International
Carpathian Control
Conference ICCC' 2002
MALENOVICE,
CZECH REPUBLIC
May 27-30, 2002

IMPLEMENTATION ISSUES OF PKI TECHNOLOGY

Victor-Valeriu PATRICIU, Marin BICA and Ion BICA
Department of Computer Engineering,
MTA – Military Technical Academy of Bucharest,
Bucharest, Romania, vip@mta.ro

Abstract: Today, one of the biggest concerns about using the Internet for business-critical data is security. This paper will concentrate on the area of software security based on public key cryptographic technology. The Public Key systems make it possible for two parties to communicate securely without either having to know or trust the other party. This is possible because a third party, called the Certification Authority, that both the other parties trust identifies them, and certifies that their keys are genuine. This third party guarantees that they are who they claim to be. A public key infrastructure (PKI) is a set of technologies and security policies that a company can use to issue, revoke, and manage digital certificates within its organizational structure. The paper tries to analyse some of major deployment aspects of an organizational PKI and the main design issues for a Public Key Infrastructure (PKI), needed to secure network applications.

Key words: PKI technology, Public Key Cryptography, Certificate Authority, PKI deployment.

1 Public Key Cryptography and Information Security

Typically cryptographic functions require *keys* which are used to *encrypt* and *decrypt* the data and are known only by trusted entities. There are two commonly known flavours of key-based cryptography, known as symmetric key, and asymmetric key. As the names suggest, symmetric key cryptography uses the same key to encrypt and decrypt data, while asymmetric key cryptography uses two keys which are mutual inverses (one decrypts the other's encryption). Asymmetric cryptography is known as *public key cryptography*, because one half of the key pair can be published without compromising the overall security of the system.

Public keys may be maintained in a database, with associated subject identity and other information, each record is known as a certificate. *Certificates*, however, do not by themselves enhance the trust in the system as a trusted third party is still required to create

the certificates and prove trust. The trusted third party is know as the Certification Authority (CA) and it enables trust using a public key technique known as digital signatures.

Digital (electronic) signatures are a technique which uses an entity's private key to encrypt a digest calculated on a message. This enables other entities to verify the signature by decrypting the signature using the signing entity's public key and comparing with a message digest calculated on the message locally. Digital signatures prove integrity of the message, and authenticity if the public key can be trusted to be authentic (Figure 1).

Since the CA's public key must be well known by all users, and since it is a cryptographically strong key, the CA can provide trust within the system by signing all certificates it issues. As long as the signing CA's public key for any certificate is known and trusted, the certificate can be used as proof of a binding between a client and a particular public key.

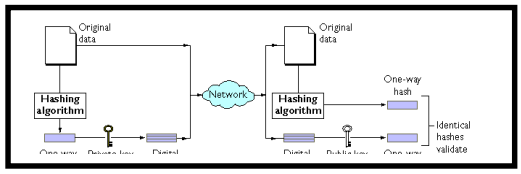

Figure 1. Digital signature process

The key management problem is now apparent. How are certificates passed around a public key system, and how is the CA's public key, which is used to verify certificates, broadcast/published securely? These problems can be resolved by the provision of a *Public Key Infrastructure (PKI)* which supports certification and broadcast of certificates using a simple architecture. It is a desirable feature of this PKI architecture that it be easily scalable to enable support for very large network environments such as the Internet.

2 PKI Architecture

The basic architecture of the PKI consists of 3 main servers: the *Certification Authority (CA), Certification Server (CS)* and *Certificate Revocation Server (CRS)* (Figure 2). These main servers form the basis of the PKI structure conceptually known as a *domain*. The interdomain interface is provided to clients and the other PKI servers by the CS, which communicates with other domain CSs to request or send interdomain information. There are 2 types of CA's: *organisational* and *public*.

PKI are based on X.509 version 3 certificates. On top of the architectural details, a client Application Program Interface (API) is provided, with C bindings, for development of applications which may want to utilise the PKI and public-key cryptography for security services.

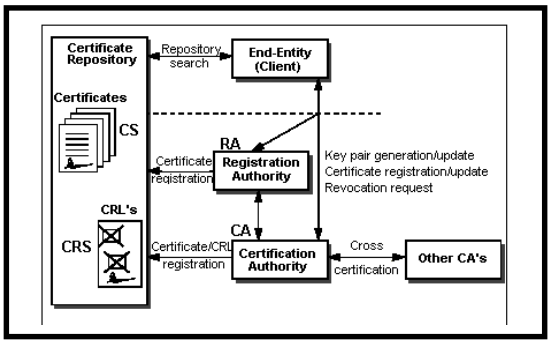

Figure 2. PKI entities

There are a wide variety of issues presented when designing and implementing PKI ranging from common issues encountered in all areas such as language choice and design methodology through to specifics such as interdomain hierarchy structures and revocation techniques. We will outline the existing problem and difficulty issues about PKI implementation.

3 PKI Structure

The structure of any PKI requires at least *2 functional blocks*:

- For creating and revoking certificates (CA);
- For storing certificates and CRLs(CS-repository).

Because the trust in a PKI system resides within the certificates themselves, the CA must be a trusted entity, but no such requirement need be placed on the CS. The CA must reside within a protected zone of the site, and be maintained by a trusted administrator. Two interfaces to the CA are provided to clients, one to *create certificates*, and one to *revoke certificates*. The CA has no other interfaces to external clients. Certificates, and Certificate Revocation Lists (CRLs) are registered with the CS by the CA or a special „gichet“, called *Registration Authority*.

The CS receives Certificates and CRLs from the CA and stores these items in the corresponding database. The CS provides several other interfaces to clients within the local domain as well as an interdomain interface. Clients may contact the CS requesting certificates by subject name or serial number, they may also request CRLs from the CRS interface. Interdomain clients may access the same facilities through the local CS. The CS may reside anywhere within the installation and need not be trusted as it merely stores certificates in which the trust is inherent.

The provision of services from separate server entities facilitates the partitioning of the larger network into localised domains. Smaller domains enable local security management and reduce the administration overhead to achievable levels. Each domain would maintain its own internal PKI structure of CA and CS/CRS servers and certified end-entities (clients). Interdomain trust however, becomes somewhat more difficult to resolve. In order for 2 clients to establish trust, each must be able to retrieve and verify the other's certificate. This process requires that each client can obtain a trusted copy of the public key for the CA which certified the other client. If the clients reside in different domains then the CAs of each domain must be certified within the local domain, or some path of CA/domain certification must exist between the client domains. This process is known as cross certification and requires off-line communication between administrato#rs of each domain to certify each CA in the other's domain.

4 Certificate Chains

The PKI provides the facility to partition the world into localised security domains. Domains are typically localised within organisation boundaries, encompassing the trust region of that organisation only. Domains may contain from one to many hosts and many clients, though the purpose of domains is to localise a manageable number of clients and machines within the scope of a security administrator.

The simplest PKI system is comprised of only a single domain, within which all entities exist. This simplifies certificate validation issues such as locating the CA public key, but cannot be used in the real world as the sheer size of the domain would be unmanageable and very few, if any, entities are willing to share trust in a single CA. Splitting the world into domains creates other problems. These problems can be resolved by establish trust, via cross certification, with other domains. Each domain may opt to use its own ad hoc routing method or a global hierarchy can be established within which every domain resides, cross certified with at least one other domain (see Figure 3).

Certificate retrieval is an important issue in PKI systems as verification of certificates requires public key decryption which can be a potentially high-cost operation. It is therefore desirable to optimise the certificate retrieval process to reduce waiting times and overheads. Retrieval of a certificate from within the local domain is a straightforward operation. The client contacts the CS outlining the details, either subject name or serial number, of the desired certificate. The CS checks the request, if valid it searches the certificate database and returns the certificate, if found. The client must then verify the certificate for authenticity using the CA's public key to verify the digital signature on the certificate. This is the expensive operation, decrypting the signature using the public key.

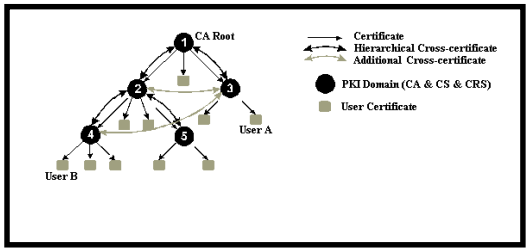

Figure 3. Certificate hierarchy

A subsequent check on a certificate with a valid signature is to check whether it has been revoked. The client must contact the CRS requesting a copy of the latest CRL. Once it receives a copy of the CRL the client must verify the signature on the CRL using the CA's public key. If the CRL is valid then the client searches for the certificate serial number in the CRL. If the serial number is found then the certificate has been revoked and may only be usable for decrypting data encrypted by the certificate owner before the certificate was revoked.

The case of *interdomain certificate retrieval* is more complicated and depends heavily on the domain interconnection strategy adopted within the PKI. To retrieve a certificate from a remote domain requires that the CAs of the local and remote domains are cross-certified or there is a path of cross-certified domains linking the local and remote domain's CAs. Retrieving a certificate from a remote domain yields a chain of certificates corresponding to the path of cross-certification of CAs/domains through the hierarchy and the desired certificate. Validating the certificate chain requires that each certificate's digital signature is verified with the issuing CA's public key. This process begins with the first certificate in the chain, which is generated by the local CA, yielding the public key of the issuing CA of the next certificate in the chain. The process continues, validating certificates in the chain using previous certificates in the chain, until the final certificate can be verified.

Cross certification performs two essential operations within the PKI domain hierarchy (Figure 3). Firstly it propagates trust between domains, enabling interdomain

communications secured using PKI services. Secondly, it enables short-cuts through the domain hierarchy, speeding up certificate retrieval between any two domains by reducing hop-distance between the two domains through the hierarchy to a single hop.

5 Certificate Revocation

Once created a certificate remains valid for the duration between its inception/creation and expiry times. However, there are situations in which the certificate may be compromised either through owner carelessness, system compromise or mandatory retirement of the certificate. In these circumstances a facility is required to enable cancellation of certificates. This process is known as *revocation*. Only the owner of the certificate or the administrator of the CA should have the authorisation to revoke a certificate.

Revocation information must be publicly available, exactly as for certificates. This information is typically represented as a list, signed by the CA, each entry detailing a revoked certificate. Information contained in each list entry includes: certificate serial number, revoker identity, time and date of revocation and reason for revocation. This list is known as a *Certificate Revocation List (CRL*), is issued by the CA and is made available to clients via the CRS. The CRL is maintained by the CA and as each revocation occurs, and entry is added to the list. As the CA's domain ages the CRL grows, listing all revoked certificates. It is obvious that as time goes by the CRL may become quite large, especially in a domain with a large client base and high turn over of certificates. As a mechanism, CRLs are the most obvious solution to the problem of storing and relaying revocation information, however obesity of CRLs needs to be addressed.

Delta CRLs have been proposed as a solution to the CRL problem. In this idea CRLs are issued over short time-frames, with each CRL adding to the revocation database comprised of all previous CRLs. Hence each CRL issued is effectively a "delta" or update of previous revocation information. Delta CRLs are open to man-in-the-middle attacks. In the case of CRLs this is simple as the whole CRL is provided and revocation status is determined by existence of a matching revocation entry.

6 PKI Deployment Steps

Setting up a PKI that suits the security goals involves making numerous decisions before installing any software. To help an organization in this decision making, it must follows some steps for a *PKI* deployment.

- *Start & Planning the Project* are the following components: Project planning; Engaging sponsors and project leaders; Seizing the initial project; Developing and documenting a project management plan.
- *Requirements Analysis and PKI Design* focuses on: analyzing, designing and documenting Certificate Policy and Certification Practice Statements, documenting PKI system requirements and design, documenting PKI facility needs, identifying staff and training needs, procuring hardware and software.
- *PKI Components Development & Testing* by collecting metrics (usability, administration load, system loading, etc.). Other aspects of Development and Testing involve: assessing facilities for enhancements, training PKI staff.

- *Pilot Installation & Deployment* involves the amalgamation of all the PKI components, thus building a pilot system against which all the functional, performance and operational requirements can be tested. Active deployment involves: engaging the pilot user community, running the pilot for four to six weeks, rollout to the enterprise incrementally not require any client software to be installed or configured on your users' systems. *User creation* is basically a three-step process, which entails user initialization, shared secret distribution and user registration. Administrators must initialize each user in PKI. Once the administrators initialize the users, they assign a shared secret. In the standard PKI environment this shared secret consists of a reference number and an authorization code. The hurdle to overcome with the shared secret is efficient distribution to large end user populations. Alternatives for shared secret distribution include envelopes from a blind printer, Web site distribution, and interactive voice recognition system. Finally, the client will generate its own signing key pair (in support of non-repudiation) and sends the verification public key up to the PKI. The PKI generates the user's encryption key pair (in support of key backup) along with the encryption and verification certificates.

7 Conclusion

Like most IT implementations, PKI deployment requires planning and additional work up-front. However, some PKI software (Entrust, Keon RSA, etc.) provides easy administration and management of secure business applications. As a result, you only have to administer security once for all business applications and the end user will only have to remember one password for all applications. For organizations, the PKI capability may adopt the following principles: use commercial products, use smart cards for protection of cryptography, digital signature, access control, keys and certificates. The proposed structure consist of CA, CS and CRS, where the CA is in a restricted area to minimise the risk of compromise. Off-line certification and secured communications channels also add to the security of the servers within the domain unit. Interdomain security is maintained through limited access via the CS and cross-certification between the CAs of trusting domains

This design and implementation identified a number of outstanding and quite difficult issues which impact upon both the design and implementation of an infrastructure for public key technology. Interdomain hierarchy, certificate retrieval and trust are problems which, although theoretically solved, remain to be proven in practice. We realize that PKI deployment is more than deploying a technology; it is a new way of doing trusted e-business.

References

1. Housley R., Polk T., *Planning for PKI,* John Wiley, 2001.
2. Housley R., Ford W. and Solo D., "Internet Public Key Infrastructure", PKIX Working Group, June 1996.
3. Ford W., *Computer Communications Security*, Prentice Hall, New Jersey, 1994.
4. Ford W., Baum M. , *Secure Electronic Commerce – Building Infrastruture for Digital Signatues and Encrypti*on, Prentice Hall, 1997.
5. Patriciu,V.V., Pietrosanu M., Bica I., Voicu N., Vaduva C, *Securitatea comertului electronic*, Ed All, Bucureşti, 2001.