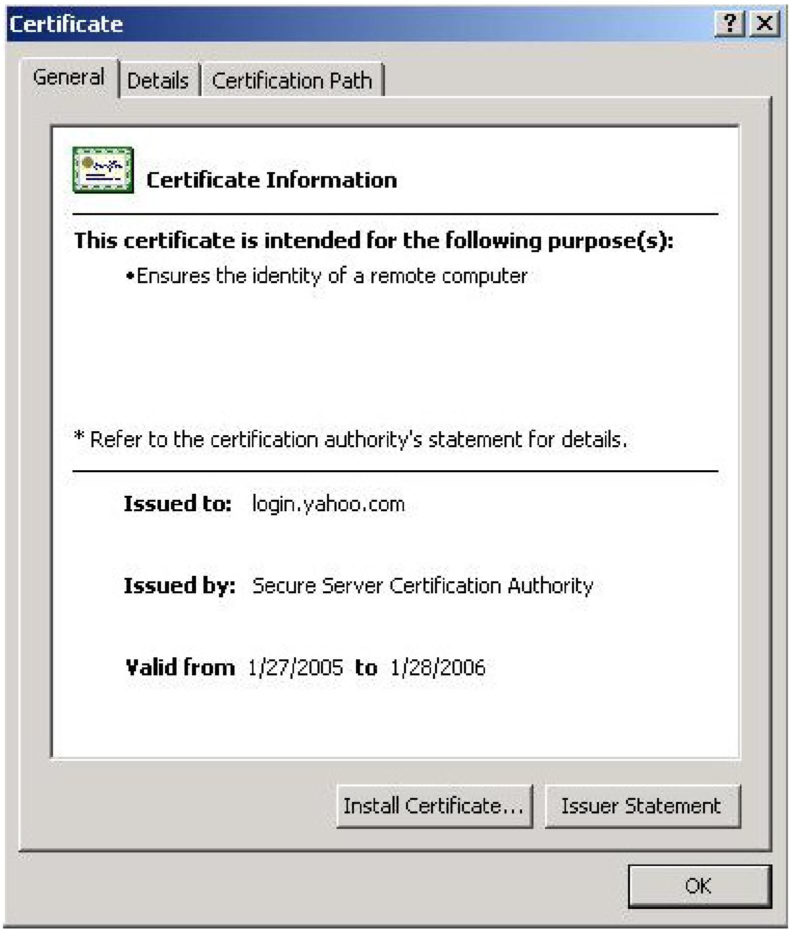
Digital certificates



Certificate Information

This certificate is intended for the following purpose(s):

- Ensures the identity of a remote computer

* Refer to the certification authority's statement for details.

Issued to: login.yahoo.com

Issued by: Secure Server Certification Authority

Valid from 1/27/2005 to 1/28/2006

Name – Vivek kumar

EM No – 230409

Subject – E-Business technologies

Prof. –Dr. Eduard heindl

Certificate of Declaration

I certify that the work in this term paper has been written by me. Any help that I have received in my research work and the preparation of the term paper itself has been acknowledged at the end of paper. In addition, I certify that all information sources and literature used are indicated in the term paper

Vivek kumar

230409

Table of contents

1.0 Introduction to Digital Certificates

Digital Certificates provide a means of proving your identity in electronic transactions, much like a driver license or a passport does in face-to-face interactions. With a Digital Certificate, you can assure friends, business associates, and online services that the electronic information they receive from you are authentic.This document introduces Digital Certificates and answers questions you might have about how Digital Certificates are used. For information about the cryptographic technologies used in Digital Certificates.

Digital certificates are the equivalent of a driver's license, a marriage license, or any other form of identity. The only difference is that a digital certificate is used in conjunction with a public key encryption system. Digital certificates are electronic files that simply work as an online passport. Digital certificates are issued by a third party known as a Certification Authority such as VeriSign or Thawte. These third party certificate authorities have the responsibility to confirm the identity of the certificate holder as well as provide assurance to the website visitors that the website is one that is trustworthy and capable of serving them in a trustworthy manner.

Digital certificates have two basic functions. The first is to certify that the people, the website, and the network resources such as servers and routers are reliable sources, in other words, who or what they claim to be. The second function is to provide protection for the data exchanged from the visitor and the website from tampering or even theft, such as credit card information.

1.1 Who Use Digital Certificates

Digital Certificates can be used for a variety of electronic transactions including e-mail, electronic commerce, groupware and electronic funds transfers. Netscape's popular Enterprise Server requires a Digital Certificate for each secure server.
For example, a customer shopping at an electronic mall run by Netscape's server software requests the Digital Certificate of the server to authenticate the identity of the mall operator and the content provided by the merchant. Without authenticating the server, the shopper should not trust the operator or merchant with sensitive information like a credit card number. The Digital Certificate is instrumental in establishing a secure channel for communicating any sensitive information back to the mall operator
Virtual malls, electronic banking, and other electronic services are becoming more commonplace, offering the convenience and flexibility of round-the-clock service direct from your home. However, your concerns about privacy and security might be preventing you from taking advantage of this new medium for your personal business. Encryption alone is not enough, as it provides no proof of the identity of the sender of the encrypted information. Without special safeguards, you risk being impersonated online. Digital Certificates address this problem, providing an electronic means of verifying someone's identity. Used in conjunction with encryption, Digital Certificates provide a more complete security solution, assuring the identity of all parties involved in a transaction. Similarly, a secure server must have its own Digital Certificate to assure users that the server is run by the organisation it claims to be affiliated with and that the content provided is legitimate.

2.0 Types of Digital Certificate:-

2.1 Identity Certificates

An Identity Certificate is one that contains a signature verification key combined with sufficient information to identify (hopefully uniquely) the keyholder. This type of certificate is much subtler than might first be imagined and will be considered in more detail later.

2.2 Accreditation Certificates

This is a certificate that identifies the keyholder as a member of a specified group or organisation without necessarily identifying them. For example, such a certificate could indicate that the keyholder is a medical doctor or a lawyer. In many circumstances, a particular signature is needed to authorise a transaction but the identity of the keyholder is not relevant. For example, pharmacists might need to ensure that medical prescriptions are signed by doctors but they do not need to know the specific identities of the doctors involved.

Here the certificate states in effect that the keyholder, whoever they are, has 'permission to write medical prescriptions'. Accreditation certificates can also be viewed as authorisation (or permission) certificates. It might be thought that a doctor's key without identity would undermine the ability to audit the issue of medical prescriptions. However, while such certificate might not contain keyholder identity data, the certificate issuer will know this so such requirements can be met if necessary.

2.3 Authorisation and Permission Certificates

In these forms of certificate, the certificate signing authority delegates some form of authority to the key being signed. For example, a Bank will issue an authorisation certificate to its customers saying 'the key in this certificate can be used to authorise the withdrawal of money from account number 271828'. In general, the owner of any resource that involves electronic access can use an authorisation certificate to control access to it. Other examples include control of access to secure computing facilities and to World Wide Web pages. In banking an identity certificate might be used to set up an account but the authorisation certificate for the account will not itself contain identity data. To identify the owner of a certificate a bank will typically look up the link between account numbers and owners in its internal databases. Placing such information in an authorisation certificate is actually undesirable since it could expose the bank or its customers to additional risks.

3.0 The Parties to a Digital Certificate

In principle there are three different interests associated with a digital certificate:

3.1 The Requesting Party –

The party who needs the certificate and will offer it for use by others – they will generally provide some or all of the information it contains.

3.2 The Issuing Party –

The party that digitally signs the certificate after creating the information in the certificate or checking its correctness.

3.3 The Verifying Party (or Parties) –

Parties that validate the signature on the certificate and then rely on its contents for some purpose. For example, a person – the requesting party – might present paper documents giving proof of identity to a government agency – the issuing party – who will then provide an identity certificate that could then be used by a bank – the verifying party – when the requesting party opens a bank account.

The term 'relying party' is sometimes uses instead of 'verifying party' but this can be misleading since the real purpose is to identify a party who checks the certificate before relying on it. In a credit card transaction many parties might handle a certificate and hence rely on it in some way but only a few of these might actually check the validity of the certificate. Hence a 'verifying party' is a party that checks and then relies on the contents of a certificate, not just one that depends on it without checking its validity. The actual parties involved in using a certificate will vary depending on the type of certificate.

4.0 Public and Private key-

Public-key refers to a cryptographic mechanism. It has been named public-key to differentiate it from the traditional and more intuitive cryptographic mechanism known as: symmetric-key, shared secret, secret-key and also called private-key.

Symmetric-key cryptography is a mechanism by which the same key is used for both encrypting and decrypting; it is more intuitive because of its similarity with what you expect to use for locking and unlocking a door: the same key. This characteristic requires sophisticated mechanisms to securely distribute the secret-key to both parties2. Public-key on the other hand, introduces another concept involving key pairs: one for encrypting, the other for decrypting. This concept, as you will see below, is very clever and attractive, and provides a great deal of advantages over symmetric-key:

- Simplified key distribution
- Digital Signature
- Long-term encryption

However, it is important to note that symmetric-key still plays a major role in the implementation of a Public-key Infrastructure or *PKI*.

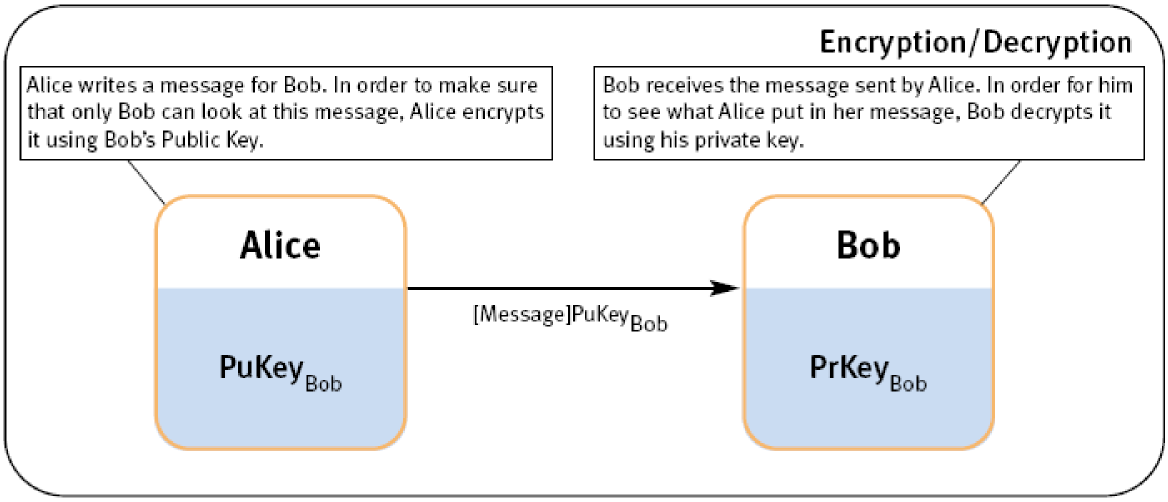
4.1 A definition

Public-key is commonly used to identify a cryptographic method that uses an *asymmetric-key* pair3: a *public-key* and a *private-key* 4. Public-key encryption uses that *key pair* for encryption and decryption. The public-key is made public and is distributed widely and freely.

The private-key is never distributed and must be kept secret. Given a key pair, data encrypted with the public-key can only be decrypted with its privatekey; conversely, data encrypted with the private-key can only be decrypted with its public key. This characteristic is used to implement encryption and digital signature.

4.2 Encryption and Decryption-

Encryption is a mechanism by which a message is transformed so that only the sender and recipient can see. For instance, suppose that Alice wants to send a private message to Bob. To do so, she first needs Bob's public-key; since everybody can see his public-key, Bob can send it over the network in the clear without any concerns. Once Alice has Bob's public-key, she encrypts the message using Bob's public-key and sends it to Bob. Bob receives Alice's message and, using his private-key, decrypts it.


Encryption/Decryption

Alice writes a message for Bob. In order to make sure that only Bob can look at this message, Alice encrypts it using Bob's Public Key.

Bob receives the message sent by Alice. In order for him to see what Alice put in her message, Bob decrypts it using his private key.

Alice
$\text{PuKey}_{\text{Bob}}$
$[\text{Message}]\text{PuKey}_{\text{Bob}}$
Bob
$\text{PrKey}_{\text{Bob}}$

Figure 1: Encryption/Decryption principles

5.0 Digital Signature

The digital certificate was digitally signed. The holder of adigital certificate can also use it to digitally sign other digital documents, forexample, purchase orders, grant applications, financial reports or student transcripts. A digital signature is not an image of your pen and ink signature—itis an attachment to a document that contains an encrypted version of the document created using the signer's private key. Once a document is signed, no part of that document can be changed without invalidating the signature. Thus if someone obtained a copy of your digital certificate and changed the name in it to be their own name, any application receiving that modified certificate would see immediately that the signature on it was not valid. In this sense, a digital credential is much better than a traditional ID card to prove that the holder is really the person to whom it was issued. In fact, digital signatures in general are much more useful than pen and ink signatures since anyone checking the signature also can find out something about the signer in order to know whether the signature is meaningful.

5.1 Digital Signature and Verification

Digital signature is a mechanism by which a message is authenticated i.e. proving that a message is effectively coming from a given sender, much like a signature on a paper document. For instance, suppose that Alice wants to digitally sign a message to Bob. To do so, she uses her private-key to encrypt the message; she then sends the message along with her public-key (typically, the public key is attached to the signed message). Since Alice's public-key is the only key that can decrypt that message, a successful decryption constitutes a Digital Signature Verification, meaning that there is no doubt that it is Alice's private key that encrypted the message.

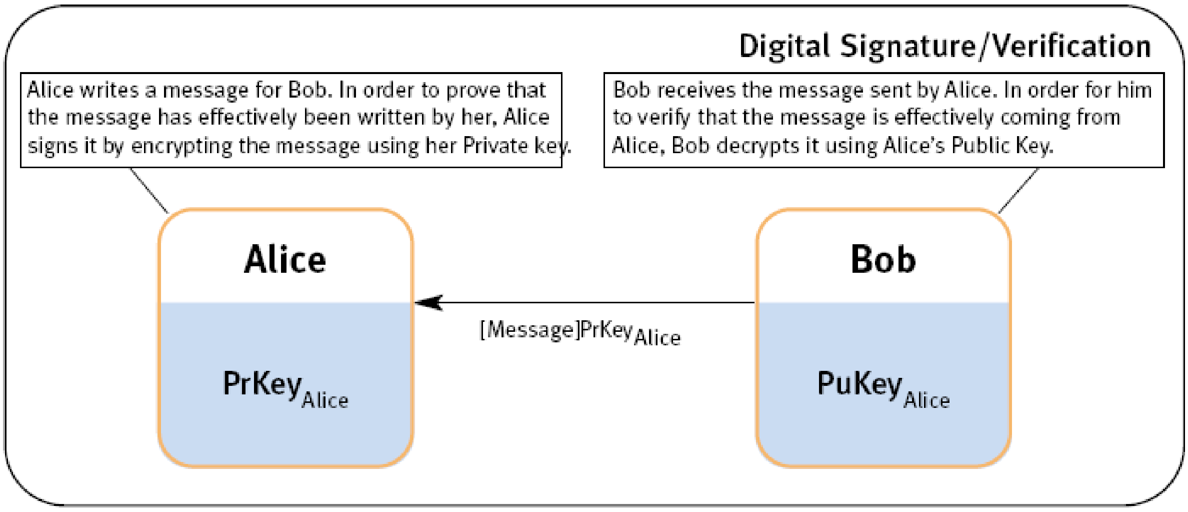

Digital Signature/Verification
Alice writes a message for Bob. In order to prove that the message has effectively been written by her, Alice signs it by encrypting the message using her Private key.
Bob receives the message sent by Alice. In order for him to verify that the message is effectively coming from Alice, Bob decrypts it using Alice's Public Key.
Alice
PrKey Alice
[Message]PrKey Alice
Bob
PuKey Alice

Figure 2: Digital Signature/Verification principles

5.2 How long do digital signatures remain valid?

Normally, a key expires after some period of time, such as one year, and a document signed with an expired key should not be accepted. However, there are many cases where it is necessary for signed documents to be regarded as legally valid for much longer than two years; long-term leases and contracts are examples. By registering the contract with a digital time-stamping service at the time it is signed, the signature can be validated even after the key expires.

If all parties to the contract keep a copy of the time-stamp, each can prove that the contract was signed with valid keys. In fact, the time-stamp can prove the validity of a contract even if one signer's key gets compromised at some point after the contract was signed. Any digitally signed document can be time-stamped, assuring that the validity of the signature can be verified after the key expires

5.3 Hashing

For Digital signature, another technique used is called hashing. Hashing produces a message digest that is a small and unique7 representation (a bit like a sophisticated checksum) of the complete message. Hashing algorithms8 are a one-way encryption, i.e. it is impossible to derive the message from the digest. The main reasons for producing a message digest are:
1 The message integrity being sent is preserved; any message alteration will immediately be detected;
2 The digital signature will be applied to the digest, which is usually considerably smaller than the message itself;
3 Hashing algorithms are much faster than any encryption algorithm (asymmetric or symmetric).
The following sections explains what really happens when encrypting and signing a message on one hand, and when decrypting a message and verifying its signature on the other hand.

5.4 Steps for signing and encrypting a message-

Figure 3 below shows the set of operations required when Alice wants to send a signed and encrypted message to Bob.

Alice sending a signed and encrypted message to Bob

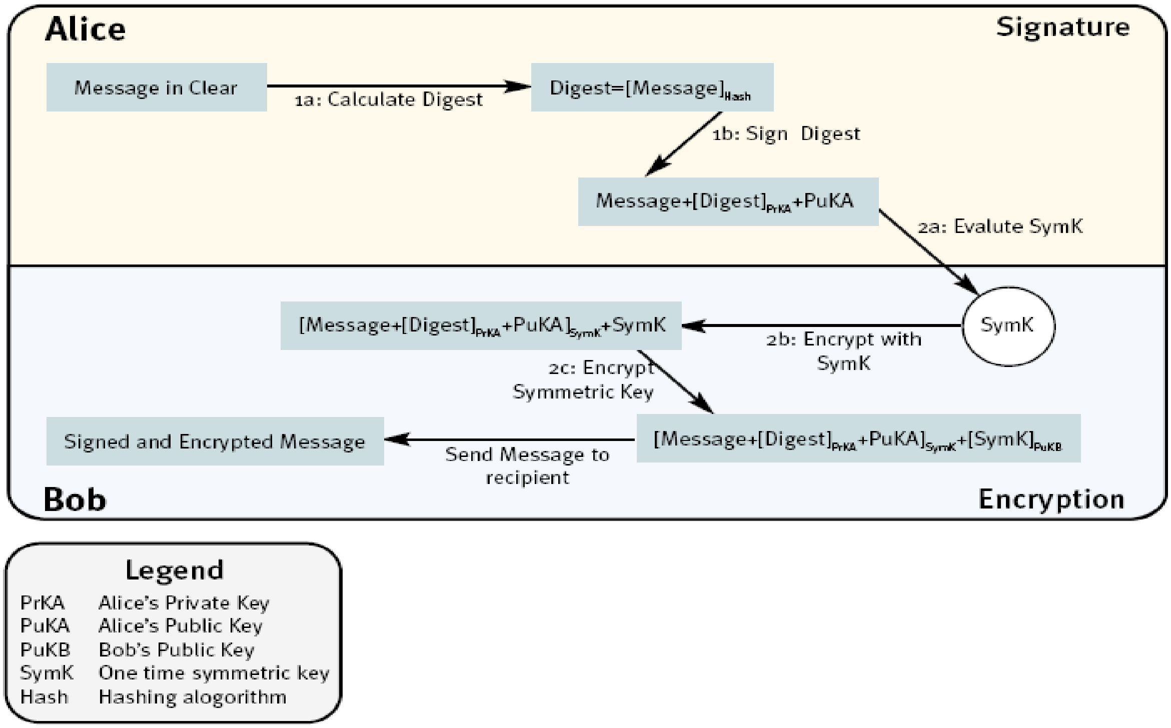

Alice
Signature
Message in Clear
1a: Calculate Digest
Digest=[Message]$_{Hash}$
1b: Sign Digest
Message+[Digest]$_{PrKA}$+PuKA
2a: Evalute SymK
SymK
2b: Encrypt with SymK
[Message+[Digest]$_{PrKA}$+PuKA]$_{SymK}$+SymK
2c: Encrypt Symmetric Key
[Message+[Digest]$_{PrKA}$+PuKA]$_{SymK}$+[SymK]$_{PuKB}$
Send Message to recipient
Signed and Encrypted Message
Bob
Encryption
Legend
PrKA Alice's Private Key
PuKA Alice's Public Key
PuKB Bob's Public Key
SymK One time symmetric key
Hash Hashing alogorithm

Figure 3: Signature and Encryption details with keys

1) Message signature.

Digital signature includes two steps:

a) ***Message digest evaluation.*** The main purpose for evaluating a digest is to ensure that the message is kept unaltered; this is called message integrity.
b) ***Digest signature***. A signature is in fact an encryption using the issuer's (Alice in this case) private-key. Included in the signature is also the hashing algorithm name used by the issuer. The issuer's public-key is also appended to the signature. Doing so lets anyone decrypt and verify the signature using the issuer's public-key and hashing algorithm. Given the properties of public-key encryption and hashing
algorithms, the recipient has proof that:
i) The issuer's private-key has encrypted the digest;
ii) The message is protected against any alteration.

2) Message encryption.
Encryption includes the following 3 steps:
a) ***Creation of a one time symmetric encryption/decryption key***. Remember that encryption and decryption algorithms using asymmetric-keys are too slow to be used for long messages; symmetric-key algorithms are very efficient and are therefore used.
b) ***Message encryption.*** The whole message (the message itself and the signature) is encrypted using SymK, the symmetric-key evaluated above.
c) ***Symmetric-key encryption***. SymK is also used by the recipient to decrypt the message. SymK must therefore be available to the recipient (Bob) only. The way to hide the Symk from everybody except the recipient is to encrypt it using the recipient's public-key. Since SymK is a small piece of information compared to a message (that could be very long), the performance penalty associated with the relative inefficiency of asymmetric-key algorithms is acceptable. One interesting point to mention is that if Alice wants to send the same message to more than one recipient, say Bob and John for instance, the only additional operation to be performed is to repeat 'step 2) c)' for John. Hence, the message that both Bob and John would receive would look like:
[Message+[Digest]PrKA+PuKA]SymK+[SymK]PuKB+[SymK]PuKJ . Notice that the exact same SymK will be used by Bob and John to decrypt the message.

5.5 Steps for Decrypting and verifying the signature of a message-

Figure 4 below shows the set of operations required when Bob wants to decrypt and verify the message sent by Alice.

1)Message decryption.
The decryption includes the following steps:
a) ***Symmetric-key decryption***. The one time symmetric-key has been used to encrypt the message. This key (SymK) has been encrypted using the recipient's (Bob) public-key. Only Bob can decrypt SymK and use it to decrypt the message9.
b) ***Message decryption.*** The message (which includes the message itself and the signature) is decrypted using SymK.
2) Signature verification.
The signature verification includes the following 3 steps:

a) ***Message digest decryption***. The digest has been encrypted using the issuer's (Alice) private-key. The digest is now decrypted using the issuer's public-key included in the message.
b) ***Digest evaluation***. Since hashing is a one-way process i.e. the message cannot be derived from the digest itself, the recipient must re-evaluate the digest using the exact same hashing algorithm the issuer used.
c) ***Digests comparison***. The digest decrypted in a) and the digest evaluated in b) are compared. If there is a match, the signature has been verified, and the recipient can accept the message as coming unaltered from the issuer. If there is a mismatch this could mean that:
i) The message has not been signed by the issuer or
ii) The message has been altered.
iii) In both cases, the message should be rejected.

Bob Decrypting and Verifying message sent by Alice

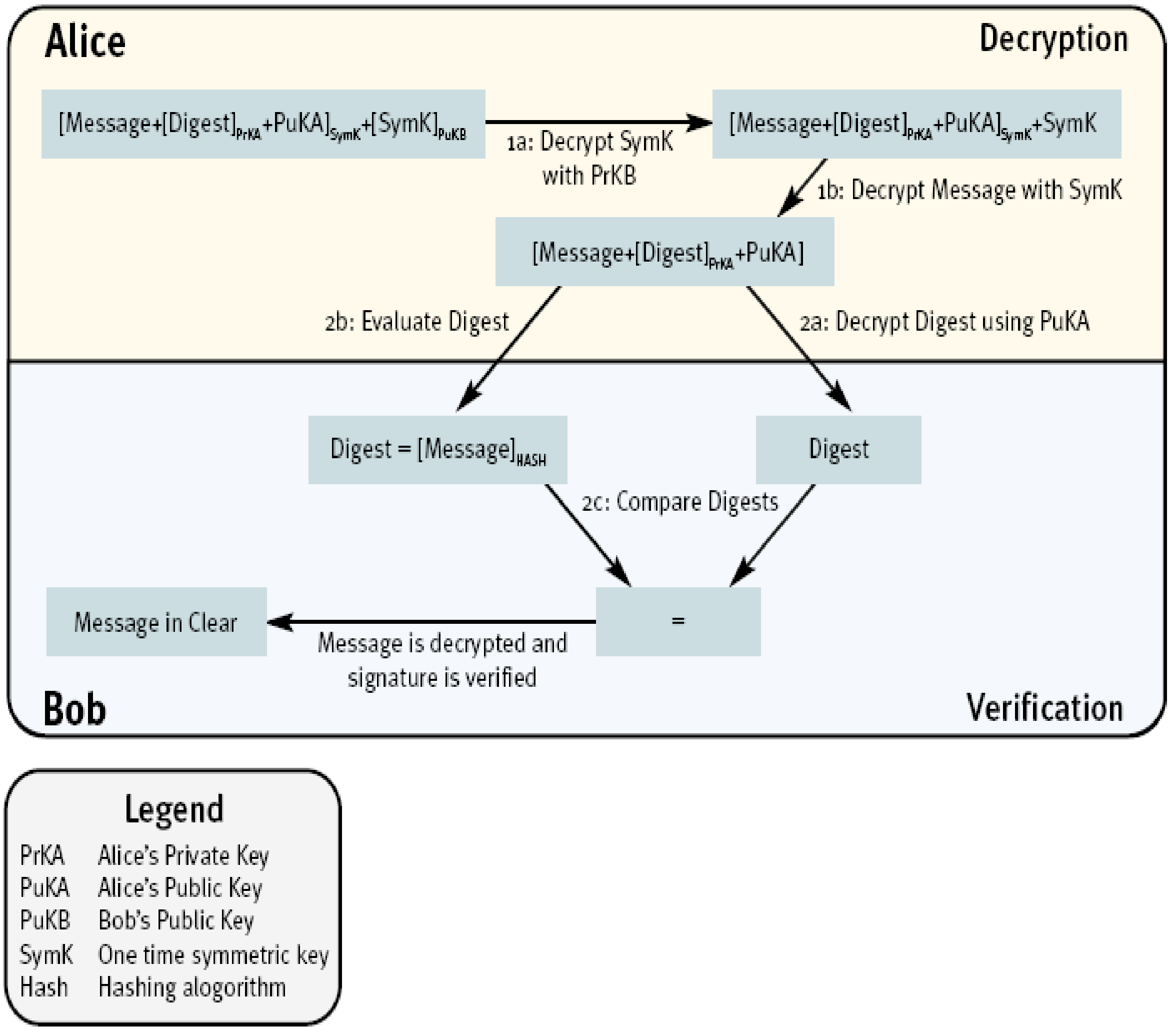

Alice
Decryption
$[\text{Message}+[\text{Digest}]_{PrKA}+\text{PuKA}]_{SymK}+[\text{SymK}]_{PuKB}$
1a: Decrypt SymK with PrKB
$[\text{Message}+[\text{Digest}]_{PrKA}+\text{PuKA}]_{SymK}+\text{SymK}$
1b: Decrypt Message with SymK
$[\text{Message}+[\text{Digest}]_{PrKA}+\text{PuKA}]$
2b: Evaluate Digest
2a: Decrypt Digest using PuKA
$\text{Digest} = [\text{Message}]_{HASH}$
Digest
2c: Compare Digests
Message in Clear
Message is decrypted and signature is verified
=
Bob
Verification
Legend
PrKA Alice's Private Key
PuKA Alice's Public Key
PuKB Bob's Public Key
SymK One time symmetric key
Hash Hashing alogorithm

Figure 4: Decryption and verification details with keys

6.0 Digital certificate contains-

A digital certificate contains among other things:
1) The CA's identity
2) The owner's identity
3) The owner's public-key
4) The certificate expiry date
5) The CA's signature of that certificate
With a certificate instead of a public-key, a recipient can now verify a few things about the issuer to make sure that the certificate is valid and belongs to the person claiming its ownership:
1) Compare the owner's identity
2) Verify that the certificate is still valid
3) Verify that the certificate has been signed by a trusted CA
4) Verify the issuer's certificate signature, hence making sure it has not been altered.
Bob can now verify Alice's certificate and be assured that it is Alice's private-key that has been used to sign the message. Alice must be careful with her private-key and must not divulge how to get to it; by doing so, she is enforcing one aspect of the non-repudiation feature associated with her digital signature. As will be seen in section 3.2, there is more to consider for effective non-repudiation support. Note that certificates are signed by a CA, which means that they cannot be altered.The CA signature can be verified using that CA's certificate.

6.1 Certificate validation added to the process

When Alice encrypts a message for Bob, she uses Bob's certificate. Prior to using the public-key included in Bob's certificate, some additional steps are performed to validate Bob's certificate:
1) Validity period of Bob's certificate
2) The certificate belongs to Bob
3) Bob's certificate has not been altered
4) Bob's certificate has been signed by a trusted CA
Additional steps would be required to validate the CA's certificate in the case where Alice does not trust Bob's CA. These steps are identical to the ones requires to validate Bob's certificate. In the example below, it is assumed that both Bob and Alice trust that CA.

In the Figure 5 below, a Certificate validation step is added to what is shown in Figure 3. Only the fields required for the validation of a certificate are displayed.

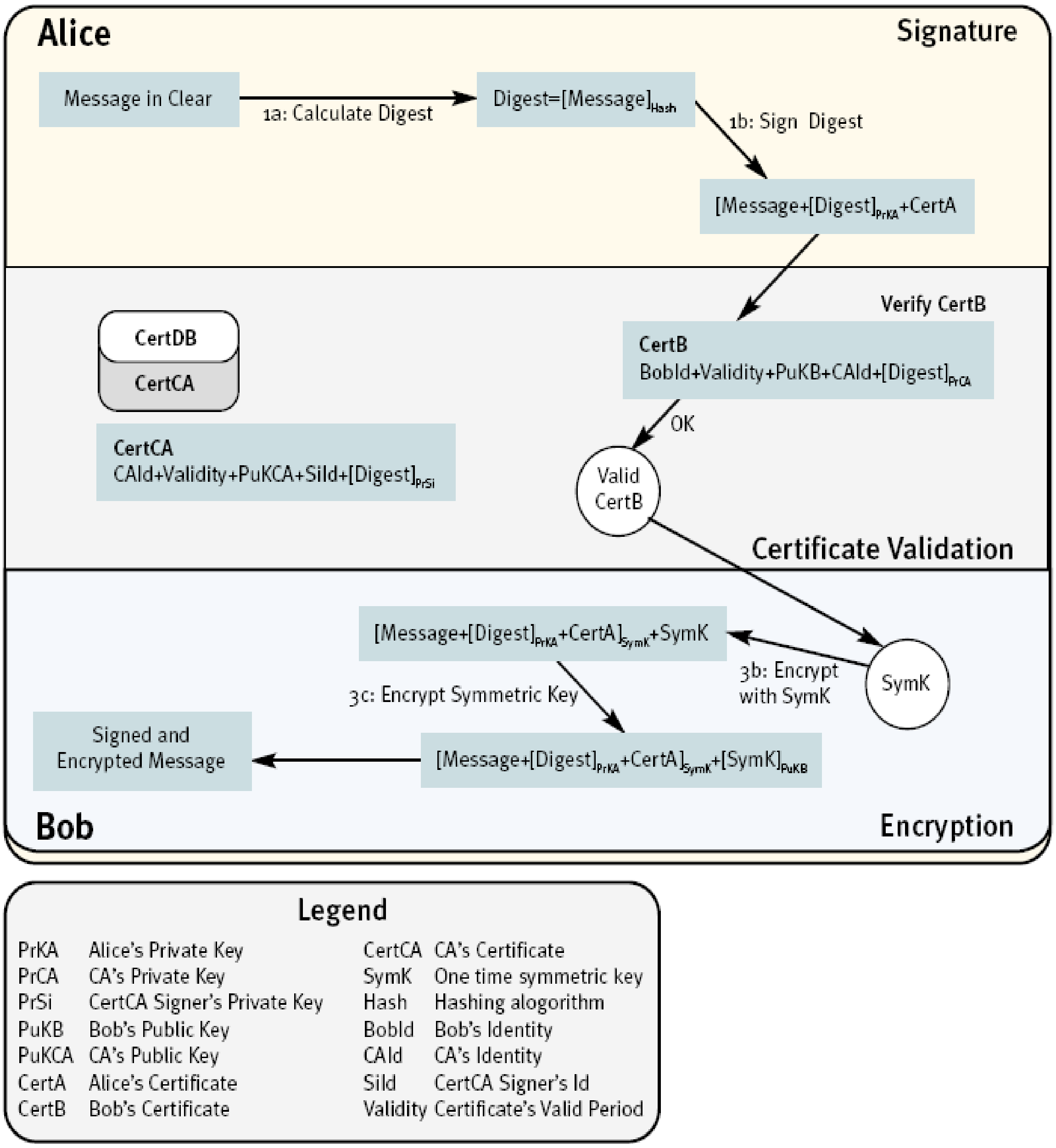
Alice wants to make sure that the PuKB included in CertB belongs to Bob and is still valid.

• She checks the Id field and finds BobId, which is Bob's identity. In fact, the only thing she really knows is that this certificate appears to belong to Bob.
• She then checks the validity fields and finds that the current date and time is within the validity period. So far the certificate seems to belong to Bob and to be valid.
• The ultimate verification takes place by verifying CertB's signature using the CA's publickey (PuKCA found in CertCA). If CertB signature is ok, this means that:
a) Bob's certificate has been signed by the CA in which Alice and Bob has put all their trust.
b) Bob's certificate integrity is proven and has not been altered in any way.

c) Bob’s identity is assured and the public-key included in the certificate is still valid and belongs to Bob. Therefore, Alice can encrypt the message and be assured that only Bob will be able to read it.

Similar steps will be performed by Bob on Alice’s certificate before verifying Alice’s signature.

Alice sending a signed and encrypted message to Bob revisted


Alice
Signature
Message in Clear
1a: Calculate Digest
Digest=[Message]$_{Hash}$
1b: Sign Digest
[Message+[Digest]$_{PrKA}$+CertA
Verify CertB
CertDB
CertCA
CertB
BobId+Validity+PuKB+CAId+[Digest]$_{PrCA}$
OK
CertCA
CAId+Validity+PuKCA+SiId+[Digest]$_{PrSi}$
Valid CertB
Certificate Validation
[Message+[Digest]$_{PrKA}$+CertA]$_{SymK}$+SymK
3b: Encrypt with SymK
SymK
3c: Encrypt Symmetric Key
Signed and Encrypted Message
[Message+[Digest]$_{PrKA}$+CertA]$_{SymK}$+[SymK]$_{PuKB}$
Bob
Encryption
Legend

PrKA	Alice’s Private Key	CertCA	CA’s Certificate
PrCA	CA’s Private Key	SymK	One time symmetric key
PrSi	CertCA Signer’s Private Key	Hash	Hashing alogorithm
PuKB	Bob’s Public Key	BobId	Bob’s Identity
PuKCA	CA’s Public Key	CAId	CA’s Identity
CertA	Alice’s Certificate	SiId	CertCA Signer’s Id
CertB	Bob’s Certificate	Validity	Certificate’s Valid Period

Figure 5: Signature and Encryption details with certificates

6.2 Digital Certificate Standards

X.509 Certificates

This standard is designed around a link between a digital signature key and a name in an X.500 directory that is hopefully sufficient to identify a person or an entity. It also embodies a capability for certificate extensions and these can be marked as critical or non-critical so that any extra features that they offer can be controlled. The aim here is to ensure that a verifying party will not accept a certificate as valid if there are critical extensions present that it cannot interpret. In contrast, a non-critical extension is not essential for validating the certificate and any inability to interpret it does not necessarily make the certificate invalid. More generally X.509 allows fields for which there is no universal definition of the semantics involved.

Although X.509 is widely used, it does have some potentially serious weaknesses including the assumption that it is always necessary to link a key to an identified person or entity; the assumption that, even when such a link is appropriate, it is always possible to uniquely identify the person or entity involved.

The problem here is that X.500 presumes the existence of a global directory structure in which every entity that needs to be identified can be traced somewhere within its hierarchy. In the real world, however, things are not so simple and there are many situations where this will not be possible or even desirable – the Central Intelligence Agency, for example, is unlikely to add the names of its employees to an open X.500 directory hierarchy.

However, a more serious problem is that the link between names and keys is much less important than it at first appears. As discussed earlier, a certified digital signature is much better viewed as a permission for the keyholder to use the key for a specific purpose.

6.3 Third Party Digital Signature Certification Authorities

Some governments are advocating the concept of licensed ‘third party digital signature certification authorities’ – Certification Authorities (CA) for short – as essential enablers for Electronic Commerce. The idea is that such companies will offer their clients digital signatures that are certified by these companies as belonging to these clients. The problem with this approach is that it is modelled on the use of digital signatures for ‘identity’ rather than ‘empowerment’ and this is not the most effective model for electronic commerce applications. Considering professional organisations, and using certified doctors’ digital signatures as an example, what would a third party CA contribute in comparison with the General Medical Council or the British Medical Association? They could never match the professional expertise of the latter bodies in a direct certification role and could not expect to certify doctors’ digital signatures to the same level of professional assurance. Hence, while signature certification controlled directly by professional bodies might reduce the probability of certifying bogus doctors’ signatures, it is hard to see this as the result of a third-party CA operation. Faced with using a medical prescription digitally signed with a signature certified by the General Medical Council or by ‘The Acme Signature Company’ it is not hard to see which of these most people would choose.For certification to be worthwhile, the certificate issuer should have something of value to offer in signing a certificate. In the credit card example, the certificate issuer is providing a guarantee while in the professional/organisational example this is a promise of performance with, possibly, a promise of compensation for error.

6.4 New Certificate Research

It is now being increasingly recognised that unique global names are not necessary to support certificates. For example, most people employ more localised ways of identifying those with whom they interact through direct meetings, through their immediate colleagues or using their (or their companies') address books.

In particular the improvements that result in considering certificates as empowerment mechanisms for digital signatures provide for direct links between keys and permissions without the complications that are involved in links to names and identities. As we have already seen in the examples given earlier, many real world examples map more easily onto certificates that empower keys rather than those that seek to link keys with names. Signatures represent the powers given to their owners to undertake actions so it very often makes sense to bind these actions to signature keys rather than infer the power of a signature owner through their identity.

7.0 Companeis provide digital certificates-

1) **RSA (www.rsa.com)**
RSA is the premier provider of security solutions for business acceleration. As the chosen security partner of more than 90 percent of the Fortune 500, RSA helps the world's leading organizations succeed by solving their most complex and sensitive security challenges. In September 2006, after over 20 years providing leadership to the security industry, RSA Security joined forces with EMC Corporation and Network Intelligence to form the Security Division of EMC.

2) **Thawte(www.thawte.com)**
Thawte is a leading global Certification Authority. Our SSL and code signing digital certificates are used globally to secure servers, provide data encryption, authenticate users, protect privacy and assure online identifies through stringent authentication and verification processes. Our SSL certificates include Wildcard SSL Certificates, SGC SuperCerts and Extended Validation SSL Certificates.

3) **Verisign** (www.verisign.com)

VeriSign (Nasdaq: VRSN) is the trusted provider of Internet infrastructure services for the digital world. Billions of times each day, companies and consumers rely on our Internet infrastructure to communicate and conduct commerce with confidence. VeriSign offerings include SSL, SSL Certificates, and digital content solutions, Extended Validation, two-factor authentication, identity protection, managed network security, public key infrastructure (PKI), security consulting, information management, and solutions for intelligent communications, and content.

8) Conclution-

Digital Signatures have potential uses in Electronic Commerce but attempts to apply them in ways that mirror written signatures are unlikely to be effective because this analogy is misleading.

‘Trusted Third Parties (TTP)’ as Certificate Authorities for digital signatures are often justified using an analogy with the role of the financial institutions in conventional commerce. In practice, however, this seems to be based on a misinterpretation of what th ese organisations provide since the trust relationships involved are only superficially three party ones. When analysed in more detail they are most often seen to be sequences of closed two-party relationships that combine to give the appearance of a relationship involving three-parties.

Because of this, it seems unlikely that open digital certificates have a significant role in Electronic Commerce. It also turns out that digital certificates are more effective as mechanisms for attaching permissions to digital signatures instead of names or identities (as the analogy with written signatures leads us to expect). And these properties in combination lead to uses of digital signatures, not as vehicles for identity, but rather as mechanisms that can represent the closed trust relationships on which commerce depends. Identity based digital signatures and the associated Certification Authorities have little immediate relevance in the development of Electronic Commerce. Put in the simplest terms, they are unnecessary for this purpose and seem more likely to delay the emergence of an electronic marketplace than they are to promote its development.

9) References-

1. PKI Implementing and managing E- security, Mcgraw-hill , 2001(Andrew nash, William duane ,Celia joseph, Derek brink).

2. E-Mail security how to keep your electroic messages private , John wiley & sons, inc.1995(Bruce schneir)

3. Curry, Ian, Entrust Technologies, "Getting Acquainted With Entrust/Solo and Public-key Cryptography", version 1.0, July 1997

4. Curry, Ian, Entrust Technologies, "Version 3 X.509 Certificates", July 1996, version 1.0 Branchaud, Marc, "A Survey of Public-key Infrastructures", Department of Computer Science, McGill University, Montreal, 1997

5. Curry, Ian, Entrust Technologies, "Key Update and the Complete story on the Need for Two Key Pairs", version 1.2, August 2000

6. RSA, "Intro to PKCS Standards", http://www.rsasecurity.com/solutions/developers/whitepapers/IntroToPKCSstandards.pdf

7. IETF/PKIX web site, http://www.ietf.org/html.charters/pkix-charter.html

8. www.rsa.com

9. www.verisign.com

10. www.thawte.com