	M	M	M	
	1.4	Certificate usage		M	M	M	M	
		1.4.1	Appropriate certificate uses	M	M	M	M	
		1.4.2	Prohibited certificate uses	M	M	M	M	
	1.5	Policy administration		M	M	M	M	
		1.5.1	Organization administering the document	M	M	M	M	
		1.5.2	Contact person	M	M	M	M	

		1.5.3	Person determining CPS suitability for the policy	M	M	M	M	
		1.5.4	CPS approval procedures	M	M	M	M	
	1.6	Definitions and acronyms						
2	PUBLICATION AND REPOSITORY RESPONSIBILITIES							This section is mainly process documentation that governs how the doc is change controlled. Should apply to all PKI use cases.
	2.1	Repositories		M	M	M	M	
	2.2	Publication of certification information		M	M	M	M	
	2.3	Time or frequency of publication		M	M	M	M	
	2.4	Access controls on repositories		M	M	M	M	
3	IDENTIFICATION AND AUTHENTICATION (11)							This section is absolutely required for S/MIME certificates since those certificates are credentials for end users. Therefore, it's key to ensure that a S/MIME PKI have detailed procedures in place to address identification and authentication for end users. For the other 3 PKI purposes, it is less important to verify the identity of end systems that the certificates will be installed on.
	3.1	Naming		M	M	M	M	Naming conventions should be clearly specified in the CP/CPS for each enterprise PKI use case. Note that all subsections should be completed for S/MIME PKIs.
		3.1.1	Types of names	M	O	O	O	
		3.1.2	Need for names to be meaningful	M	O	O	O	

		3.1.3	Anonymity or pseudonymity of subscribers	M	O	O	O	
		3.1.4	Rules for interpreting various name forms	M	O	O	O	
		3.1.5	Uniqueness of names	M	O	O	O	
		3.1.6	Recognition, authentication, and role of trademarks		O	O	O	
	3.2	Initial identity validation		M	O	O	O	Validating the identity of the certificate end user is critical for S/MIME PKI, but for other PKIs it is optional.
		3.2.1	Method to prove possession of private key	M	O	O	O	
		3.2.2	Authentication of organization identity	M	O	O	O	
		3.2.3	Authentication of individual identity	M	O	O	O	
		3.2.4	Non-verified subscriber information	M	O	O	O	
		3.2.5	Validation of authority	M	O	O	O	
		3.2.6	Criteria for interoperation	M	O	O	O	
	3.3	Identification and authentication for re-key requests		M	M	M	M	Again, for S/MIME re-key request procedures should be clearly spelled out since a re-key request essentially revokes an existing certificate, thus an unauthenticated re-key request could invalidate a valid end user S/MIME certificate. Re-key requests should also be addressed for the other PKI use cases.
		3.3.1	Identification and authentication for routine re-key	M	M	M	M	
		3.3.2	Identification and authentication for re-key after revocation	M	M	M	M	

	3.4	Identification and authentication for revocation request		M	M	M	M	As for re-key request, revocation request procedures should be clearly documented and include details on how the identity of the revocation requester is verified.
4	CERTIFICATE LIFE-CYCLE OPERATIONAL REQUIREMENTS (11)							Similar to Section 3, this section is critical for the S/MIME PKI use case since S/MIME certificates are credentials for end users. For the other use cases, this section is optional since the other use cases are for devices and don't uniquely identify end users.
	4.1	Certificate Application		M	O	O	O	When this section is included, it should clearly state who is able to request a certificate (the end user directly or an administrator acting on behalf of the end user).
		4.1.1	Who can submit a certificate application					
		4.1.2	Enrollment process and responsibilities					
	4.2	Certificate application processing		M	O	O	O	The processes in this portion of the CP/CPS are very important in that they outline how the originator of the request is authenticated as well as how the subject of the certificate is authenticated. Much of the underlying security assumptions of the PKI depend on these processes being executed securely.
		4.2.1	Performing identification and authentication functions					

		4.2.2	Approval or rejection of certificate applications					
		4.2.3	Time to process certificate applications					
	4.3	Certificate issuance		M	O	O	O	This section should state how the certificate is actually issued to subscribers and include any process steps that users must execute to retrieve their certificate. Mainly applicable to S/MIME PKI use case.
		4.3.1	CA actions during certificate issuance					
		4.3.2	Notification to subscriber by the CA of issuance of certificate					
	4.4	Certificate acceptance		M	O	O	O	This section should list out any special steps used by subscribers to indicate that they have received their certificate. This section is usually very simple as certificate acceptance may be implied as opposed to requiring a positive action by the user.
		4.4.1	Conduct constituting certificate acceptance					
		4.4.2	Publication of the certificate by the CA					
		4.4.3	Notification of certificate issuance by the CA to other entities					

	4.5	Key pair and certificate usage		M	O	O	O	This section must be present in a CP/CPS as it clearly states how users/systems may use certificates issued by the PKI. This section permits the PKI administrator to demonstrate that the certificates are intended for a specific use in the event a dishonest subscriber uses their certificate in an unapproved fashion.
		4.5.1	Subscriber private key and certificate usage					
		4.5.2	Relying party public key and certificate usage					
	4.6	Certificate renewal		M	O	O	O	This section also must be present in a CP/CPS as it defines how an expiring certificate is renewed. Key elements of this section include: who is permitted to request certificate renewals and what circumstances are permitted for a certificate to be re-issued. The other sections can be cross-referenced to the processes in sections 4.1 - 4.5
		4.6.1	Circumstance for certificate renewal					
		4.6.2	Who may request renewal					
		4.6.3	Processing certificate renewal requests					
		4.6.4	Notification of new certificate issuance to subscriber					
		4.6.5	Conduct constituting acceptance of a renewal					

			certificate					
		4.6.6	Publication of the renewal certificate by the CA					
		4.6.7	Notification of certificate issuance by the CA to other					
	4.7	Certificate re-key		M	O	O	O	This section must be present in a CP/CPS as it documents the circumstances in which a new key pair may be generated for a subscriber. Note that it is acceptable for the CP/CPS to simply use the certificate renewal procedure for certificate re-key for the HTTPS, 802.1x, and IPSec PKI use cases. For S/MIME certificates, an explicit re-key process may be required for certain cases such as key compromise.
		4.7.1	Circumstance for certificate re-key					
		4.7.2	Who may request certification of a new public key					
		4.7.3	Processing certificate re-keying requests					
		4.7.4	Notification of new certificate issuance to subscriber					
		4.7.5	Conduct constituting acceptance of a re-keyed certificate					
		4.7.6	Publication of the re-keyed certificate by the CA					
		4.7.7	Notification of certificate issuance by the CA to other					

	4.8	Certificate modification		M	O	O	O	This section must be present in a CP/CPS as it documents the process by which a new certificate with different contents may be issued to a subscriber. It should list out what specific fields in a certificate may be changed via the modification process. Beyond this, the certificate modification is a repeat of the certificate issuance steps 4.1 - 4.5.
		4.8.1	Circumstance for certificate modification					
		4.8.2	Who may request certificate modification					
		4.8.3	Processing certificate modification requests					
		4.8.4	Notification of new certificate issuance to subscriber					
		4.8.5	Conduct constituting acceptance of modified certificate					
		4.8.6	Publication of the modified certificate by the CA					
		4.8.7	Notification of certificate issuance by the CA to other					
	4.9	Certificate revocation and suspension		M	O	O	O	This section of the CP/CPS is another critical portion of the CP/CPS as it documents the circumstances under which a certificate may be revoked. Key parts of this section include the circumstances for revocation, who can request revocation, and publication of revocation. The same sections are also appropriate for certificate suspension (suspension is different from revocation

								in that it is temporary, meaning that a certificate that has been suspended is expected to return to service at some point in time).
		4.9.1	Circumstances for revocation					
		4.9.2	Who can request revocation					
		4.9.3	Procedure for revocation request					
		4.9.4	Revocation request grace period					
		4.9.5	Time within which CA must process the revocation request					
		4.9.6	Revocation checking requirement for relying parties					
		4.9.7	CRL issuance frequency (if applicable)					
		4.9.8	Maximum latency for CRLs (if applicable)					
		4.9.9	On-line revocation/status checking availability					
		4.9.10	On-line revocation checking requirements					
		4.9.11	Other forms of revocation advertisements available					
		4.9.12	Special requirements re key compromise					
		4.9.13	Circumstances for suspension					
		4.9.14	Who can request suspension					
		4.9.15	Procedure for suspension request					
		4.9.16	Limits on suspension period					

	4.10	Certificate status services		M	O	O	O	This section documents how the PKI publishes updates on the status of certificates (revoked certificates via CRLs or OCSP). It is best practice for all PKI use cases to provide online certificate status services, but is especially critical for S/MIME.
		4.10.1	Operational characteristics					
		4.10.2	Service availability					
		4.10.3	Optional features					
	4.11	End of subscription		M	O	O	O	This section documents procedures to be followed by the PKI administrator to end services for subscribers. This section should be present for S/MIME PKI use cases.
	4.12	Key escrow and recovery		M	O	O	O	This section is another absolutely critical portion of the CP/CPS. Processes used to gain access to subscriber private keys are defined here as well as the requirements that must be met for such requests to be granted. The overall security of the PKI rests heavily on the controls implemented in the PKI to ensure that private key recovery is carefully controlled.
		4.12.1	Key escrow and recovery policy and practices					
		4.12.2	Session key encapsulation and recovery policy and practices					
5	FACILITY, MANAGEMENT, AND OPERATIONAL CONTROLS (11)							As in Sections 3 & 4, the overall controls

				M	O	O	O	governing the PKI operation are more stringent for S/MIME certificates.
	5.1	Physical controls		M	O	O	O	
		5.1.1	Site location and construction	M	O	O	O	
		5.1.2	Physical access	M	O	O	O	
		5.1.3	Power and air conditioning	M	O	O	O	
		5.1.4	Water exposures	M	O	O	O	
		5.1.5	Fire prevention and protection	M	O	O	O	
		5.1.6	Media storage	M	O	O	O	
		5.1.7	Waste disposal	M	O	O	O	
		5.1.8	Off-site backup	M	O	O	O	
	5.2	Procedural controls		M	O	O	O	
		5.2.1	Trusted roles	M	O	O	O	
		5.2.2	Number of persons required per task	M	O	O	O	
		5.2.3	Identification and authentication for each role	M	O	O	O	
		5.2.4	Roles requiring separation of duties	M	O	O	O	
	5.3	Personnel controls		M	O	O	O	
		5.3.1	Qualifications, experience, and clearance requirements	M	O	O	O	
		5.3.2	Background check procedures	M	O	O	O	
		5.3.3	Training requirements	M	O	O	O	
		5.3.4	Retraining frequency and requirements	M	O	O	O	
		5.3.5	Job rotation frequency and sequence	M	O	O	O	
		5.3.6	Sanctions for unauthorized actions	M	O	O	O	
		5.3.7	Independent contractor requirements	M	O	O	O	
		5.3.8	Documentation supplied to personnel	M	O	O	O	
	5.4	Audit logging procedures		M	O	O	O	

		5.4.1	Types of events recorded	M	O	O	O	
		5.4.2	Frequency of processing log	M	O	O	O	
		5.4.3	Retention period for audit log	M	O	O	O	
		5.4.4	Protection of audit log	M	O	O	O	
		5.4.5	Audit log backup procedures	M	O	O	O	
		5.4.6	Audit collection system (internal vs. external)	M	O	O	O	
		5.4.7	Notification to event-causing subject	M	O	O	O	
		5.4.8	Vulnerability assessments	M	O	O	O	
	5.5	Records archival		M	O	O	O	
		5.5.1	Types of records archived	M	O	O	O	
		5.5.2	Retention period for archive	M	O	O	O	
		5.5.3	Protection of archive	M	O	O	O	
		5.5.4	Archive backup procedures	M	O	O	O	
		5.5.5	Requirements for time-stamping of records	M	O	O	O	
		5.5.6	Archive collection system (internal or external)	M	O	O	O	
		5.5.7	Procedures to obtain and verify archive information	M	O	O	O	
	5.6	Key changeover		M	O	O	O	
	5.7	Compromise and disaster recovery		M	O	O	O	
		5.7.1	Incident and compromise handling procedures	M	O	O	O	
		5.7.2	Computing resources, software, and/or data are corrupted	M	O	O	O	
		5.7.3	Entity private key compromise procedures	M	O	O	O	
		5.7.4	Business continuity capabilities after a disaster	M	O	O	O	
	5.8	CA or RA termination		M	O	O	O	
6	TECHNICAL SECURITY CONTROLS (11)							Section 6 must be completed fully for all enterprise PKI use cases. The reasoning

							is that the underlying technical security controls (in addition to the Identification and Authentication policies/procedures) provide a great deal of the security properties of X.509 certificates.
	6.1	Key pair generation and installation		M	M	M	M	This section must document the cryptographic and hash functions that are used in certificates issued by the PKI as well as the key sizes used for certificates, both root and subscriber certificates. Additionally, this section should document the processes and technical controls used for key pair generation.
		6.1.1	Key pair generation	M	M	M	M	
		6.1.2	Private key delivery to subscriber	M	M	M	M	
		6.1.3	Public key delivery to certificate issuer	M	M	M	M	
		6.1.4	CA public key delivery to relying parties	M	M	M	M	
		6.1.5	Key sizes	M	M	M	M	
		6.1.6	Public key parameters generation and quality checking	M	M	M	M	
		6.1.7	Key usage purposes (as per M.509 v3 key usage field)	M	M	M	M	
	6.2	Private Key Protection and Cryptographic Module Engineering		M	M	M	M	All subsections of this section must be completed for all enterprise PKI use cases. Protection of the root private key is paramount to ensuring that unauthorized personnel are not able to issue or revoke certificates in the CA's name.

		6.2.1	Cryptographic module standards and controls	M	M	M	M	
		6.2.2	Private key (n out of m) multi-person control	M	M	M	M	
		6.2.3	Private key escrow	M	M	M	M	
		6.2.4	Private key backup	M	M	M	M	
		6.2.5	Private key archival	M	M	M	M	
		6.2.6	Private key transfer into or from a cryptographic module	M	M	M	M	
		6.2.7	Private key storage on cryptographic module	M	M	M	M	
		6.2.8	Method of activating private key	M	M	M	M	
		6.2.9	Method of deactivating private key	M	M	M	M	
		6.2.10	Method of destroying private key	M	M	M	M	
		6.2.11	Cryptographic Module Rating	M	M	M	M	
	6.3	Other aspects of key pair management		M	O	O	O	This section is primarily relevant to S/MIME certificates since public key archival processes are required if digital signatures applied to emails must be verified after the certificate itself has expired. For the other enterprise PKI use cases this is less important.
		6.3.1	Public key archival	M	O	O	O	
		6.3.2	Certificate operational periods and key pair usage periods	M	M	M	M	

	6.4	Activation data		M	M	M	M	Activation of the private key corresponding to a certificate should be addressed in the S/MIME and HTTPS enterprise use cases, since for S/MIME certificates the activation process provides a final check to ensure that the person who the private key is generated for is the person who is activating and using the key. For HTTPS enterprise use cases, this section should describe how the private key is activated when the web server is started (depending on the specific type of web server in use, a private key password may need to be entered when the system is restarted or the private key may not be passphrase encrypted on these servers.
		6.4.1	Activation data generation and installation	M	M	M	M	
		6.4.2	Activation data protection	M	M	M	M	
		6.4.3	Other aspects of activation data	M	M	M	M	
	6.5	Computer security controls		M	M	M	M	This section should document the security controls that are implemented on the systems that compromise the PKI (CA server and any supporting systems). This section is required for all enterprise PKI use cases. References to well known computer security benchmarks (NIST SP800-53 and related guidance, DoD standards, vendor security benchmarks, CIS guidelines etc)
		6.5.1	Specific computer security technical	M	M	M	M	

			requirements					
		6.5.2	Computer security rating	M	M	M	M	
	6.6	Life cycle technical controls		M	M	M	M	This section is required for all enterprise PKI use cases. It contains details describing how changes to the hardware/software are managed and covers change control processes used to approve new software/hardware deployed in the PKI.
		6.6.1	System development controls	M	M	M	M	
		6.6.2	Security management controls	M	M	M	M	
		6.6.3	Life cycle security controls	M	M	M	M	
	6.7	Network security controls		M	M	M	M	This section is required for all enterprise PKI use cases. It describes how the network supporting the PKI is protected from the rest of the enterprise environment. In general, the PKI network should be segmented from the rest of the enterprise network. The CP/CPS should state how this segmentation is implemented.
	6.8	Time-stamping		M	M	M	M	This section is required for all enterprise PKI use cases. The accuracy of timestamps used by the CA is critical since accurate time enables the CA to create timely CRLs and accurate certificate lifetimes. Look to see that the CA is using a secure time synchronization protocol such as ntp with authentication enabled.

7	CERTIFICATE, CRL, AND OCSP PROFILES							As in Section 6, the certificate profiles as well as the revocation mechanisms must be fully documented for the PKI to provide the security services it is intended to provide. Refer to the enterprise PKI use case certificate profiles to verify that those profiles are documented in the CP/CPS.
	7.1	Certificate profile		M	M	M	M	
		7.1.1	Version number(s)	M	M	M	M	
		7.1.2	Certificate extensions	M	M	M	M	
		7.1.3	Algorithm object identifiers	M	M	M	M	
		7.1.4	Name forms	M	M	M	M	
		7.1.5	Name constraints	M	M	M	M	
		7.1.6	Certificate policy object identifier	M	M	M	M	
		7.1.7	Usage of Policy Constraints extension	M	M	M	M	
		7.1.8	Policy qualifiers syntax and semantics	M	M	M	M	
		7.1.9	Processing semantics for the critical Certificate Policies	M	M	M	M	
	7.2	CRL profile		M	M	M	M	
		7.2.1	Version number(s)	M	M	M	M	
		7.2.2	CRL and CRL entry extensions	M	M	M	M	
	7.3	OCSP profile		M	M	M	M	
		7.3.1	Version number(s)					
		7.3.2	OCSP extensions					
8	COMPLIANCE AUDIT AND OTHER ASSESSMENTS							While all enterprise PKI use cases should be subjected to audits as a security best

							practice, a S/MIME PKI absolutely must be audited to ensure that the PKI is being operated in a secure fashion.
	8.1	Frequency or circumstances of assessment		M	O	O	O	
	8.2	Identity/qualifications of assessor		M	O	O	O	
	8.3	Assessor's relationship to assessed entity		M	O	O	O	
	8.4	Topics covered by assessment		M	O	O	O	
	8.5	Actions taken as a result of deficiency		M	O	O	O	
	8.6	Communication of results		M	O	O	O	
9	OTHER BUSINESS AND LEGAL MATTERS							This section is optional for all enterprise PKI use cases with the exception of the clauses marked for S/MIME due to the personal information that may be stored/processed as part of S/MIME certificate issuing processes. This section may also be applicable in cases where the PKI function is funded by other business units within the enterprise.
	9.1	Fees		O	O	O	O	
		9.1.1	Certificate issuance or renewal fees	O	O	O	O	
		9.1.2	Certificate access fees	O	O	O	O	
		9.1.3	Revocation or status information access fees	O	O	O	O	
		9.1.4	Fees for other services	O	O	O	O	
		9.1.5	Refund policy	O	O	O	O	
	9.2	Financial responsibility		O	O	O	O	
		9.2.1	Insurance coverage	O	O	O	O	
		9.2.2	Other assets	O	O	O	O	
		9.2.3	Insurance or warranty coverage for end-	O	O	O	O	

			entities					
	9.3	Confidentiality of business information		O	O	O	O	
		9.3.1	Scope of confidential information	O	O	O	O	
		9.3.2	Information not within the scope of confidential information	O	O	O	O	
		9.3.3	Responsibility to protect confidential information	O	O	O	O	
	9.4	Privacy of personal information		O	O	O	O	
		9.4.1	Privacy plan	M	O	O	O	
		9.4.2	Information treated as private	M	O	O	O	
		9.4.3	Information not deemed private	M	O	O	O	
		9.4.4	Responsibility to protect private information	M	O	O	O	
		9.4.5	Notice and consent to use private information	M	O	O	O	
		9.4.6	Disclosure pursuant to judicial or administrative process	M	O	O	O	
		9.4.7	Other information disclosure circumstances	M	O	O	O	
	9.5	Intellectual property rights		O	O	O	O	
	9.6	Representations and warranties		O	O	O	O	
		9.6.1	CA representations and warranties	O	O	O	O	
		9.6.2	RA representations and warranties	O	O	O	O	
		9.6.3	Subscriber representations and warranties	O	O	O	O	
		9.6.4	Relying party representations and warranties	O	O	O	O	
		9.6.5	Representations and warranties of other participants	O	O	O	O	
	9.7	Disclaimers of warranties		O	O	O	O	
	9.8	Limitations of liability		O	O	O	O	
	9.9	Indemnities		O	O	O	O	

	9.1	Term and termination		O	O	O	O	
		9.10.1	Term	O	O	O	O	
		9.10.2	Termination	O	O	O	O	
		9.10.3	Effect of termination and survival	O	O	O	O	
	9.11	Individual notices and communications with participants		O	O	O	O	
	9.12	Amendments		O	O	O	O	
		9.12.1	Procedure for amendment	O	O	O	O	
		9.12.2	Notification mechanism and period	O	O	O	O	
		9.12.3	Circumstances under which OID must be changed	O	O	O	O	
	9.13	Dispute resolution provisions		O	O	O	O	
	9.14	Governing law		O	O	O	O	
	9.15	Compliance with applicable law		O	O	O	O	
	9.16	Miscellaneous provisions		O	O	O	O	
		9.16.1	Entire agreement	O	O	O	O	
		9.16.2	Assignment	O	O	O	O	
		9.16.3	Severability	O	O	O	O	
		9.16.4	Enforcement (attorneys' fees and waiver of rights)	O	O	O	O	
		9.16.5	Force Majeure	O	O	O	O	
	9.17	Other provisions		O	O	O	O	

Appendix B. Code Listing

The source for the cert2xml.pl script that is used in the certificate profile assessment process is listed below. Note that the listing has additional line breaks embedded for formatting purposes.

```
#!/usr/bin/perl -w
use strict;
use Data::Dumper;
use XML::Simple;

sub trim($);


my $incert = $ARGV[0];
my %certfields = (
      extensions => 0,
      keylen => 0,
      startDate => 0,
      endDate => 0,
      version => 0,
      signatureAlgorithm => 0,
      isCA => 0,
      subjectStd => 0,
      subjectnStd => 0,
      issuerStd => 0,
      issuernStd => 0,
      publicKeyAlgorithm => 0
);


# Get the set of extensions from the certificate
      my $exts = `openssl x509 -in $incert -text -noout -certopt
"no_header,no_sigdump,no_version,no_subject,no_issuer,no_validity,no_se
rial,no_pubkey,no_signame"`;
$certfields{extensions} = $exts;
```

```
# Get the signature algorithm
my $alg = `openssl x509 -in $incert -text | grep "Signature Algorithm"`;
$alg = trim($alg);
# Get rid of the duplicate text by doing a multi-line regex match
$alg =~ m/^.*:(.*)\s+.*/m;
$alg = $1;
$certfields{signatureAlgorithm} = $alg;

# Get the public key algorithm
my $pubkeyalg = `openssl x509 -in $incert -text | grep "Public Key Algorithm"`;
$pubkeyalg =~m/^.*:(.*)$/;
$pubkeyalg = trim($1);
$certfields{publicKeyAlgorithm} = $pubkeyalg;


# Get the key length
my $keylen = `openssl x509 -in $incert -text | grep "Modulus"`;
$keylen = trim($keylen);
$keylen =~ m/^.*\((.*)\)/;
$certfields{keylen} = $1;


# get the startdate
my $start = `openssl x509 -in $incert -startdate`;
$start =~ s/--.*--$//s;
$start =~ m/^notBefore=(.*)/;
$start = trim($1);
$certfields{startDate} = $start;

# get the enddate
my $end = `openssl x509 -in $incert -enddate`;
$end =~ s/--.*--$//s;
$end =~ m/^notAfter=(.*)/;
$end = trim($1);
$certfields{endDate} = $end;

# get the certificate version
my $version = `openssl x509 -in $incert -text | grep -i version`;
```

```
$version =~ m/^.*:(.*)$/;
$certfields{version} = $1;

# check to see if a certificate is a CA cert
my $isCA = 0;
$isCA = `openssl x509 -in $incert -text | grep "CA:TRUE"`;
$isCA = trim($isCA);
if(length($isCA) > 0)
{
      $isCA = "TRUE";
}
else
{
      $isCA = "FALSE";
}

$certfields{isCA} = $isCA;

# get the certificate subject
my $subject = `openssl x509 -in $incert -nameopt sep_semi_plus_space \
-subject -noout`;
# get rid of the 'subject=' preceding the openssl output
$subject =~ s/^subject=\s+//;
$certfields{subjectStd} = $subject;

# get the certificate issuer
my $issuer = `openssl x509 -in $incert -nameopt sep_semi_plus_space \
-issuer -noout`;
# get rid of the 'issuer=' preceding the openssl output
$issuer =~ s/^issuer=\s+//;
$certfields{issuerStd} = $issuer;

printCert(\%certfields);

# subroutine to print the fields from the certificate profiles
# in the order they are presented in the profile tables
sub printCert
{
```

```
        my %cert = %{$_[0]};
        my $xs = XML::Simple->new(ForceArray => 1);
        my $xml = $xs->XMLout(\%cert);
        print $xml;
}


# subroutine to trim whitespace from front and back of a string

sub trim($)
{
        my $string = shift;
        $string =~ s/^\s+//;
        $string =~ s/\s+$//;
        return $string;
}
```