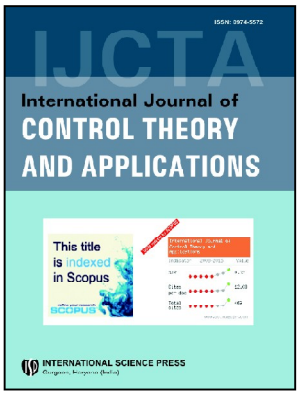

International Journal of Control Theory and Applications

ISSN : 0974-5572

Volume 9 • Number 50 • 2016

Elliptic Curve Cryptography and its Application in the Secure Socket Layer/Transport Layer Security Protocol

M. Cimi Thomas[1*] and S. Sheeja[2]

[1*] *Research Scholar, Department of Computer Science, Karpagam University, Coimbatore, India*
[2] *Associate Professor and Head, Department of Computer Applications, Karpagam University, Coimbatore, India*
[1*]*Corresponding Author Email: cimithomas@yahoo.co.in*

Abstract: The web applications and online transactions are popular and growing tremendously, so large volume of personal and critical information is transmitted over the network. Ensuring the security of the information is vital for the growth of web applications. Various security protocols are available and they use different encryption algorithms for message encryption, key agreement and for authentication. In this paper, Elliptic Curve Cryptography (ECC) and its application in the Secure Socket Layer (SSL) is reviewed.ECC is an asymmetric key encryption method which offers high security with smaller key sizes and it is used in the security protocols without degrading the performance of the web servers. The performance improvement achieved by using ECC in Secure Socket Layer protocol is examined and a study is done about the side channel attacks on the elliptic curve implementation in Open SSL.

Keywords: encryption, decryption, elliptic curve cryptography, SSL, TLS.

1. INTRODUCTION

In the present digital world large amount of personal and sensitive information are transmitted over the internet. Ensuring the security of the data during storage and transmission is considered as a major concern. Encryption algorithms are used to secure the personal data from the unauthorized attacks and they are the building blocks of security protocols like SSL. Security protocols use symmetric key encryption algorithms to encrypt data transmitted over the network and asymmetric key encryption algorithms for authentication and key agreement. Most of the E-commerce websites and online banking sites provide secure transactions using the SSL. Though SSL provides adequate security, SSL processing is a time consuming factor and causes additional burden for the web servers resulting in the slow response time. RSA [1] is the commonly used public key encryption algorithm in security protocols. But recently Elliptic Curve Cryptography (ECC) has become popular and has gained interest among the researchers and is being used in the security protocols by replacing RSA. The attraction of ECC is that it offers high security with smaller key sizes. When used in the security protocols, ECC improves the performance of the web servers and offers memory and bandwidth savings. In the paper a study of Elliptic Curve Cryptography is conducted and its application in SSL is analyzed. The rest of the paper is discussed as, the Section 2 provides

an overview of ECC, SSL/TLS protocol and their impact on the web servers is explained in the Section 3, the Section 4 shows the application of the ECC in SSL and the Section5 describes various side channel attacks successfully done on the elliptic curve algorithms in Open SSL.

2. ELLIPTIC CURVE CRYPTOGRAPHY OVERVIEW

Elliptic curve cryptography is an asymmetric key encryption algorithm developed by Victor Miller[2]and Neil Koblitz [3].An elliptic curve over a field F is the curve given by the equation $y^2=x^3+ax+b$ and the point *O,* which is named as the point at infinity or the zero point.

The elliptic curve arithmetic is based on the scalar point multiplication which involves the multiplication of a point of the curve 'n' times to get a different point on the same curve, where n is a scalar. Scalar multiplication is performed by different combinations of point additions and point doublings. These operations can be explained geometrically as below.

Point Addition

Let the elliptic curve has two different points P and Q. To perform the addition of these points, a straight line is drawn between them as shown below. Let –R be the point at which the straight line intersect the curve .The result of the addition is the point R which is the reflection of this point about the x axis.

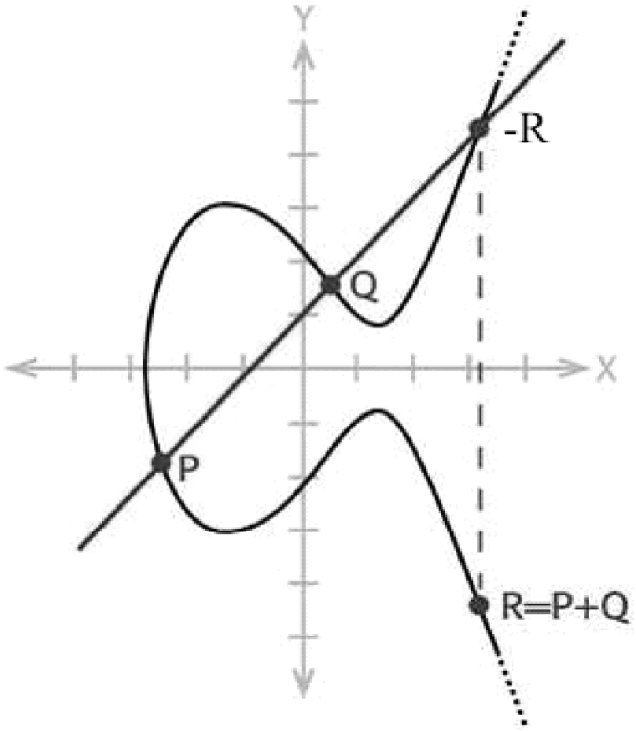

Figure 1: Illustration of Addition of Two Different Points

Point Doubling

Let P be a point on the curve, addition of P to itself is done with the help of a tangent line drawn to the curve at P. Let –R be the point where the tangent line intersects the elliptic curve. The result of the doubling is the point R, which is the reflection of the point –R about x axis.

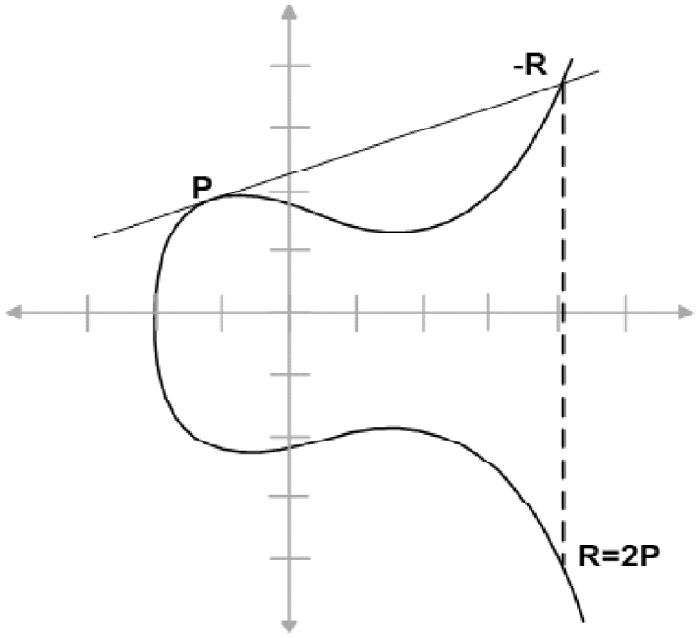

Figure 2: Illustration of Point Doubling

For the cryptographic schemes, elliptic curves over two finite fields are used. They are prime field Fp, where p is a prime and binary field $F2^m$,where m is a positive integer. An elliptic curve over the Fp is defined as a prime curve and elliptic curve over the $F2^m$ is defined as a binary curve.

Prime curves are more suitable for the software implementations and the equation of the elliptic curve over Fp is defined as :$y^2 \bmod p = (x^3+ax+b) \bmod p$ where $(4a^3+27b^2) \bmod p \neq 0$ and x , y, a,b ª [0,p-1].

To construct a public key cryptographic system using the elliptic curves, choose a secure elliptic curve E and a point A on the curve. Generate a large random integer 'n' which is the private key. The public key B is computed by nA, where B is another point on the elliptic curve. Once the key pair (n,B) is generated, it is used for the encryption/decryption, digital signatures and key management systems.

Public key cryptosystems are constructed relying on the hardness of the certain mathematical problems. RSA relies on the integer factorization problem. The security of the ECC rests on the hardness of the discrete logarithm problem over the points on the elliptic curve. The Elliptic Curve Discrete Logarithm Problem (ECDLP) states that given a base point A and a point B=nA lying on the curve, it is hard to determine n.

Elliptic Curve Diffie Hellman (ECDH)[4] is the elliptic curve analogue of Diffie Hellman Key Exchange. In ECDH key exchange, the communicating parties X and Y agree to use the same curve parameters .They generate the private keys $n_X$and $n_Y$and the public keys $B_X=n_X.G$ and $B_Y=n_Y.G$ where G is the base point. Public keys are exchanged and each party multiplies is private key with other's public key to arrive at a common shared secret.$n_X.B_Y=n_Y.B_X= n_X*n_Y*G$.

3. SECURE SOCKET LAYER (SSL) /TRANSPORT LAYER SECURITY (TLS)

SSL is commonly used protocol to secure the information transmitted over internet. SSL[5]is considered as an additional layer between the application layer and transport layer in the TCP/IP protocol suite and provides confidentiality ,integrity and authentication of the data exchanged .Transport Layer Security(TLS) is the successor of SSL, internet standard version of SSL and is similar to SSL v3. TLS1.2 is the current version of TLS.

SSL/TLS has two main sub protocols.SSL processing starts with a Handshake protocol followed by the Record Layer protocol.SSL/TLS handshake is needed to initiate a logic connection and to establish the security capabilities of the client and server. During the handshake the server and the client establish their authenticities by exchanging the digital certificates. After the authentication a master secret shared by the client and server is established with the help of asymmetric key cryptographic algorithms. The record layer protocol in the SSL/TLS comes in to picture after a successful handshake is completed between the client and the server and provides confidentiality and integrity to SSL/TLS connections. The record layer protocol calculates keys for symmetric encryption from master secret and uses them for generating message authentication codes and also to encrypt data using faster symmetric key encryption algorithms.SSL/TL Suses a cipher suite which includes asymmetric key encryption algorithms for authentication and key exchange in the handshake phase, symmetric key encryptions to encrypt data in data transfer phase and hash functions to ensure integrity in handshake phase and data transfer phase.

SSL and Web Servers

SSL protocol provides necessary security for the online transactions but processing time is more. Secure web servers run slower than non-secure ones. This high processing time creates delay in loading the web pages and causes frustration to online shoppers resulting in revenue loss for E-commerce websites. Increased processing time is due to the cryptographic algorithms used in the SSL. Several research papers are published by analyzing the performance of the secure web servers and the effect of SSL on internet servers. One of the earliest works in this area is published by K Kant[6],where an analysis of web server performance and the impact of SSL on web servers in terms of various parameters are done and the study had concluded that the inclusion of SSL increases the computational cost of online transactions.

In[7], a detailed study is conducted on the SSL processing in the HTTPS web server transactions. The study is conducted on Open SSL with the cipher suite RSA-3DES-CBC-SHA1. RSA is used for asymmetric encryption, 3DES in CBC mode for the symmetric encryption and SHA1for hashing. It is found out that SSL processing takes 71.6% of the transaction time and this is mostly due to the cryptographic operations. The non-cryptographic portion takes only a fraction of time. While analyzing the processing time needed for crypto operations in SSL handshake, it is found that asymmetric key encryption using the RSA takes 90% of the total cryptographic processing time for small files (banking transactions).

The impact of the SSL on web server performance is studied by the author in [8]and has concluded that the SSL degrades the performance of web servers. The increase of total processing time is mainly due to the public key encryption using RSA 1024 bits in the handshake phase. In the recent years key size of RSA is increased to withstand security attacks.

In order to resist the future security attacks ,NIST has given guidelines to migrate from RSA 1024-bits key to RSA 2048- bit keys as of 1/1/2014[9].This increase in the key size will affect the performance of the web servers. Elliptic curve cryptography which provides an equivalent security with smaller keys can be used in handshake phase of SSL. The smaller key sizes increase the speed of computations, consumes less power and also results in the memory and bandwidth savings.ECC offers stronger security with less server overhead and helps to reduce CPU cycles required for the server cryptographic operations.

4. APPLICATION OF THE ECC IN SECURE SOCKET LAYER PROTOCOL

ECC based SSL handshake is explained as follows. The client and server finalize an ECC based cipher suite. The server certificate contains the ECDH public key of the server which is signed by a certificate authority using the Elliptic Curve Digital Signature Algorithm (ECDSA).After authenticating the server, client send its ECDH public key to the server. Then the client and server use their own ECDH private keys and the other's public key to

derive the pre master key. A comparative study of the 1024-bit RSA based SSL handshake and 163-bit ECC based SSL handshake is done in [10].Handshake crypto latency and server crypto throughput are compared. In terms of the server crypto throughput ECC is five times better than RSA. In terms of handshake crypto latency, ECC runs twice as fast as RSA when both SSL client and SSL server are in the same platform but RSA beats ECC when the client and server runs in the different platforms. But when the comparison is made between 2048-bit RSA and 193-bit ECC, ECC outperforms RSA. It is concluded that as key size increases, servers perform better for ECC based SSL.

The performance improvement obtained by replacing RSA with ECC in the SSL protocol is studied in [11]. The experiments are conducted on Apache 2.0.45 web server compiled with ECC enhanced version of Open SSL. Open SSL contains open source implementation of SSL and TLS. The results show that the use of ECC allows the web server to handle more requests compared to RSA. Precisely Apache web server handles 13%-31% more https requests per second when ECC with160 bits is used instead of RSA with 1024 bits. Server performance is improved by 120%-279% when ECC 224 bits is used instead of RSA with 2048 bits.

5. SIDE CHANNEL ATTACKS ON ELLIPTIC CURVE ALGORITHMS IN SSL

Side channel attacks are used for cryptanalysis of elliptic curve cryptosystems. Side channel attacks are done by observing side channel information. These are the running time, power consumption and electromagnetic radiation[12].Attackers try to get the secret key by studying the working of the algorithms. Side channel attacks are easier when the end points of communication are mobile devices. The most common side channel attacks on ECC are Simple Power Analysis Attacks (SPA), Differential Power Analysis Attacks (DPA)and Timing Attacks (TA)[13].A survey of side channel attacks carried out on ECC and their counter measures are discussed in the work[14]. Both the active and passive side channel attacks are possible in ECC. Passive attack is a Simple Power Analysis attack, where the numerical value of the scalar k is found out if the attacker can differentiate point addition and point doubling from power traces. Active attacks are possible by incorporating some faults in the cryptographic scheme. One such scheme is weak curve based analysis, where the attacker will move the original strong curve to a weak one.

Timing attacks are done by measuring the time needed to perform the basic operations in the algorithm. Scalar multiplication being the basic operation in ECC, many side channel attack resistant algorithms are suggested in the literature for scalar multiplication. The commonly used method is Montgomery' ladder[15]. The authors of [16] describe a timing attack on Open SSL's ladder implementation of the elliptic curves over binary fields. They could get the private key of TLS server where the server uses ECDSA for authentication. In [17]authors has succeeded in recovering most of the bits of the scalar k, when the multiplication of the scalar k and base point G is computed by Montgomery ladder in the Open SSL .Scalar k is the pseudo randomly generated secret nonce which keeps the encryption scheme secure. The attack was done using FLUSH+ RELOAD technique to monitor the conditional statements used in the algorithm.

In [18], authors have conducted a study on the side channels attacks on the ECDSA running on the mobile devices. Attack was done on ECDSA implementation of Open SSL running on android and iOS devices by keeping a magnetic probe near the device. They found out the end points of each signing operations and by analyzing the signals key extraction is possible. Constant time implementation of double and add operations eliminate this attack.

In [19],authors have suggested a method in which doublings and additions in point multiplication occur in a fixed pattern which withstand side channel attacks. A new optimized elliptic curve implementation in Open SSL is described in [20].In this implementation, single point multiplication takes constant time. Hence there is no leakage of timing information about secret scalar and the method is immune to timing attacks.

6. CONCLUSION

Ensuring the security of the data is vital for the growth of web applications. Security protocols use encryption algorithms for message encryption, key exchange and authentication. To secure online transactions, SSL is the commonly used protocol. But SSL processing is a time consuming factor and secure web servers are slower than non- secure ones. The high processing time is due to the asymmetric key encryption algorithms used in SSL. RSA is the asymmetric key encryption algorithm which is commonly used in SSL .In the recent years the key size of RSA is increased to withstand the attacks which causes additional burden on the web servers. ECC is a good alternative to RSA because of the smaller key sizes. The smaller key sizes of ECC offers memory and bandwidth savings. The incorporation of elliptic curve cryptography in SSL improves the performance of web servers but side channel attacks are possible in the elliptic curve implementations in SSL. Most of the side channel attacks exploit the weaknesses in the implementations of the scalar point multiplication, which is the fundamental operation in elliptic curve arithmetic. Side channel attacks are defeated by constant time implementation of scalar point multiplication.

REFERENCES

[1] R.L. Rivest, A. Shamir and Adleman, L, "A method for obtaining digital signatures and public-key cryptosystems", Communications of the ACM, Vol. 21, No. 2, pp. 120-126, 1978.

[2] V.S. Miller, "Use of elliptic curves in cryptography", In Conference on the Theory and Application of Cryptographic Techniques. Springer Berlin Heidelberg, pp. 417-426, 1985.

[3] N. Koblitz, "Elliptic curve cryptosystems", Mathematics of computation, Vol. 48, No. 177, pp. 203-209, 1987.

[4] W. Stallings, "Cryptography and network security: principles and practices", Pearson Education India, 2006.

[5] A. Kahate, "Cryptography and network security", Tata McGraw-Hill Education, 2013.

[6] K. Kant, R. Iyer and P. Mohapatra, "Architectural impact of secure socket layer on internet servers", In International Conference on Computer Design, pp. 7-14, 2000.

[7] L. Zhao, R. Iyer, S. Makineni and L. Bhuyan, "Anatomy and performance of SSL processing", In IEEE International Symposium on Performance Analysis of Systems and Software, pp. 197-206, 2005.

[8] M.A. Alnatheer, "Secure Socket Layer (SSL) Impact on Web Server Performance", Journal of Advances in Computer Networks, Vol. 2, No. 3, 2014.

[9] http://csrc.nist.gov/publications/nistpubs/800-131A/sp800-131A.pdf

[10] Gupta, V., Gupta, S., Chang, S., & Stebila, D. (2002, September). Performance analysis of elliptic curve cryptography for SSL. In Proceedings of the 1st ACM workshop on Wireless security (pp. 87-94). ACM.

[11] V. Gupta, D. Stebila, S. Fung, S.C. Shantz, N. Gura and H. Eberle, "Speeding up Secure Web Transactions Using Elliptic Curve Cryptography", In NDSS, 2004.

[12] Marc Joye, "Elliptic Curves and Side Channel Analysis", ST Journal of System Research, Vol. 4, No. 1, pp. 283-306, 2003.

[13] E.K. Reddy, "Overview of the side channel attacks", International Journal of Advanced Networking and Applications, Vol. 4, No. 6, pp. 1799, 2013.

[14] J. Fan, X. Guo, E. De Mulder, P. Schaumont, B. Preneel and I. Verbauwhede, "State-of-the-art of secure ECC implementations: a survey on known side-channel attacks and countermeasures", In IEEE International Symposium on Hardware-Oriented Security and Trust (HOST), pp. 76-87, 2010.

[15] P.L. Montgomery, "Speeding the Pollard and elliptic curve methods of factorization", Mathematics of computation, Vol. 48, No. 177, pp. 243-264, 1987.

[16] B.B. Brumley and N. Tuveri, "Remote timing attacks are still practical", In European Symposium on Research in Computer Security. Springer Berlin Heidelberg, pp. 355-371, 2011.

[17] Y. Yarom and N. Benger, "Recovering Open SSL ECDSA Nonces Using the FLUSH+ RELOAD Cache Side-channel Attack", IACR Cryptology e Print Archive, Vol. 140, 2014.

[18] D. Genkin, L. Pachmanov, I. Pipman, E. Tromer and Y. Yarom, "ECDSA key extraction from mobile devices via nonintrusive physical side channels", In Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, pp. 1626-1638, 2016.

[19] B. Möller, "Securing elliptic curve point multiplication against side-channel attacks", In International Conference on Information Security. Springer Berlin Heidelberg, pp. 324-334, 2001.

[20] E. Käsper, "Fast elliptic curve cryptography in Open SSL", In International Conference on Financial Cryptography and Data Security. Springer Berlin Heidelberg, pp. 27-39, 2011.