Network Security, spring 2008

Final Project Report

X.509

This report is the final report for the Network Security course module of the LP 2 of the second semester in the Network Design course. The course topic for which this report is to be written is X.509. This project group involves 3 members and as such 6 pages of pure text is expected, plus visuals if we so choose.

First of all, we will start this report from what X.509 really is then, a little bit of history, its application areas, authentication procedures then a conclusion to summarize it all.

X.509 can be briefly said to be a specification which was published by the International Telecommunications Union (ITU) for the secured management and distribution of digitally signed certificates over secured internet networks. X.509 specifies standard formats for public key certificates, certificate revocation lists, attribute certificates and a path validation algorithm. It was developed as part of the X.500 directory service. A certificate is an electronic document that affirms the binding of a public key to an individual or entity. It allows for the verification of the claim that a specific public key belongs to a specific individual. It is issued and digitally signed by a trusted third party or Certificate Authority (CA).

It will be prudent to briefly throw light on the X.500 since it has a bearing on the X.509. The X.500 is a series of computer networking standard which covers electronic directory services. This encompasses the overall namespace and the protocol for querying and updating. X.500 comprises a wide range of services including name resolution, directory access protocol (DAP), and query resolvers, schemas, naming conventions and physical storage structures. It however didn't do well in the real world due to its cumbersomeness due to its monolithic directory view and the heavy load it placed on clients and OSI based protocols over TCP.

The date of birth for X.509 is 3rd July 1998 in association or as an extension of the X.500 standard. First published in 1988 as ITU-T 509 or ISO/IEC 9594-8 as part of the X.500 recommendations, it defined a standard format for a certificate. This was the first, as such was called version 1. However when the X.500 was revised in 1993, 2 more fields were added to the x.509. This resulted in the version 2 format of the X.509. Attempts to implement the versions 1 and 2 proved their inefficiency in the time, more obviously with the need for more fields to be added as a requirement. As a solution for the needed enhancements for the version 1 and version 2, the version 3 which is the current version was developed. By 1996 the basic version 3 standardization format has been completed.

We concentrate more on the version 3 of the X.509 since it is the best and more enhanced of the three versions. In general terms, users of a public key require that the corresponding private key holder or owner is the one they claim to be. This sort of trust is achieved through the use of public key certificates. These certificates are data structures that bind public key values to subjects. This binding is achieved by having a trusted Certificate Authority (CA) digitally sign each certificate. The signature of the certificate can be checked by certificate-using client for its genuineness. X.509 assumes a very stiff hierarchy of certificate authorities (CAs) in the issuance of certificates. Also it is

opposed to other web of trust models in that, only specialized CAs can sign and confirm the validity of others' key certificates.

For the X.509 certificate to fulfill its purposes, there are some protocols which should be used. These include:

Operational Protocols: Operational Protocols are required to properly deliver certificates and Certificate Revocation Lists (CRLs), as well as their status information to certificate using clients open request. These operational protocols make provision for variant means of certificate and CRL delivery. This includes delivery procedures like LDAP, HTTP, FTP, and X.500.

Management Protocols: These are the protocols which are needed to support online interactions and Public Key Infrastructure (PKI) and management entities. Some functions which are significant to management protocols include:

Registration: Registration is the initial contact that a user first makes itself known to a CA, either directly or through a Registration Authority (RA) prior to the issuance of a certificate or bunch of certificates to that user.

Initialization: This is the process of securely installing key components with appropriate relationships with the key pair in the Public Key Infrastructure (PKI). An example is where a client need to be initialized with the public key and other trusted information of trusted CA's which will be used in validating certificate paths. It is also noteworthy to know that a client typically needs to be initialized with its own key pair.

Certification: This process is where a CA issues a user with a certificate for his/her public key and returns the certificate to the use's client system or post the certificate in a repository.

Key pair recovery: This is where user client key materials can be backed up by a CA or a backup system for recovery purposes in case a user forgets password or loses a key chain file.

Key Pair Update: This is the regular update of key pair, with new ones and also when new certificates are issued. It is very reasonable to update keys since user systems either get new keys or even get newer key pair altogether.

Revocation request: This is when an authorized person advises a CA of an abnormal situation which is tantamount to a security breach which consequently requires a certificate to be revoked. This is a very important measure because when the security of the keys is compromised, drastic measures should be taken to reduce the adverse effects of the breach if not entirely eliminated.

Cross certification: This is the process whereby two CAs exchanges information in the hopes of establishing a cross certificate. A cross certificate here is certificate which is issued by one CA to another CA which contains a signature of a CA used for the issuance of certificates.

We also want to mention here that, we found out that, other ways other than online implementations of these functions exists. These are offline methods like hardware tokens used to achieve the same results.

We also found out that some of the above functions can be amalgamated into a single function so that multiple functions can be performed in one operation. An example is the registration, initialization, and certification functions being combined into one protocol exchange.

This certification system is a type of digital identity, with the help of which it can provide a means of proving your identity in electronic transactions, much like a passport does in face to face interactions. With a digital ID, you can assure friends, business associates, and online services that the electronic information they receive from you are authentic.

A Digital identity is issued by a certification authority and signed with the certification authority's private key.

A Digital ID typically contains the

Owner's public key

2. Owner's name

3. Expiration date of the public key

4. Name of the issuer (the certification authority that issued the Digital ID)

5. Serial number of the Digital ID

6. Digital signature of the issuer

The most widely visible application of X.509 certificates today is web browsers that support the secure socket layer protocol. Secure socket layer is a security protocol that provides privacy and authentication for your network traffic. These browsers can only use this protocol with web servers that support secure socket layer.

Other certification systems beyond X.509 could be are PGP or SKIP but none of them could be considered as superior to the others as their features as well as design and usage logic vary greatly.

Other technologies that rely on X.509 certificate include

1. Various secure email standards, such as PEM and S/MIME.

2. E-Commerce protocols, such as SET.

The most widely accepted format for digital ID's defined by the CCITT X.509 international standard, thus certificates can be read or written by any application complying with X.509.

The validity attribute comes has further options for an upper and lower date limit, which eventually decides the life of the certificate.

An identity Meta system for the internet would integrate all the different identity technologies into it X.509, which is used in smart cards, Kerberos, and Security assertions markup language, which is used increasingly in federation across the web.

A public key certification is a digitally signed statement from one entity, saying that the public key and some other information, of another entity has some specific value.

X.509 system also includes the method for CRL (certificate revocation list) implementations which is also an important aspect of certificate, should a certificate be compromised.

Certification structure of X.509

A X.509 version 3 digital certificate has 3 main variables

The certificate

The certificate signature algorithm and

The certificate signature.

The certificate is described by attributes such as

Version,

Algorithm ID,

Serial number,

Validity,

Issuer,

Subject,

Subject public key information.

Protocols that support X.509 certificates include:

1. IP Security
2. IP Security Transport layer security (SSL / TLS)
3. Secure multipurpose internet mail extensions (S/MIME)
4. Smartcard
5. SSH
6. HTTPS
7. EAP
8. LDAPv3

Certificate validation

To validate a certificate, we need another certificate, one that matches the issuer in the first certificate. First we verify that the second certificate is of a CA kind, that is it can be fact be used to issue other certificates. This is done by inspecting a value of CA attribute in the X.509 x3 extension section. Then we take the RSA public key from this CA certificate, use it to decode the signature on

the first certificate to obtain an MD5 hash, which must match an actual MD5 has computed over the rest of the certificate.

Authentication Procedures

There are three authentication procedures used for a number of applications for X.509 and all have the use for public key signatures. The three authentication procedures are as follows:

One-way authentication

Two –way authentication

Three-way authentication

The way that this procedure works is by two parties knowing each other's public key, either by obtaining it throw each other's certificates from a directory or obtained in the message that it is included in from each side.

One-Way Authentication

One-way Authentication is the process involving a single transfer of information from one user to another. This is done by:

Identifying the user from which the message was sent from and finding out if the message was indeed originated from the user.

Was the message from the user intended for itself?

Is it original? (Not sent many time before).

Within a message in also includes a number of things which are important. This is the *timestamp, nonce* and the identifier from the receiver and a signature of the sender's public key.

Timestamp: A Timestamp prevent message delay. It consists of generated time/ expiration time

Nonce: A Nonce can be used to detect or reused in replay attacks. The value of the nonce muct be the same as the expiration time of the message so the receiver can store the nonce until the it expires. Any new messages with the same nonce will be rejected.

Two-Way Authentication

Two-Way authentication is basically to verify the identity of each other. It is the same sequence from the One-Way authentication but from the other user (reversed).

Identifying the user from which the message was sent from and finding out if the message was indeed originated from the user.

Was the message from the user intended for itself?

Is it original? (Not sent many time before).

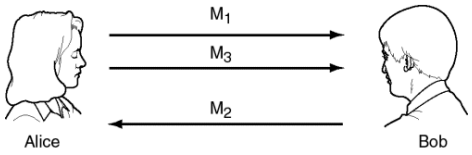
The reply message consists of a nonce from the other user for validation and also the timestamp and nonce from the sender.

Three-Way Authentication

The Three-Way Authentication is the last step and consists of a final message from the sender to the receiver and contains a signed copy of the nonce.

The reasoning for this is due to the fact that the timestamp do not need to be cheeked because of the nonce are cheeked by each side when they are echoed back for detection of replay attacks.

The diagram below shows the Authentication of all the steps amongst two sides.



M_1	E_{AR} (T, N, B)
M_2	E_{BR} (T', N, N')
M_3	E_{ar} (N')

E_{AR}	Encrypted with Alice's private key
E_{BR}	Encrypted with Bob's private key
T	Time stamp
N	Nonce
B	Bob's identity
T'	New time stamp
N'	New nonce

Key and policy Information

This category is basically an addition of information about the subject, issuer key and mainly the indicators of certificate policy.

A certificate policy is a set of rules which deals in security requirements. The areas in which it includes are:

Authority key identifier: Provides the means to identify a public key to verify the signature on the certificate/CRL. This is differentiated.

Subject key identifier: Provides the identification of the public key being certified.

Key usage: Applies and provides restrictions on the certified public key.

Private-key usage period: Provides the period of use of the private key corresponding to the public key.

Certificate policies: This means that the certificate is supported by other policies together with other information.

Policy mapping: This is the process by which a CA indicates that one or more of the policies can be same or equivalent to another policy in the CA´s domain.

Conclusion

X.509 is a strong authentication available in a variety of protocols and provides security plus administrative benefits and drawbacks. Thus X.509 provides good framework of services, technology, protocols, and standards that enable you to deploy and manage a strong information security system that is based on public key technology.

They are some benefits in the use of strong authentication in security and administrative aspects. The benefits should be considered in the application being deployed and the details of the configuration.

X.509 gives you the benefit to use and provide verification in a protocol between two applications and being a core element for definition of certificates with definition that defines a model of strong authentication.

Model of strong authentication has two important features which make X.509 a must have, It uses a minimum amount of protocol exchanges, thus making it important in high performance applications and does not depend upon data confidentially, giving it a significant performance advantage in applications that depend upon authentication, as they do not need data confidentiality.

Authors: Jerry Lartey

Vilhelm Wareus

V. Ramachandrareddy

References : Network Security Essentials (William Stallings)

http://www.ietf.org/rfc/rfc2459.txt

(www.microsoft.com/library/media/1033/technet/images/archive/security/prodtech/windows/iis/f1107_big.gif)

http://en.wikipedia.org/wiki/X.509

www.microsoft.com

www.isode.com