International Journal of Scientific Engineering and Applied Science (IJSEAS) – Volume-2, Issue-6,June 2016
ISSN: 2395-3470
www.ijseas.com

A Survey on Issues That Arises When Securing The World Wide Web (WWW) And Protocols To Secure WWW

Ashwini Rao
Student, CSE Dept, Jaipur National University,
Jagatpura , Jaipur,INDIA
ashwinirao1993@gmail.com

Abstract

This paper is to discuss the practical issues that arise when securing the access to the World Wide Web (WWW). Different protocols that are proposed to secure the WWW is briefly discussed and the current status is reviewed for the U.S. export regulations. An attack on SSL 2.0 is discussed which exploits some of the weaknesses in this protocol. The setup of a secure server with access control is explained.
The most important security services are confidentiality, integrity, authentication, and non-repudiation. The security services of communication system must be defined, while designing it. One of the technology that can meet these security services with its techniques and standards is the Public-Key Infrastructure (PKI). A PKI system should have a Certificate Authority (CA) for issuing public-key certificates. The main objective of this work is to design and implement a CA system that can create and assign public key certificates. Hence, the system enables secure communication and proper authentication.

1 Introduction

TheWorldWideWeb (WWW) is one of the primary reasons for the current success of the Internet. It has been so successful that many users think of them as similar. The WWW is a client-server technology: information is available on servers and is accessed by clients. The communication between these two parties is defined in the HyperText Transfer Protocol (HTTP) [2]
To be more useful today and in the future, the WWW needs to be secured. Services such as
entity authentication, data authentication, data confidentiality and nonrepudiation are essential for applications such as those used in electronic commerce or in an Intranet environment.

A lot of effort has already been done to secure the WWW. This paper intends to give an overview of the current situation with a focus on more practical issues. We start with an overview of the proposed protocols and the current status of the U.S. export regulations. Weaknesses in SSL 2.0 are exploited in an attack. We then focus on setting up a secure server and adding strong cryptography to export browsers. Finally, we look at the performance of the system.
The need for security becomes critical, since all information sent to the Internet is basically public. The ability to provide trust and confidence to transactions over the Internet might be the most critical element of security. Some of the few technologies that can accomplish to accommodate the scale of transactions across the Internet, include Public Key Infrastructure (PKI). PKI can be viewed as critical to the commercial sector and also to the government sector.

Therefore, we require many aspects for successful PKI, such as insurance and legal aspects, have been greatly improved. Without either having to know or trust the other party, it possible for two parties to communicate securely through the Public-key system. However, this is only possible because a third party that the other parties trust identifies them, and certifies that their keys are genuine [22].
This third party is called the Certificate Authority (CA). CA guarantees that they are who they claim to be. The CA does this by registering each user's identification information, and issuing them with a set of Private keys and a set of Public Key Certificates.

2 Protocols

Currently we have four proposals for providing security services to the WWW:
- Netscape's *Secure Sockets Layer* (SSL) [3];
- Microsoft's *Private Communication Technology* (PCT) [4];
- *Secure HyperText Transfer Protocol* (S-HTTP) [5], from Enterprise Integration Technologies and Terisa Systems;
- *Transport Layer Security* (TLS) [6], an IETF working group.

All four protocols provide entity authentication, data authentication and data confidentiality. In contrary to SSL and PCT, which are both situated in the transport layer, S-HTTP is situated in the application layer, and can thus offer nonrepudiation of origin (in a legal sense). The two protocols PCT and SHTTP have not been a success. SSL has become a de facto standard on the Internet. The recent version of SSL is 3.0 and it has a number of improvements [7, 8] over release 2.0 [9] (see 2.2). There are free implementations available such as SSLeay [10] and SSLRef [11].

IETF is a successor to SSL. It has organized a working group TLS (*Transport Layer Security*) that has adopted SSL 3.0 in its initial release of TLS 1.0. The services offered by TLS is same as that of SSL, but there are some minor differences. TLS will probably replace SSL in the future as the first TLS based products are already being implemented.

2.1 Export regulations and limitations

The U.S. export policy has serious implications on the level of security that can be obtained by the most popular WWW browsers and servers. Until January 1997, the strongest cryptography that U.S. companies were allowed to export was limited to a 40 bit
security level. Thus, the international versions of Netscape Navigator and Internet Explorer (the two most popular browsers) were limited to 40 bit security. '40 bit security' applies to the maximum length of the key used for symmetric encryption. Maximum of 512 bits of Keys used for asymmetric encryption (key management). The maximum level of security that can be exported is now 56 bits. There are unfortunately certain restrictions. The export of 56 bit is only allowed if the vendor commits to implementing key recovery by January 1, 1999, and a governmental approval is needed for each customer the product is transferred to. As the export of 56 bit still implies such restrictions, the IETF-TLS working group still defines 40 bit in the export ciphers used in TLS [6]. We can also offer high level security (128 bits or more) to international companies, if it can be shown that it is necessary (e.g., financial institutions, banks). In that case, strong cryptography can be enabled in export browsers by a Global Server Certificate from VeriSign.

It is obvious that export products do not fully provide data confidentiality as their encryption routines are limited to (practically) 40 bit [12]. The quality of the user authentication service should not be influenced, although export browsers are unable to generate keys that are used for client authentication longer than 512 bit (though there is a possibility of generating longer keys with another program and import them in your browser). When 128 bit (MAC) keys are exchanged using 512 bit RSA, they do not represent a complete 128 bit strength, as dividing 512 bit RSA is easier than a 128 bit exhaustive search (but still much more difficult than 40 bit).

2.2 Security problems

We can categorize these problems into three ways:

Security flaws in the protocol:

Release 2.0 of the SSL protocol contained several security flaws [7, 8]. These flaws were solved in the SSL 3.0 specification:

- SSL 3.0 uses a HMAC-like [13] construction which is more secure than the MAC construction in SSL 2.0.
- In SSL 2.0, the MAC secret and the encryption key are the same. It is more secure to use different keys. Moreover, also the export restrictions limits the MAC secret.
- The integrity of the handshake messages is not ensured in SSL 2.0.
- The export regulations restrict the length of the 'key' used for encryption. In SSL 2.0, this is applied to the exchanged *Master Key* (which is used during a session). Also note that client only can choose the *Master Key* in SSL 2.0. In SSL 3.0 the restriction applies to encryption keys (used in one connection) derived from a large *Pre Master Key*. This *Pre Master Key* will derive a *Master Key* with input from both parties.
- In SSL 2.0, the client authentication token is not dependent on the global handshake.

The last flaws result in a possible attack scenario where an attacker can authenticate to the server as another client if this attacker has discovered the *Master Key* used in the session between this client and the server [7, 8]. As per the research, it is found that many servers still use SSL 2.0 (e.g., Amazon.com).

Table 1. Notation

Symbol	Significance
MK_{CS}	40bit Master Key
NC	new connection request of client
C_C	Challenge
CI_S	Connection ID
CC_S	Certificate Challenge
$Sign_C$	Client's digital signature

The attack is visualized in Figure 1 and the notation is described in Table 1. Suppose that in a 128 bit session, WWW-server offers more services to clients where the client authenticates. An international client has set up a 40 bit secured connection to the server. An attacker can then obtain the *Master Secret* by an exhaustive search on this 40 bit secret.

Within the current session, the client asks for a new connection. The session ID and some *Challenge Data* to the server is sent to this end .

The attacker intercepts this request message and replaces the 40 bit algorithms, that the client asks for, with 128 bit algorithms. The same *Master Key* is used because the connection is made in the same session.

The server receives the request and hence the request is granted. It returns a *Connection ID* to the client. All parties involved (client, server, attacker) can now calculate the new connection keys (40 bits for the client, and 128 bits for the attacker and server), because these only depend on

the *Master Secret*, the *Challenge Data* and the *Connection ID*. The server now asks the 'client' for authentication by sending it the *Certificate-Challenge Data* encrypted to the attacker. The attacker send this packet to the client.

The unsuspicious client then sends its *Certificate* and a signature on a message to the server. This message depends on the *Certificate-Challenge Data*, the *Server Certificate*, the *Master Key*, the *Challenge* and the *Connection ID*. Because this signature does not depend on the actual key strength or the algorithms used, the attacker is able to replay this token to the server to authenticate itself as the client.

In this way, the server offers the extra services to the attacker. Moreover, the attacker also masquerades effectively as the client. At the same time, the attacker can act as the server (provided that it intercepts all messages sent by the client to the server).

In SSL 3.0 this attack does not succeed because the client also signs all previous handshake messages (thus also the one in which it requests a 40 bit connection) in the last step.

Implementation problems:

On the whole SSL 3.0 is quite secure [14]. A bad implementation however can also cause problems.

Especially the generation of

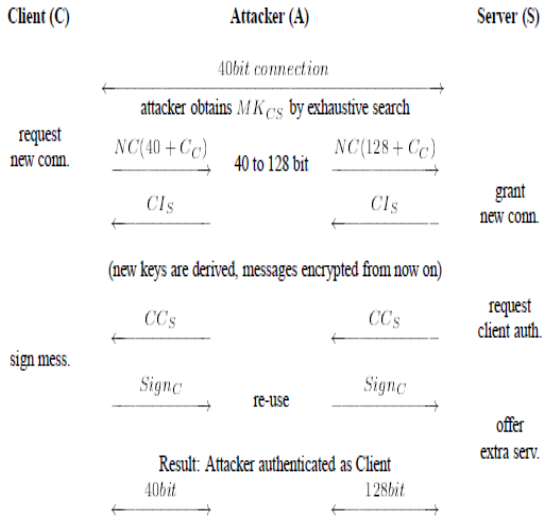

Figure 1. Possible attack scenario in SSL 2.0

random numbers, essential to all the protocols mentioned above, is a difficult task. An earlier version of the Netscape browser had a security weakness due to a bad implemented random generator [15]. Versions of Internet Explorer also appear to have bugs.

Practical problems. Not only the protocol and the implementation are important. The security of the system can be questioned if other programs (e.g., viruses) or users can read the part of the memory allocated for the temporary storage of decrypted messages or secret keys. During the installation of the secure server, security holes can also arise.

3. Server

In this section, we discuss the server side.

3.1 Apache-SSLeay based secure server

Commercially available software can be chosen for setting up a secure server. It is also possible to setup a secure server based on source code that is publicly available and that is not limited by any export regulation. This can be done by using the SSLeay library [10], which can be integrated in

Apache, a plain http server of which the source code is also available. The Apache-SSLeay [16] approach has become very popular and is used more and more in both academic and commercial environments. Note that research
shows that Apache is used by 50% of all web sites.

3.2 Obtaining a server certificate

A certificate is needed before the server can actually offer services on the World Wide Web. Certificates normally have to be obtained from a Certification Authority (CA).
The easiest way to get a server certificate is to create one not (and should not be) recognized and trusted by a standard browser. It is however possible to add this kind of certificate to the list of trusted certificates. A cautious client should only do this if it has more background information of the server, as in an Intranet situation.

Another way to obtain the certificate is to get one from an official CA such as Verisign or Thawte. By default, the browsers contain the certificates of these CAs, and the issued
server certificate can thus be verified. This method is preferred, especially if worldwide access to the server is needed. A disadvantage is the long certification procedure and the cost of the certificate.

3.3 Access control

It is possible to have the access to some of the server pages restricted. This is useful if the information on these pages is confidential and should not be available to the general
public, or if the client has to register before he is allowed to access the pages.
If a page with restricted access is requested, the clients need some kind of mechanism to prove their identity, and the server has to be able to verify it. Several mechanisms can be used for this purpose:

Username/Password:
The username/Password technique is already provided by normal servers. In most cases, the password is sent from client to server in clear text (more precisely, base64 encoded). When using SSL, the communication is encrypted, including the password. This technique remains however vulnerable to password guessing and dictionary attacks.

X.509 client certificates:
Access can also be granted based on SSL's client authentication. The server administrator can act as a CA and issue certificates only to those people who may access the page. It is also possible to use
certificates of an existing CA and to verify the distinguished name they contain.

Attribute certificates:
Another possibility is to use the Role Based Access Control (RBAC) mechanism. Access is granted to people based on their role (e.g., secretary, accountant, manager, etc). Attribute certificates are credentials that prove which roles one is allowed to act as. They are usually administered centrally and are issued by an Attribute Server. An alternative could be the use of proprietary
extensions in the X.509 certificates. One of the advantages of this system over the X.509 based certificates is that a certificate revocation mechanism is not needed as attribute certificates are issued with a limited lifetime. Manageability is also an advantage of RBAC as access restrictions only have to be defined for each role and not for each user.
Role Based Access Control is implemented for example in the SESAME environment [17], and it is used in Trusted Web [18].

3.4 Security of the server

During the installing of server, number of problems can be arisen. The implemented access control is only performed when accessing the server via the WWW. Everybody who can login to the machine the server is running on, might access the server pages locally. They might also try to locate the server's private key. Therefore, the server's private key should be readable only for the https daemon. The key should also be encrypted, so that a (non-trivial) password has to be provided at the startup of the server. These precautions
are all necessary to ensure the server's private key is indeed private. Generating the keys yourself when requesting the server certificate is of course the primary condition to be sure of the secrecy of your server's private key.

4 Public key cryptography

Public key cryptography were introduced in 1976 to solve the key management problem of symmetric cryptography. Public key (or asymmetric) cryptography use two different keys, "private" and "public" keys. The private key is kept confidential while the public one is published in a common directory so that everyone can access it. The key pair are relative to each other. That is, when a message is encrypted with one of them, it can only be decrypted with the other. Deriving the private key from the public one is mathematically infeasible. So that, the sender (Alice) encrypts her message with the recipient's (Bob's) public key and the receiver can decrypt the message with his corresponding private key. Public key systems are slower than symmetric key systems because of the large key length and the nature of the public key algorithms. Hence, it is not necessary to encrypt long messages with public key cryptography. Instead of this, the key used in the symmetric cryptography is encrypted with public key cryptography, then the message is encrypted with a symmetric key system.
Among the widely implemented public key systems are the RSA and the El Gamal systems. In recent years, Elliptic Curve
Cryptosystems (ECC) have also been emerged. Both the industry and the international standards community have widely adapted the RSA system for public key cryptography implementations. However, in this work, we will use the ECC because it achieves the same security level as other peers along with a much smaller key length. In fact, ECC presents some key attributes that are truly important in scenarios where some resources are limited. For example: processing power, storage space, bandwidth, and power consumption [23], [24]. ECC was discovered in 1985 by V. Miller as an alternative method for public key cryptography. During that time, it was very difficult to perform the necessary calculations. Further, implementations became much more efficient. Hence the performance of ECC to take the same amount of time as implementations of integer factoring schemes for the same number of bits. This in turn implies in a reduction of cost, size, and processing time because elliptic curves require fewer bits for the same security level. Over recent years, as the bit length for secure RSA use has increased, a heavier processing lean on applications using RSA has been observed. This burden can be serious, especially it can affect the
e-commerce sites that conduct large number of secure transmission. Hence, ECC are gaining more and more attraction [25], [26]. The ECC is unlike earlier cryptosystem, an elliptic curve works with a finite Abelian group formed by the points on an elliptic curve defined over a finite field. ECC can be used for key distribution,

encryption/decryption, and digital signature algorithm (DSA). We use the key distribution algorithm to share a secret key for symmetric cryptography, the encryption/decryption algorithm is used for confidential communication, and the DSA is used for authentication and validating the integrity [27].

5. Public key infrastructure

Public Key infrastructure (PKI) is a popular encryption and authentication approach which is the combination of software, encryption technologies, and services that enable enterprises to protect the security of their communication and business transactions on networks. PKI enables users to communicate securely regardless of the distance between them using a commonly shared certificate known as the chain of trust.

The users will not have any preexisting relationship. PKI provides the basic services of confidentiality, data integrity, authenticity,

and non-repudiation. PKI permits these by offering a way of identifying and trusting another Internet user, through the use

digital certificate. Digital certificate contains the Internet user's name and some other credentials (The content of digital certificates depends on the organizational policies and some other private issues). A digital certificate can also be used to verify a digital signature, which can be attached to e-mail messages or other types of electronic messages. This signature is created using public key cryptography [28], [29].

In general, PKI system mainly consists of a CA that accepts user requests for a certificate. The CA also acts as the authority, which issues and manages security credentials and

public keys for message encryption. Depending on application of the system the organization of the components of any specific PKI system can vary accordingly.

These components will have the details: end-users, Registration Authorities (RA), Certification Authority (CA), Public Key Certificates (PKC), Certificate Repositories (CR), Certificate Policies (CP), and Certificate Practices Statement (CPS) [30]. Below is a summary of the functionalities of these components:

1. The End-users: The end-users are the people who are using the system. They are considered as the key element of the system where in the system will include application, policies, and practices those are built up for them. In general, the end-user may request certificates from a CA, receive the certificate from the CA, and then use the certified keys and certificates in PKI enabled application services.

2. Registration Authorities: A Registration Authority (RA) is common component of a PKI. It's not mandatory to use it. An RA perform some of the administrative tasks that a CA would normally undertake. The objective of an RA is to verify the identity of end user's and determine if an end entity is entitled to have a public key certificate issued. To avoid the complexity of tasks, many PKI implementations separate the operations performed by the CA and the RA.

3. Certification Authority (CA): It is a trusted authority in a network that issues and manages security credentials and public keys for message encryption. As part of a PKI, a

CA checks with a registration authority to verify information provided by the requestor of a digital certificate. If the requestor's information is successfully verified by RA, then the CA will be allowed to issue a digital certificate. Note that the CA is responsible for the distribution and revocation of the certificate. Depending on

the PKI implementation, the 136 certificate might include the owner's public key, the expiration date of the certificate, the owner's name, and other information about the public key owner [31].

4. Public Key Certificates (PKC): A Public Key Certificate or a "digital certificate" is an electronic set of credentials for an individual that offers proof of identity. The digital certificate contains information like the name, organization, expiration date, and the subject's public key and a digital signature of a trusted third party. Any entity that wants to use any certificate, first checks the validity of the digital signature contained in it. The validity is indicated through the expiration date. There are many types of certificate, such as X.509 Public Key Certificates, Simple Public Key Certificates (SPKC), and Pretty Good Privacy (PGP) Certificates [32], [33]. Every certificate types have their own data structures. In most of the PKI systems it has been used widely and due to this our work has adopted third Version of X.509 public key certificates.

Fig. 1 illustrates the structure of an X.509 v3 certificate [34]. However, it is important to note that there is no one single definition of a public key certificate defined in the IETF standards. Vendors and integrators have their own ideas on what extensions and particular data an X.509 certificate should contain. Hence, it is preferred that each organization evaluates its business needs relative to the constructs of the public key certificates that it wishes to issue.

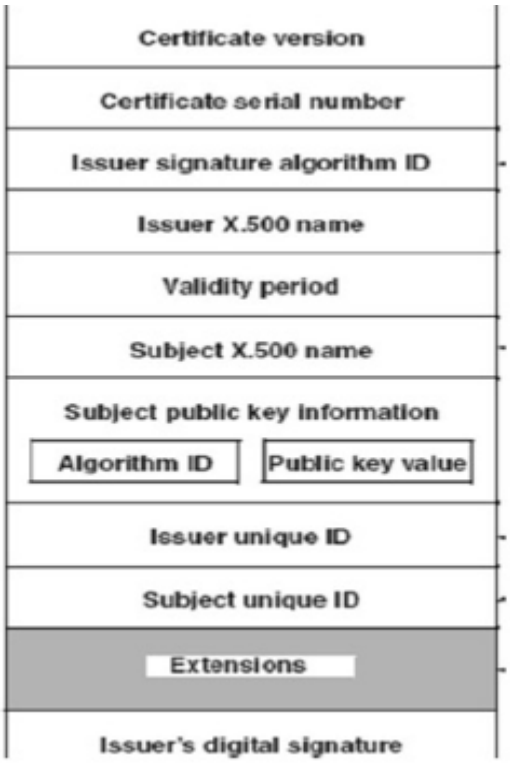

Figure 2: X.509 V3 certificate [34]

5. Certificate Repositories (CR): A certificate repository (or a certificate directory) is an optional but common component of a PKI. It can simply be posted on a public web page, and put in a database or some other form. Besides the certificates, other PKI related information such as Certificate Revocation Lists (CRL) can be stored in the repository [35].

6. Certificate Policy (CP): It is a documented set of rules and commitments made by a CA to indicate the applicability of a certificate to a particular group of users or set of applications. The main purpose of CP is to determine the security policy that is followed by a certification organization. Also, it can be used as a reference for other

organizations that need to establish a domain-trust relation with this organization.
7. Certificate Practices Statement (CPS): The CPS is a statement that a CA employs in issuing public key certificates. The CPS document list out all the procedural and operational practices of a PKI [34].

6 Conclusion

When securing the WWW, we can notice many issues. In this paper we have discussed about those issues. The current status of protocols and export regulations also reviewed in this paper. An attack on SSL 2.0 indicates how some of its weaknesses can be exploited. With the help of some freely available software it is possible to set up a secure server. In this modern age of linking every computer to a network, there is a clear need for an access control mechanism. In an Intranet situation, a solution based on attribute certificates would enhance the overall security and manageability of the system.
Besides the basic security requirements, the developed system can use an approach that can contribute in facilitating the revocation of the certificates. It should also give these certificates additional security/performance advantage by using the Elliptic Curve Cryptography (ECC) instead of the RSA cryptography**.**

References

[1] Baltimore Technologies. J/SSL. http://www.baltimore.ie.
[2] T. Berners-Lee, R. Fielding, and H. Frystyk. Hypertext Transfer Protocol – HTTP/1.0, May 1996. RFC1945.
[3] A.O. Freier, P. Karlton, and P.C. Kocher. The SSL Protocol Version 3.0, March 1996. Internet Draft.
[4] D. Simon. Microsoft Corporation's PCT protocol version 2.0, April 1996. Internet Draft.
[5] E. Rescorla and A. Schiffman. The Secure Hypertext Transfer Protocol, May 1996. Internet Draft.
[6] T. Dierks and C. Allen. The TLS Protocol Version 1.0, November 1997. Internet Draft.
[7] J. Claessens and J. Uyttenhove. Beveiliging van het World Wide Web. Thesis, K.U.Leuven, 1997.
[8] M. Vandenwauver. Practical Network Security Aspects, 1998. PhD. Thesis, K.U.Leuven.
[9] K. Hickman and T. Elgamal. The SSL Protocol, 1995. Internet Draft.
[10]E.Young.SSLeay. http://www.ssleay.org .

[11]Netscape.SSLRef.http://home.netscape.com/.

[12] M. Blaze, W. Diffie, R. Rivest, B. Schneier, T. Shimomura, E. Thompson, and M. Wiener. Minimal Key Lengths for Symmetric Ciphers to Provide Adequate Commercial Security, January 1996.
[13] M. Bellare, R. Canetti, and H. Krawczyk. Keying Hash Functions for Message Authentication. In N. Koblitz, editor, *Advances in Cryptology, Proceedings 790of Crypto '96 - LNCS 1109*, pages 1–15. Springer-Verlag, 1996.
[14] D. Wagner and B. Schneier. Analysis of the SSL 3.0 protocol. In *The Second USENIX Workshop on Electronic Commerce Proceedings*, pages 29–40. USENIX Press, 1996.
[15] I. Goldberg and D. Wagner. Randomness and the Netscape Browser. *Dr. Dobbs Journal*, January 1996.

[16]B.Laurie.Apache-SSL. http://www.apache-ssl.org.

[17] M. Vandenwauver, R. Govaerts, and J. Vandewalle. How Role Based Access Control is Implemented in SESAME. In *Proceedings of the 6-th Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises*, pages 293–298. IEEE Computer Society, 1997.

[18] Secure Solutions Experts. TrustedWeb, Advanced Intranet Security. http://www.trustedweb.com

[19] IAIK. IAIK Java Security. http://jcewww.iaik.tu-graz.ac.at/.

[20] C2Net. Safe Passage Web Proxy. http://www.c2.net/.

[21] MarketNet. Internet WorkHorse. http://www.mkn.co.uk.

[22] W. Stallings, *Cryptography and Network Security Principles and Practices*, Prentice Hall, 2003.

[23] N. Potlapally, S. Ravi, A. Raghunathan, N. K. Jha, "A Study of The Energy Consumption Characteristics of Cryptographic Algorithms and Security Protocols", IEEE Transactions on Mobile Computing, Vol. 5, No. 2, 2005.

[24] W. Rao, Q. Gan, "The Performance Analysis of Two Digital Signatures Schemes Based on Secure Charging Protocol", International Conference on Wireless Communications, Networking, and Mobile Computing, Vol. 2, September 2005.

[25] K. Klemetti, *Authentication in Extranets*, MSc. Thesis, Helsinki University Of Technology, July 2001.

[26] A. Jurisic and A. Menezes, *Elliptic Curves and Cryptography*, 2005.

[28] Certicom Corp., "The Elliptic Curve Cryptosystem", A Certicom White Paper, 1998.

[29] H. Ray, "Technological infrastructure for PKI and digital certification". J. Computer Communications, University of Canterbury, Vol. 24, No. 14, 2001.

[30] L. M. KohnFelder, *Towards a Practical Public-key Cryptosystem*, B.S. Thesis, Massachusetts institute of technology, May 1978.

[31] S. Kiran, P. Lareau and S. Lloyd, "PKI Basics - A Technical Perspective", November 2002.

[32] B. Lee, "Certificate authorities: Who Do You Trust?", Data Communications, vol. 27, no. 4, March 1998.

[33] E. Yildiz, *A Proposal for Turkish Government Public Key Infrastructure Trust Model*, MSc Thesis, December 2001.

[34] P. Zimmermann, "The official PGP User's Guide", Cambridge, MA, MIT Press, 1995.

[35] S. Choudhury, K. Bhatnagar, and W. Haque, *Public Key Infrastructure Implementation and Design*, M&T Books , 2002.

[36] D. Kopparhed, "A Secure Model for Certificate Distribution and Management for Dynamic Access Control", August, 2007.