Wireless Security Perspectives

Cellular Networking Perspectives

Editors: Les Owens, David Crowe. Email: wsp@cnp-wireless.com Vol. 3, No. 5. June, 2001

An Overview of Public Key Infrastructure (PKI)

Michael Crerar

Cryptography has been around since antiquity as a means to keep confidential messages confidential. In fact, until relatively recent times, cryptography has been used solely for privacy. That is, the science of secret writing – classical cryptography, as shown in Figure 1 – only offered the capability to hide information from prying eyes. Today, however, cryptography is a very powerful tool that can be used for data integrity (preventing data from unauthorized modification), user authentication ("Are you who you say you are?"), non-repudiation (proof that a message was sent or received), as well as confidentiality (protection from eavesdropping). Also, until recent times, cryptography has been hampered by key management problems.

With the introduction and deployment of PKI (Public Key Infrastructure), scalable and very secure systems – providing all the services mentioned above – are possible. PKI, through the use of public-key cryptography, allows the re-creation of secure electronic equivalents of traditional paper-based commerce. Pundits believe that PKI is becoming an enabling technology for business-to-business e-commerce. Moreover, the total market for the technology is expected to reach $3 Billion by 2004. Many companies have allocated extensive budgets to explore the benefits of PKI, and several have wagered their entire futures on developing effective and scalable PKI solutions to satisfy this perceived need. Many believe PKI is essential to the success of e-commerce and electronic business-to-business transactions.

A PKI must supply:

- products to generate, store and manage cryptographic keys;
- procedures to dictate how the keys and certificates should be generated, distributed and used; and
- policies to define the rules under which the cryptographic system should operate.

All PKI systems must provide the following two things:

- Certification: The process that binds a public-key to an individual, organization, or something else (for example, a credential)
- Validation: The process that verifies that a certificate is still valid.

PKI systems require the implementation of new network components:

- Certificate Authority (CA): Issues and revokes certificates and is ultimately responsible for their authenticity.
- Registration Authority (RA): Verifies identity and registration information.
- Directory: Stores certificates in a central location
- Certificate revocation list (CRL): Data and directory structure for publishing certificates that have been revoked.

About Wireless Security Perspectives

Price

The basic subscription price for Wireless Security Perspectives is $300 for one year (12 issues) for delivery by emailed PDF file or by first class mail within the US or Canada. International subscriptions are US$350 per year. The basic subscription allows for distribution to up to 10 people within one organization. Contact us for license fees to allow more readers.

We provide discounts to educational institutions and small businesses (less than 10 employees).

Back issues are available individually, or in bulk at reduced prices.

Delivery is by first class mail or by email (Acrobat PDF file).

Complete pricing information for both publications is available at:

www.cnp-wireless.com/prices.html

To obtain a subscription, please contact us at:

cnpaccts@cnp-wireless.com

Next Issue Due…

July 16th, 2001.

Future Topics

Wireless Packet Data Security • AES (Rijndael) • m-commerce security • IP Security • Public Keys & Wireless • IP Mobility security • Security issues in ad hoc wireless networks • Electronic Signatures in Wireless • Latest in Watermarking

Wireless Security Perspectives (ISSN 1492-806X (print) and 1492-8078 (email)) is published monthly by Cellular Networking Perspectives Ltd., 2636 Toronto Cresc. NW, Calgary, Alberta T2N 3W1, Canada. **Phone**: 1-800-633-5514 (+1-403-274-4749) **Fax**: +1-403-289-6658 **Email**: cnpsales@cnp-wireless.com **Web**: www.cnp-wireless.com/wsp.html. **Subscriptions**: $300 for delivery in the USA or Canada, US$350 elsewhere. **Payment**: accepted by cheque, bank transfer, American Express, Diners Club, MasterCard or Visa. **Delivery**: Email or 1st class mail. **Back Issues**: Available individually for $35 in the US and Canada and US$40 elsewhere, or in bulk at reduced rates. **Discounts**: Educational and small business discount: 25% off any order. **Copies**: Each subscriber is licensed to make up to 10 copies of each issue or back issue. Call for rates to allow more copies.

Technical Editor: Les Owens.
Article Sourcing: Betsy Harter.
Production: Doug Scofield.
Distribution: Debbie Brandelli.
Marketing: Muneerah Vasanji.
Accounts: Evelyn Goreham.
Publisher: David Crowe.

Upcoming Fraud and Security Events

The following are several upcoming fraud and security conferences that may be of interest to the wireless and network security practitioners.

The Biometrics Symposium

19-20 June, 2001

Chicago, Illinois

www.iqpc.com/cgi-bin/templates/991498013131805419921000003/genevent.html?event=1504&topic

Electronic Signatures and Public Key Infrastructures

9-10 July, 2001

Jarvis International Regents Park

London

www.iir-conferences.com/site/_prod-grp.cfm?DirName=KJ1817&ConfCode=KJ1817&iv=26

The Biometrics Consortium 2001 Conference

12-14 September, 2001

Rosen Centre Hotel

Orlando, Florida

www.itl.nist.gov/div895/isis/bc2001/home.htm

Plastic Card and Online Fraud Prevention

24-25 September, 2001

Hotel Inter-Continental

Zurich, Switzerland

www.iir-conferences.com/site/_prod-grp.cfm?DirName=KJ1823&ConfCode=KJ1823&iv=26

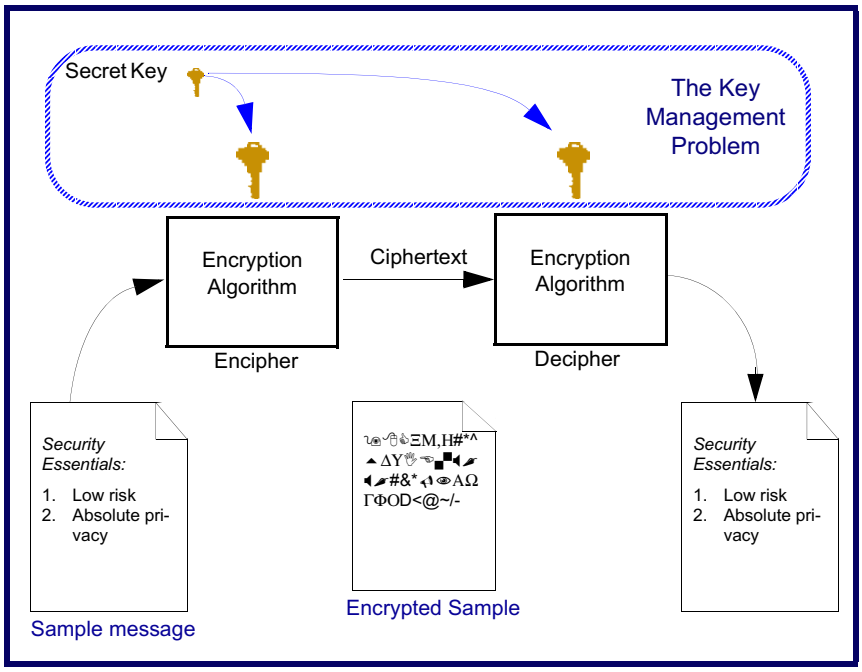
CSI 28th Annual Computer Security Conference and Exhibition

29-31 October, 2001

Washington, D.C.

www.gocsi.com/#Annual

Figure 1: Classical Cryptography and the Key Management Problem



Key Management

In the electronic world, cryptographic keys are used to identify individuals, electronic shops and service providers. Key management is the set of techniques and procedures supporting the establishment and maintenance of key sharing between two entities. It provides the means to:

- Initialize users within a domain;
- generate and distribute user and domain parameters;
- control the use of keys;
- maintain and revoke user and domain parameters; and
- store, backup and archive user and domain parameters.

Many problems in key management cannot be solved cryptographically. For example, initializing users typically involves collecting identity information about them, such as government identity numbers, names and birth dates, as well as verifying their credentials and assigning IDs.

Another example of a non-cryptographic aspect of key management is trust provisioning. One goal of key management is to establish bonds of trust leading back to a few very trustworthy entities. This entity is the bootstrap for enabling cryptographic keys and identities to be distributed in a trusted manner. These trusted individuals are often called Certification Authorities (CAs).

Certificates

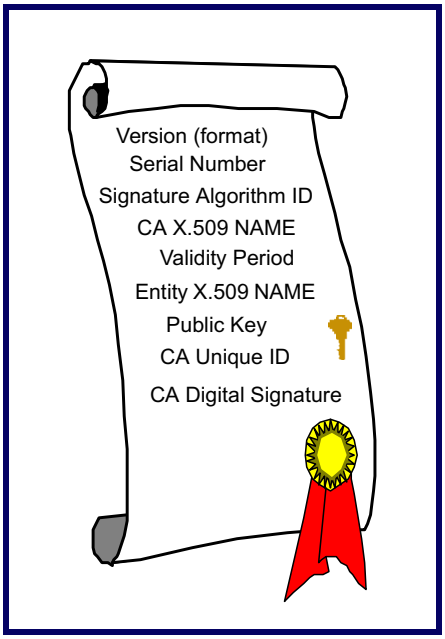
Public key cryptography provides a very scalable means to manage keys where an entity has two closely related key pairs: One publicly available for viewing, and another kept private to its owner. To use public key technology for authentication, a private key is used to digitally sign data and a public key is used to verify signatures. To use public key technology for confidentiality, the public key is used to encrypt data while the private key is used to decrypt data.

Symmetric key cryptography is a technique where only one key is shared between two communicating parties. Public key cryptography can allow for a more scalable key management solution; however, symmetric key cryptography can be implemented much more efficiently. This is why a hybrid solution is used by all major security standards. Hybrid cryptography is depicted in Figure 4.

The most popular method to securely transfer a public key is by using a digital certificate, also known as a public key certificate or an identity certificate. A CA constructs a data structure (to be signed) containing an entity's identity information and its public key. The CA signs this data and it becomes a "certificate" for the entity to prove to someone else what its public key is. Anyone who has an authentic copy of the CA's public key can verify that the CA has signed this certificate. Clients and servers often have CA public keys pre-installed in the form of a self-signed certificate for this purpose.

The entity whose public key is contained in a particular certificate is called the certificate subject. The accepted standard for defining the syntax and contents of a certificate is X.509. An X.509 certificate is depicted in Figure 2. Certificates contain information other than the public key -- in particular a validity period, restrictions on the use of the contained key and the CA's policy information. Different policies are often used for certificates issued to servers than certificates issued to individual users. To date, certificates for servers have been more widely deployed for open networks, such as the Internet.

Figure 2: X.509 Certificate



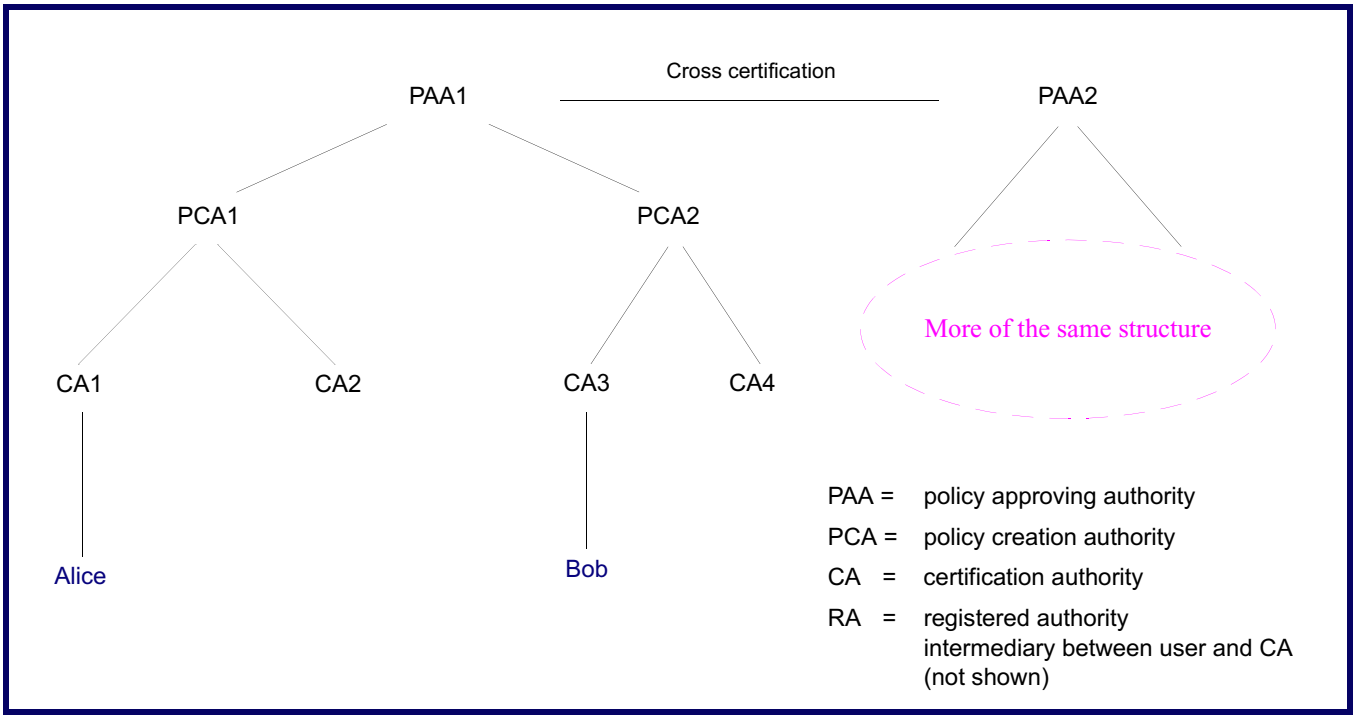
Certificate Chains

Simple implementations lead to a common CA to sign all certificates for a particular domain of users, or alternatively, the collection of CA certificates must be provisioned to all clients and servers within that domain. If the domain contains many users, or if it does not subscribe to a centralized business model, this approach will be cumbersome and ultimately ineffective.

A solution to this problem is to use a certificate chain, which is a sequence of certificates where each certificate is signed by the sub-CA whose certificate precedes it in the sequence (see Figure 3). The first certificate may be self-signed. Certificate chains are essential for using certificates in a scalable manner. In the earlier example, someone who has an authentic copy of the CA's public key can use it to verify the signature on an entity's certificate, which could then lead to acquisition of that entity's public key. This kind of architecture requires that the certificate verifier have the issuing CA's public key or public key certificate.

It is possible to construct a hierarchy of CAs with one root CA at the top, issuing certificates to several other sub-CAs.These sub-CAs can issue certificates to clients and servers, or they can issue certificates to their own sub-CAs. Provided the verifier is given, or provided it is possible to construct a chain of certificates beginning at the root and ending at the certificate for the particular entity whose public key it

Figure 3: Typical Certificate Hierarchy



needs, a certificate verifier needs only the root CA's public key to verify that the public key of any member of this hierarchy is valid.

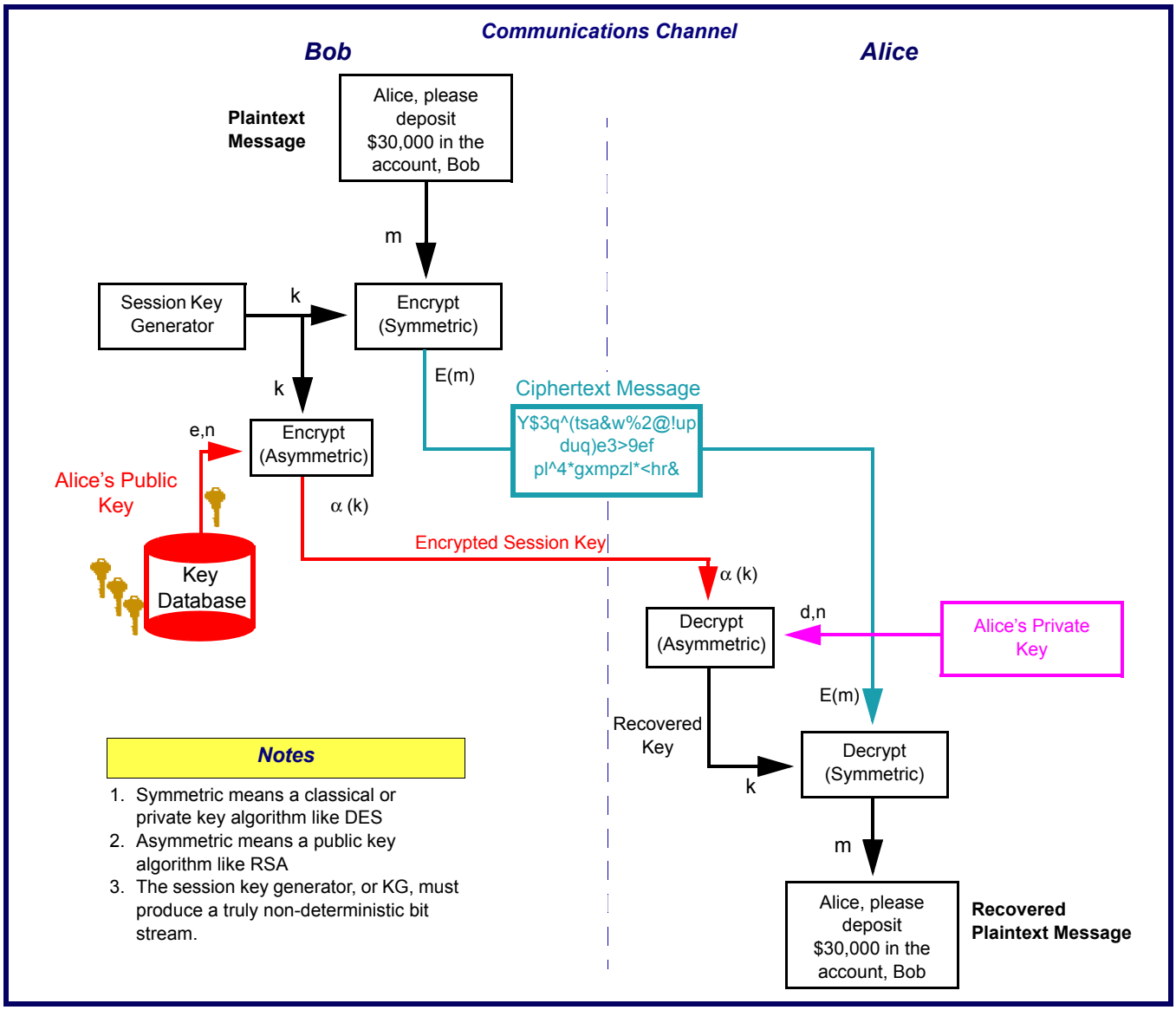
There are many subtleties in verifying that a certificate or certificate chain is valid. One disadvantage of using certificate chains is that multiple signatures must be verified. This may have implications on low-power CPU devices, such as mobile phones.

The example of certificate chains (Figure 3) describes them in the context of a hierarchical CA model. Other models also exist. Most notably is the trust model for PGP (Pretty Good Privacy), a so-called "web-of-trust" model. In this distributed approach to key management, there are no key certifying hierarchies for establishing trust. Each PGP user generates and distributes his own public key. Each user also signs the public keys of all other users with whom he communicates, creating an interconnected community of PGP users. The users of PGP then maintain a collection of these signed public keys in a file called a public-key ring. The interested reader may refer to [4] for additional information on PGP, the scheme developed by Phil Zimmermann primarily for email security.

Algorithms and Hybrid Certificates

A hybrid certificate is when the public key contained in the certificate and the private key used to sign the certificate are used in different algorithms. Hybrid certificates are not particularly common in wireline Internet applications (where the RSA signature algorithm is used almost exclusively). For wireless applications, hybrid certificates are more common. Typically, the CA will use the RSA algorithm to sign the certificate, but the key being certified may be used in the NTRU NSS or ECDSA signature algorithms (see Wireless Security Perspectives, September 2000 issue, for details and

Figure 4: Hybrid Cryptosystem



additional information on NTRU). NTRU NSS and ECDSA are emerging digital signature algorithms that offer computational efficiencies, making them an attractive alternative for the wireless environment.

The RSA algorithm works by selecting two large prime numbers (512 bits each, to produce an "RSA 1024 key"), `p` and `q`. Compute their product `n = p*q`. `n` is called the RSA modulus. Choose a number, `e`, less than `n`, that is relatively prime to `(p-1)*(q-1)`. A popular choice for `e` is $2^{16}+1$, one of the Fermat primes. This allows for very efficient implementations of RSA signature verification. Compute the value `d` so that `1 = d*e mod n`. The pair `(n, e)` is the public RSA key and the pair `(n, d)` is the private RSA key. For an example of RSA, see Appendix A.

To create an RSA digital signature on a message, `m`, compute $s = m^d \bmod n$. To verify that `s` is a valid RSA signature on `m`, verify that $m = s^e \bmod n$. `m` is usually not the message itself. Instead, it is a derivation of the message. The precise method to perform RSA signatures is described in RSA Labs' PKCS #1 specification. There has been much literature written about the RSA algorithm, such as efficient methods for generating keys and performing the modular exponentiation in the sign and verify functions.

CAs use the RSA algorithm because it has proven to interoperate on a wide-scale deployment. It is the most researched and studied algorithm, no IPR issues are surrounding it, RSA appears in almost all industry standards as the preferred algorithm, and its signature verification function can be efficiently implemented on most platforms. There is also a wide selection of secure cryptographic hardware modules for generating RSA signatures, but this is not true of almost all other public key algorithms. Cryptographic hardware is an absolute necessity for production grade CA operations.

Public Key Infrastructure Components

A PKI is composed of the underlying resources needed to implement public-key cryptography on a large scale. The core of a PKI is a network of CAs and their certificate management policies.

Registration

Certificate registration involves generating a cryptographic key pair, constructing a certificate signing request (CSR) and submitting this to a CA. A registration agent then verifies the requester's credentials and asks the CA to issue a digital certificate. The registration process is very critical to building an effective PKI, because this phase determines who gets a certificate, and what the level of trustworthiness is in each certificate. In most PKI deployments, a registration authority (RA) is enlisted to help the CA in this process. A strong PKI solution should have a very strong RA component to it – one that can validate certificate requests against particular policies and business requirements. RA products should be flexible enough to validate requests against different kinds of databases that contain customer or employee lists.

Entities requesting certificates send their certificate signing request to a CA or RA in several different mediums. Popular methods are Web-based, e-mail, special PKI client software or Wireless Application Protocol (WAP) phones. There are a few common standard formats used to request a certificate, such as PKCS #10 from RSA Labs, CMP or CMC from IETF-PKIX. A well constructed RA should support several mediums to accept certificate signing requests, and it should support a variety of formats if necessary. This configuration is particularly important in a wireless PKI where standards are still immature.

Another important activity that often coincides with certificate registration is publishing a certificate to a directory. Often, a CA will publish a newly issued certificate to a database or to an LDAP directory for access. Integration with an LDAP directory is typically an important requirement in the seamless integration of a PKI solution.

Revocation

From time to time, an entity's certificate needs to be revoked. Sometimes the private key is lost, its security is suspect or an entity has left the domain it was a part of. Generally, someone like an RA informs the CA that a certificate needs to be revoked. In most PKI solutions, the CA marks the certificate revoked in its internal records, and at fixed intervals, it publishes a list of certificates – called a certificate revocation list (CRL) – that have not yet expired but are revoked.

Other methods of indicating revocation exist. For instance, on-line status checking is when an entity wanting to know the status of a certificate's revocation by making a query to a validation authority (VA). VAs may have access to information about a certificate's current status before a CRL is published, due to a special relationship they have with the CA. One protocol used to provide this information is the IETF's on-line certificate status protocol (OCSP).

Renewal

Certificate renewal is a simplified variation of certificate registration. Most certificates are issued with a validity period of one or two years, thus allowing the RA to periodically validate an entity's credentials, which helps keep revocation lists short. CA and sub-CA certificates are often issued with validity periods of 10 or 20 years, since these anchors of trust are widely deployed in browsers and wireless handsets, and since updating them is not easy, in general.

Because the RA has already done much of the gathering of information about a user in the initial registration, the process of renewal is often easier, since they only need to verify the information is still current. During the initial registration, users may authenticate themselves to the RA by showing knowledge of a secret password only known to the user and the RA. This adds the complexity of determining how to agree upon a password.

However, in the renewal process, users can authenticate themselves to the RA by performing a digital signature and presenting their current certificate to the RA. Instead of establishing another password between the user and the RA, the RA can identify users with their current certificate.

Certain security policies mandate that users generate a new key pair when renewing their certificates. Others allow the users to continue to use the current key, only being issued a new certificate. In many instances of wireless PKI deployment, such as on subscriber identity module (SIM) or WAP identity module (WIM) cards, the key-pair cannot be regenerated, so a re-keying is not possible.

Key Usage

How entities use their keys and certificates is important from both cryptographic and policy points of view. Using the same key in different protocols can sometimes provide adversaries with bits of the private key. While this situation is rare, it is prudent to have another level of security.

From a policy point of view, many digital signature directives mandate that a key used to sign data must be unlocked using a pass phrase every time it is used. Other kinds of keys, such as a basic key exchange, need only be unlocked by a pass phrase at the beginning of a session, which makes them available for use by an application during the session.

Key life cycle also affects key usage. A private key that is used to decrypt information may need to be stored after its useful lifetime has expired, so that encrypted data can still be decrypted. However, a private key used to sign data should never be used once its lifetime has expired. In these circumstances, there is a conflict in what to do after the key's lifetime has expired and when separation of keys by their usage is appropriate.

Wireless PKI

Wireless PKI presents a number of challenges – both in development and deployment – such as low data transmission rates and battery power. Additionally, products designed to provide PKI solutions in wireline environments generally perform poorly or cannot be deployed in wireless environments at all.

More specifically, the principal differences in deploying a PKI for wireless are:

a. Message formats used in standard wireline PKI deployments are large and difficult for handsets to process;

b. Security protocols used in standard wireline PKIs are often too bandwidth-intensive for wireless networks;

c. Handsets have less flexibility in how keys and certificates can be provisioned;

d. Handsets have very limited user interfaces, affecting how the RA can gather information about someone who is requesting a certificate

e. Wireless gateways are often used to convert from wireless network protocols to Internet protocols, preventing the ability to establish an end-to-end encrypted session from a client to the application server;

f. There are many more cryptographic algorithms used by handsets than on Internet PCs and PKI-enabled applications, and CAs must support these different varieties.

A PKI portal, essentially an advanced wireless gateway providing RA services to handsets, is a first step in supporting wireless PKI. It can manage much of the translation from several wireless protocols and data formats to a common format used by CAs. However, CA products must still be able to issue more compact certificates for use in wireless, while supporting multiple algorithms to validate the key submitted for certification.

A recent key trend in PKI standards is to profile X.509 certificates for particular applications. This is a reaction to activities promoting specialized, compact and easy-to-process certificate formats including: X9.68 and WTLS simple server certificate formats. As an alternative to using non-X.509 formats, standards such as WAP have defined special flavors of X.509 certificates that can be more easily managed by handsets. Specialized, non-X.509 certificates, however, are still required to support some applications and handsets.

Conclusions

A Public-Key Infrastructure is the best way to offer the high level of security assurance needed for secure B2B and B2C e-business environments. Unfortunately, PKIs are perceived to be very complex and expensive with respect to up-front costs and user registration. Unlike wireline PKIs, deploying a PKI over wireless introduces new complexities in a different community of users, and it requires extensive software support for many different handheld devices with unique operating systems.

About Diversinet

Diversinet Corp.

www.diversinet.com

has been developing PKI solutions for wireless environments for over three years, offering a comprehensive product suite for PKI-enabling m-commerce and m-business applications on the widest variety of platforms, including GSM SIM cards, Palms, PocketPCs, WAP devices and RIM interactive pagers. Diversinet has developed products and solutions for enterprises, wireless carriers and application service providers to manage the back-end PKI components and to simplify the certificate registration process for RAs and users. This total solution provides the high level of security of PKI with minimal development effort and operational cost.

About the Author

Michael Crerar joined Diversinet in 1998. Currently, he is Director of Security Infrastructure at Diversinet Corp., and he is involved in Diversinet's wireless PKI products architecture and design. He participates in industry groups such as WAP Forum and IETF to develop standards for wireless security. Michael holds a Masters degree in cryptography from the University of Waterloo in Canada. Michael has previously held positions with Citibank and IBM's Footprint Software division. He can be reached at

mcrerar@dvnet.com.

Glossary of Terms

CA: Certificate Authority

CMP: Certificate Management Protocol

CRL: Certificate Revocation List

CSR: Certificate Signing Request

ECDSA: Elliptical Curve Digital Signature Algorithm

IETF: Internet Engineering Task Force

LDAP: Lightweight Directory Access Protocol

OCSP: On-line Certificate Status Protocol

PKC: Public-key Cryptography

PKCS: Public Key Cryptography Standard

PKI: Public Key Infrastructure

PKIX: Public-Key Infrastructure – X.509

RA: Registration Authority

RSA: Rivest-Shamir-Adelman

SIM: Subscriber Identity Module

VA: Validation Authority

WAP: Wireless Application Protocol

WIM: WAP Identity Module

Appendix A

This appendix provides a simple example of the RSA algorithm.

Suppose the following prime numbers are chosen:

$P = 17$

$Q = 31$

Here P and Q are relatively prime. That is, they have no factors but 1 (i.e., GCD(17,31) = 1)

Then the modulus is simply:

$N = P \times Q = 527$

The Euler Totient function, $\phi(N)$, is the following:

$\phi(N) = (P-1) \times (Q-1) = 480$

Then, we let the public exponent be

$e = 7$

Then the private exponent is

$d = 343$,

From $e \times d = 1 \pmod{\phi(N)}$ or

$7 \times 343 = 1 \bmod 480$

Since, $7 \times 343 = 2401 = 5 \times 480 + 1$

To encrypt a message, let's suppose the plaintext is

$M = 2$

Then Ciphertext, $C = M^e \bmod N$

$= 2^7 \bmod 527 = 128$

To decrypt, the ciphertext message is

Plaintext, $M = C^d \bmod N$

$= 128^{343} \bmod 527$

$= 128^{256} \times 128^{64} \times 128^{16} \times 128^4 \times 128^2 \times 128 \bmod 527$

$= 35 \times 256 \times 35 \times 101 \times 47 \times 128 \bmod 527$

$= 2 \bmod 527$

$= 2$

Hence the plaintext message is recovered.

Selected References and Additional Reading

[1]. Rivest, R.L., Shamir, A. and L. Adleman. A Method for obtaining Digital Signatures and Public Key Cryptosystems, Communications of the ACM, pp. 120 - 126, February 1978.

[2]. Diffie, Whitfield. The First Ten Years of Public Key Cryptography, Proceedings of the IEEE, Vol. 76, No. 5, pp. 560 - 577, May 1988.

[3]. Schneier, Bruce. Applied Cryptography: Protocols, Algorithms and Source Code in C, 2nd Edition. New York: John Wiley and Sons, Inc., 1996.

[4]. Stallings, William. Cryptography and Network Security: Principles and Practice, 2nd Edition. Upper Saddle River, NJ: Prentice Hall, 1998.

[5]. Nichols, Randall. ICSA Guide to Cryptography. McGraw-Hill, 1999.

To Probe Further

To obtain additional information about PKI for wireless, please contact the WSP editors (wsp@cnp-wireless.com) or the author, Michael Crerar, at:

mcrerar@dvnet.com

www.diversinet.com

For more information on PKI initiatives and standards effort, PKI products and services, and PKI technology, visit the following sites:

Atomic Tangerine's InfoSec University
www.infosecu.com

Baltimore Technologies
www.baltimore.com

Certco
www.certco.com

CitX Corporation
www.citx.com

European Electronic Messaging Association
www.eema.org/ecaf

Entrust
www.entrust.com

Federal PKI Steering Committee
www.cio.gov/fpkisc/

Federal Bridge CA
csrc.nist.gov/pki/rootca/

Gradient (including Entegrity Solutions)
www.gradient.com

Internet Engineering Task Force
www.ietf.org

nCipher
www.ncipher.com

Privador - Defenders of the e
www.privador.com

PKI Forum
www.pkiforum.com

The PKIX group:
www.ietf.org/html.charters/pkix-charter.html

Valicert
www.valicert.com

RSA Security
www.rsa.com

Verisign
www.verisign.com

Xcert
www.xcert.com

Odyssey Technologies
www.odysseytec.com

Fraud And Security Patent News

The US Patent and Trademark Office (USPTO) recently granted the following 20 fraud and security patents. The patent number, invention title, inventor, and assignee (owner) are provided. All of these patents issued in May 2001.

These may be of interest to our wireless security practitioners. With the listing below, one can see who is doing what in the world of inventions.

Patent Number: 6,240,517

Title: Integrated Circuit Card, Integrated Circuit card processing system, and integrated card authentication method

Inventor: Mitsuru Nishioka

Assignee: Toshiba

Patent Number: 6,240,515

Title: Method of authenticating a magnetic card

Inventor: Steven Carnegie, et. al.

Assignee: NCR Corporation

Patent Number: 6,240,513

Title: Network Security Device

Inventor: Aharon Friedman and Eva Bozoki

Assignee: Fortress Technologies, Inc.

Patent Number: 6,240,512

Title: Single Sign on Mechanism having master key synchronization

Inventor: Yi Fang et. al.

Assignee: IBM Corporation

Patent Number: 6,240,436

Title: High speed Montgomery value calculation

Inventor: Matthew McGregor

Assignee: Rainbow Technologies, Inc.

Patent Number: 6,240,188

Title: Distributed group key management scheme for secure many-to-many communication

Inventor: Lakshminath Dondeti et. al.

Assignee: Matsushita Electric Industrial Co., Ltd.

Patent Number: 6,240,187

Title: Key replacement in a public key cryptosystem

Inventor: Tony Lewis

Assignee: Visa International

Patent Number: 6,240,185

Title: Steganographic techniques for securely delivering electronic digital rights management control information over insecure communication channels

Inventor: David Van Wie and Robert Weber

Assignee: Intertrust Technologies Corporation

Patent Number: 6,240,184

Title: Password synchronization

Inventor: Dung Huynh et. Al.

Assignee: RSA Security Inc.

Patent Number: 6,240,121

Title: Apparatus and method for watermark data insertion and apparatus and method for watermark data detection

Inventor: Takanori Senoh

Assignee: Matsushita Electric Industrial Co., Ltd.

Patent Number: 6,240,074

Title: Secure communication hub and method of secure data communication

Inventor: Ronald Chandos et. al.

Assignee: Motorola

Patent Number: 6,239,976

Title: Reinforced micromodule

Inventor: Thomas Templeton

Assignee: Comsense Technologies, Ltd.

Patent Number: 6,239,881

Title: Apparatus and method for securing facsimile transmissions

Inventor: Shmuel Shaffer

Assignee: Siemens Information and communication Networks, Inc.

Patent Number: 6,237,786

Title: Systems and methods for secure transaction management and electronic rights protection

Inventor: Karl Ginter et. al.

Assignee: Intertrust Technologies Corporation

Patent Number: 6,237,137

Title: Method and system for preventing unauthorized access to a computer program

Inventor: Alan Beelitz

Assignee: Dell USA, L.P.

Patent Number: 6,237,097

Title: Robust efficient distributed RSA-key generation

Inventor: Yair Frankel et. al.

Assignee: Certco, Inc.

Patent Number: 6,237,095

Title: Apparatus for transfer of secure information between a data carrying module and an electronic device

Inventor: Stephen Curry et. al.

Assignee: Dallas Semiconductor Corporation

Patent Number: 6,237,093

Title: Procedure for setting up a secure service connection in a telecommunication system

Inventor: Harri Vatanen

Assignee: Sonera Oyj

Obtaining U.S. Patents

To review the specification and claims of these patents, visit the US Patent and Trademark Office web-site at:

www.uspto.gov.

Obtain a copy of one of these patent numbers from the US Patent and Trademark Office at the address or telephone numbers below:

General Information Services Division
U.S. Patent and Trademark Office
Crystal Plaza 3, Room 2C02
Washington, DC 20231

800-786-9199 or 703-308-4357