Enhancing Email Security by Signcryption based on Elliptic Curve

A K Mohapatra,PhD
Department of Information Technology
IGDTUW, Kashmere Gate
Delhi, India

Jyoti Kushwaha
Department of Computer Science and Engineering
IGDTUW, Kashmere Gate
Delhi, India

Tanya Popli
Department of Computer Science and Engineering
IGDTUW, Kashmere Gate
Delhi, India

ABSTRACT

E-Mail plays an important role in mankind communication. It is essential to provide security solutions for the secure transmission of email. The currently used protocols such as PGP and S/MIME are based on public key cryptography that involves huge computational costs due to key generation, key exchange and encryption. In this paper, an elliptic curve based signcryption scheme is introduced that provides several security attributes such as confidentiality, authentication, integrity, non-repudiation, and forward secrecy for electronic mails.

Keywords

Authentication, Digital Signature, Email security, Encryption, Signcryption.

1. INTRODUCTION

Electronic mail plays a vital role in our communications. With the Internet availability, it is nowadays considered as a formal communication in many institutions and organizations. Security of the email has become important as reliance on the electronic mail increases.

Two major end-to-end email security solutions available are: *Secure/Multipurpose Internet Mail Extension(S/MIME)* and *Pretty Good Privacy(PGP).* Both of them are based on public key cryptography [8] that involves huge computational costs.

In this paper, a new protocol for secure email is introduced that provides integrity, confidentiality, authentication, non-repudiation and forward secrecy. It is based on signcryption using elliptic curves.

2. EVOLUTION

Public key cryptography was invented nearly two decades ago. It has made it possible for people to communicate with one another in a secure and authenticated way over an insecure network such as Internet. In doing so a two-step approach has been followed. That is, before the message is sent, the sender signs the message using digital signature scheme, and then encrypts the message and the signature using some private key encryption algorithm under a randomly chosen secret key. This key is then encrypted using recipient's public key. We call this two-step approach signature-then-encryption [1]. Digital Signature ensures authenticity while encryption ensures confidentiality of the message.

But digitally signing a message and then encrypting it consumes more machine cycles. It also bloats the message by introducing extended bits to it. Symmetrically, larger the size of the message greater will be the time required to encrypt and verify it[1].To overcome this issue,Zheng[1] developed *Signcryption*.Signcryption is a cryptographic primitive that combines both the functions of digital signature and public key encryption logically in a single step, and with a computational cost significantly less than that needed by the traditional signature-then-encryption approach [1].The various phases of signcryption can be broadly categorized as: Initialisation phase, Signcryption phase, Unsigncryption phase and Verification phase.

In first phase, various parameters such as, for key generation etc., are set. Next phase carries out the signcryption of the message to be sent by the sender. After this phase the cipher text (encrypted message) along with sender's signature are communicated to the receiver. In third phase, the receiver unsigncrypts the cipher text to retrieve the original message. Also, the receiver verifies the authenticity of the communication by verifying the sender's signature.

Zheng [2]proposed another signcryption scheme based on elliptic curve, which saves about 58% computational cost and saving about 40% communication cost than signature-then-encryption scheme based on elliptic curve.Elliptic curve based signcryption schemes are based on the elliptic curves defined over finite field. Elliptic curves having the equation of the following form are used:

$$y^2 = x^2 + ax + b$$

This definition of elliptic curve also includes an element denoted$O$, which is called the point at infinity. For a given prime, p, the finite field of order p, $GF(p)$ is defined as the set Zp of integers $\{1,2,..p-1\}$, together with the arithmetic operations modulo p. Let G be a point on the elliptic curve over $GF(p)$. The order of the point Gcan be defined as the smallest integer n such that $nG = O$. For an integer n, the multiple of G, nG (point multiplication over elliptic curve over $GF(p)$), can be computed by repeated point additions on elliptic curve over $GF(p)$. The inverse problem of computing the multiple of a point, e, such that in the equation $P = eG, P$ and G are known, is known as elliptic curve discrete logarithm problem. When the order of G is a large prime number greater than say 2^{160}, then elliptic curve discrete logarithm problem is a hard problem and cannot be solved in polynomial time [3]. The entire elliptic curve based cryptosystems base their security on the hardness of the elliptic curve discrete logarithm problem [7].

3. EMAIL SECURITY

PGP was created by Phil Zimmerman in 1991. PGP provides confidentiality through encryption of the message and authentication through digital signature service that can be used for the security of electronic mail.

S/MIME is a security enhancement to the MIME Internet e-mail format standard, based on technology from RSA Data Security.

Both PGP and S/MIME use "signature-then-encryption" schemes for providing the confidentiality and authenticity of the message.

These protocols are based on the public key cryptography that involves huge computational costs making them unsuitable for resource constrained devices. These protocols use RSA/DSA for calculating the digital signature which involves modular exponential calculations. The EC based cryptosystems have computational advantage over the exponential systems and, hence, are more suitable for resource-constrained devices. The EC-based systems canattain to a desired security level with significantly smallerkeys than those of required by their exponential-basedcounterparts. As an example, it is believed that a 160-bit keyin an EC-based system provides the same level of security asthat of a 1024-bit key in a RSA-based system [4]. This createsgreat efficiencies in key storage, certificate size, memoryusage, and required processing so it enhances the speed andleads to efficient use of power, bandwidth, and storage that arebasic limitations of resource constrained devices.

4. PROPOSED APPROACH FOR EMAIL SECURITY

PGP and S/MIME based on "signature-then-encryption" scheme, suffers from two major problems: low efficiency and high computational cost. Signcryption is a recently proposed scheme that combines digital signature and encryption into logically one step that decreases the computational cost and communication overhead [1].

Attack on full 80-step SHA-1 is now known with complexity less than 2^{80} theoretical bound. This attack shows that collisions on SHA-1 can be found with the complexity less than 2^{69} hash operations. [6]

In 1998, the Electronic Frontier Foundation built a DES cracker that could decode DES messages in less than a week. The length of AES key can be 128, 192 or 256 bits. There is currently no evidence that AES has any weaknesses making it attackable by only brute force attack [11].

Proposed approach is as:

EC based signcryption scheme is used since it is based on the difficulty of solving the ECDLP and the factorization in elliptic curves. It has been proved by Pollard Rho that if the order, n, of G, the base point on the elliptic curve, is less than 160 than the ECDLP could be solved in polynomial time. So the value of n chosen should be a prime number greater than 160 [3].

The symmetric key for encrypting and decrypting the message is calculated on the basis of the public key of the recipient [2].

SHA-2 would be used as the hashing function to produce a hash of the message concatenated with the ID of the recipient. SHA-2 consists of a set of four hash functions with digests that are 224, 256, 384 or 512 bits. Although SHA-2 bears some similarity to the SHA-1 algorithm, the attacks on SHA-1 have not been successfully extended to SHA-2.The solution will use the version that will produce digests of 256 bits.

The symmetric key algorithm that will be used to encrypt and decrypt the message would be AES. The key length for AES algorithm can be 128, 192 or 256 bits. We would use a symmetric key of length 128 bits because no attack on AES algorithm is known yet except brute force attack and an exhaustive search for key for AES-128 would take around 10 000 000 000 000 000 000 000 000 years even if the keys are tried at the rate of 1 million keys per second.[11].

5. DETALIED DISCUSSION

The algorithm as proposed by Hwang et.al [5] consists of four phases:

- Initialization
- Signcryption
- Unsigncryption
- Judge Verification

5.1 Initialization Phase:

This phase consists of two major steps:

1) Selection of domain parameters.

The parameters are:

p -- a large prime number such that $p>2^{160}$

a, b -- two integer elements that lie in $[1,2,..p-1]$ and satisfy the equation: $4a^2 + 27b^2 \ modp \ \neq 0$

F -- the elliptic curve over finite field which is selected such that
$y^2 = x^2 + ax + bmodp$ where x and y both lie in $[1,2,..p-1]$.

O -- a point (x, y) of F at infinity.

G -- a base point on F.

n -- the order of point G such that $nG = O$ and $n>2^{160}$.

H -- one way hash function. The hash function used in our approach is SHA-256.

$E_k()/D_k()$-- symmetric key encryption and decryption algorithm. The algorithm used in our approach is AES.

2) Generation of public and private keys of users and issuing a public key for the certificate of each user. The private key of a user U is a randomly generated $d_U \in [1, p-1]$ and the corresponding public key is generated as $U_U = d_U G$.

The CA server issues a certificate $Cert_U$for the public key of each user. The certificates contain strings of information that uniquely identify users and bind their identities to their public keys. Each user has its e-mail id as its unique ID_U because e-mail ids are unique for each user.

5.2 Signcryption Phase:

In this phase, the sender requests for the public key of the receiver by sending $ID_B$to the server. The server then sends the digitally signed (ID_B,U_B) to the sender. The sender then verifies the server's response using server's public key and extracts ID_B and U_B and uses them to signcrypt the message.

The signcrypted message (C, R, s) produced in the following steps is then sent to the receiver.

1) Randomly select an integer v less than n.

2) Compute $K = vU_B = (k, l)$

3) Compute $R = vG = (r1, r2)$

4) Encrypt the message M using symmetric key k to produce C such that $C = E_k(M)$.

5) Compute $t = H((M||ID_B), r1)$.

6) Compute $s = td_A - v \ mod n$.

7) Send (C, R, s) to the receiver.

5.3 Unsigncryption Phase:

In this phase, the receiver requests for the public key of the sender by sending $ID_A$to the server. The server then sends the digitally signed (ID_A,U_A) to the receiver. The receiver then verifies the server's response using server's public key and extracts $ID_A$and U_A and uses them to unsigncrypt the message. The original message is extracted and verified by the receiver using the following steps:

1) Compute $K = d_B R = (k, l)$

2) Obtain the plaintext M by applying the symmetric key k to the decryption algorithm $D_k(C)$.

3) Compute $t = H((M||ID_A), r1)$.

4) Accept message M as correct message only if $sG + R = tU_A$, else reject it.

5.4 Judge Verification Phase:

This phase is required if we need to verify that the sender actually sent the message. In this phase, a trusted third party decides if the message was sent by the sender. In case of a dispute, the receiver sends (M, R, s) to the judge. The judge makes the decision on the basis of following steps:

1) Compute $t = H((M||ID_{B)}), r1)$.

2) If $sG + R = tU_A$, then the sender actually sent (C, R, s) to the receiver, else the sender did not send it.

6. COMPARISON

PGP and SMIME are based on "signature-then-encryption scheme" whereas the proposed approach is based on elliptic curve based signcryption scheme. These algorithms were implemented on 2.3 GHz 64 bit processor. The average time for generating keys for approach based on elliptic curve signcryption came out to be 260ms and that for the signature-then-encryption scheme came out to be 650ms. The average total time for signcrypting 1KB message came out to be 375ms and that for signature-then-encryption message came out to be 800ms.

7. CONCLUSION

An enhancement of the e-mail protocol using signcryption based on Elliptic curve which provides confidentiality, authenticity, integrity, unforgeability, non-repudiation, forward secrecy and public verifiability is introduced. The proposed approach overcomes the shortcomings of SHA-1 algorithm by using SHA-2 algorithm which has not been broken yet. This approach uses AES-128 symmetric key algorithm as no other attack is known against it, except brute-force attack, thus, ensuring confidentiality. This approach uses Elliptic curve based signcryption which saves computational cost as compared to its exponential counterparts to suit the resource constrained devices like mobile devices.

A new emerging field HECC (hyper elliptic curve cryptography) has the potential to provide better efficiency than ECC with same level of security. Thus, HECC can be used as further improvement of ECC.

8. REFERENCES

[1] Y. Zheng, "Digital signcryption or how to achieve cost (signature & encryption) << cost (signature) + cost (encryption)," in Seventeenth Annual International Cryptology Conference, 1997, pp. 165-179.

[2] Y. Zheng and H. Imai, "How to construct efficient signcryption schemes on elliptic curves", Information Processing Letters, vol. 68, pp. 227-233, 1998.

[3] N. Koblitz, A. Menezes and S. Vanstone, "The state of elliptic curve cryptography", Designs, Codes And Cryptography, vol. 19, no. 2, pp. 173-193,2000.

[4] L. Batina, S.B. Örs, B. Preneel, and J. Vandewalle, "Hardware architectures for public key cryptography," Integration, the VLSI Journal, vol. 34, no. 1, pp. 1-64, 2003.

[5] R.J Hwang, C.H Lai and F.F Su, "An efficient signcryption scheme with forward secrecy based on elliptic curve", Applied Mathematics and Computation, vol. 167, no. 2, pp. 870-881, 2005.

[6] X. Wang, Y.L. Yin, and H. Yu, "Finding collisions in the full SHA-1", inTwenty Fifth Annual International Cryptology Conference, 2005, pp. 17-36.

[7] L. Xiang-xue, C. Ke-fei and L. Shiqun, "Cryptanalysis and improvement of signcryption schemes on elliptic curves", Wuhan University Journal of Natural Sciences, vol. 10, no. 1, pp. 231-234, 2005.

[8] W. Stallings, Cryptography and Network Security, 4th ed. New Delhi, India: DK Publishing Inc., 2006.

[9] S. Kim, C. Lee, D. Kim and H. Oh, "A Practical Way to Provide Perfect Forward Secrecy for Secure E-Mail Protocol", in 4th International Conference, ICDCIT 2007, pp. 327-335.

[10] M. Toorani and A.A.B. Shirazi, "An elliptic curve-based signcryption scheme with forward secrecy", Journal of Applied Sciences, vol. 9, no. 6, pp. 1025-1035, 2009.

[11] H.O. Alanazi, B.B. Zaidan, A.A. Zaidan, H.A. Jalab, M. Shabbir and Y. Al-Nabhani, "New Comparative Study Between DES, 3DES and AES within Nine Factors", Journal of Computing, vol. 2, pp. 152-156, March 2010.