International Journal of Engineering Science Invention
ISSN (Online): 2319 – 6734, ISSN (Print): 2319 – 6726
www.ijesi.org Volume 2 Issue 8 ǁ August 2013 ǁ PP.10-16

E-Governance Security using Public Key Cryptography With special focus on ECC

Prokash Barman[1], Dr Banani Saha[2]
[1]*(Department of Computer Science & Engineering, University of Calcutta, India)*
[2]*(Department of Computer Science & Engineering, University of Calcutta, India)*

ABSTRACT : *This paper gives an idea on E-Governance Security using public key cryptography. The public key cryptography technique called Elliptic Curve Cryptography (ECC) become popular in the last few years. The main attraction of ECC is that it takes exponential time challenge for an intruder to break into the cryptographic system. ECC using 160 bits key, provides the same security as offered by the well known RSA using 1024 bits key, thus leading to lower processing time, minimal bandwidth use and speedy transactions.*

KEYWORDS*: E-Governance, Public Key Cryptography, RSA, Elliptic Curve Cryptography (ECC), Encryption.*

I. INTRODUCTION

Recent days with the advancement of Information and Communication Technology (ICT) various E-Governance systems such as Government – to – Government (G-2-G), Government – to – Business (G-2-B), Government – to – Citizen (G-2-C), Government – to – Employee (G-2-E) etc have evolved nowadays. The hackers and intruders may attack the E-Governance systems directly or indirectly. The unauthenticated access of information and alteration of valuable Government data may be the target segment of the attackers. Hence the E-Governance systems need to be highly secured. With the use of Public Key Cryptography algorithm, risks may be minimized during various E-Governance transactions.

We discuss Public Key Cryptography algorithms and its utilization in E-Governance security in different sections of this paper, which are arranged as follows- Section 2 defines some terms related to the E-Governance security. RSA and ECC Public Key Cryptography algorithms have been discussed in section 3. The advantages of ECC over RSA are depicted in section 4. The section 5 describes various ECC implementations for E-Governance system. Conclusion is depicted in section 6.

II. E-GOVERNANCE

The best utilization of the information and communication technology (ICT) to improve efficient service delivery to the citizen is called e-governance. E-governance is the process of delivering information to the user or client in an efficient way for the benefit of both client & government.

1.1 E-Governance Risk Factors

The E-governance risk factors [1] seen are as follows

2.1.1 Spoofing:

In this technique the attacker attempts to gain access of E-Governance system by using fallacious identity either by stealth or by using false IP address. Once the access is gained, the attacker abuses the E-Governance system by elevation of privileges.

2.1.2 Repudiation:

The attacker can mount repudiation attack during the E-Governance transaction, which is the ability of the user to deny its performed transaction.

2.1.3 Disclosure of E-Governance Information:

Unwanted information disclosure can take place easily, in case of the compromised E-Governance system.

2.1.4 Denial of Service: Attackers can perform Denial of Service (DoS) attack by flooding the E-Governance server with request to consume all of its resources so as crash down the system.

2.1.5 Elevation of Privilege

2.1.6 Cyber Crimes

1.2 Security measures to reduce E-Governance risk factors

The information transmitted in the internet is broken into data packets which may travel over different routes to reach the destination. The most vulnerable point of interception of information is the points of ***entry and exit*** from the internet. [1]

Encrypting and decrypting the packets moving between ***these two fragile points*** may be done to keep the data secured and intact. There are different types of Cryptography algorithm to achieve the above target are described below:

1.2.1 Symmetric or Secret Key Encryption:

Encryption algorithms that use the same key for encrypting and for decrypting information are called symmetric-key algorithms. It is also called a secret key because it is kept as a shared secret between the sender and receiver of information. Else, the confidentiality of the encrypted information is compromised. Strength of Symmetric Key encryption depends of the size of the key used. Symmetric key encryption is much faster than public key encryption, (may be 100 to 1,000 times faster).

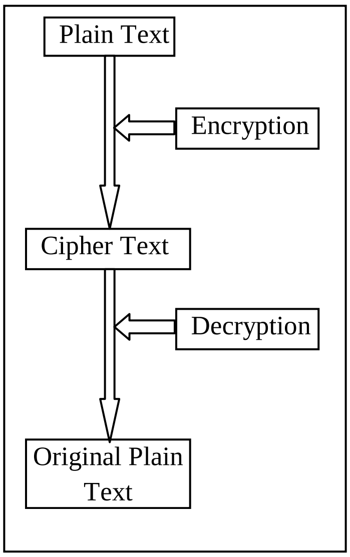


Figure 1: Encryption and Decryption

Encryption: $E_K(M)=C$
Decryption: $D_K(C)=M$

Where,

M=>Original Message
E=> Encryption Function
D=>Decryption Function
C=>Cipher Text
K=>Secret Key

Symmetric/ Secret Key Encryption

1.2.2 Asymmetric or Public Key Encryption:

Encryption algorithm that use different keys for encrypting and decrypting information are most often called public key algorithm[3] but are sometimes also called asymmetric key algorithms[3]. Public key encryption requires the use of both a private key (a key that is known to only its owner) and a public key (a key that is available to and known to other entities on the network). All the user's public key may be published in the directory so that it is accessible to all the users of the organization. The two keys are different but one is complement to other. Information that is encrypted with the help of public key can be decrypted only with the corresponding private key of the set.

Encryption: $EK^1(M)=C$
Decryption: $DK^2(C)=M$; $DK^2(EK^1(M))$
Where,

M=>Original Message
E=> Encryption Function
D=>Decryption Function
C=>Cipher Text
K^1=>Private Key
K^2=>Public Key
$K^1 \neq K^2$

Asymmetric/ Public Key Encryption

1.2.3 Secret Key Exchange[5]

In case of online communications using symmetric key cryptography, the secret key must be shared with communicating parties and protected from unauthorized parties. To exchange secret key online, public key encryption algorithm may be implied to keep the key exchange secure.

III. PUBLIC KEY CRYPTOGRAPHY

There are two prominent Public Key Cryptography algorithms mostly used for secured transaction i.e. RSA and ECC. The ECC is the most recent technique with low key size, minimum processing overload and provide same security compared to large key based RSA algorithm.

1.3 RSA

RSA [2] is an algorithm for public-key cryptography that is based on the presumed difficulty of factoring large integers. RSA is the first letters of the surname of ***Ron Rivest, Adi Shamir and Leonard Adleman***, had first publicly described it in 1977. In 1973, an equivalent system had developed by an English mathematician, Clifford Cocks, but it was classified until 1997. The participating user of RSA algorithm creates and then publishes the product of two large prime numbers, along with their public key, as an auxiliary value. The prime factors must be kept secret. A message can be encrypt by anyone using the public key, but with currently published methods, if the public key is large, someone with knowledge of the prime factors can feasibly decrypt the chipper text. Whether breaking RSA encryption is as hard as factoring is an open question known as the RSA problem.

As an illustration of this take two large prime numbers (300) digits, multiply them together. As a result you will find

a) a large number (More or equals 300 digits)
b) it has two factors, both are prime (Multiplier)

The two prime numbers can be given easily from which the product can be calculated. But finding the primes from the given product is more difficult. In fact if the numbers get adequately large, it is almost impossible to find them. So the multiplying of two large prime numbers together is the easy forward function in this algorithm. The inverse factor finding operation is considerably more difficult and in practice its almost impossible. The RSA system employs this fact to generate public key and private key pairs. The keys are the functions of the product and the primes. The arrangement of this cryptosystem is to operate in easy forward function – multiplication. Conversely, the operation need to make difficult to find the plaintext from cipher text using only public key, because the inverse operation needs to solve difficult inverse factoring problem.

1.4 Elliptic Curve Cryptography (ECC)

As the key length of secure RSA has increased recent years, it has put a heavier processing load on application using RSA. This problem has been reduced using smaller key length Elliptic Curve Cryptography (ECC). The ECC is based on Elliptic Curve.

Elliptic Curve:

Elliptic Curves are not actually ellipses. The curve are so named because they are depicted by cubic equation, similar to those used for calculating the circumference of an ellipse. The locus of a point, whose coordinates conform to a particular cubic equation along with the point at infinity O (the point at which the locus in the projective plane intersects the line at infinity) is known as an **elliptic curve [7]**.
An elliptic curve is defined in a two dimensional, standard, x, y Cartesian coordinate system are given below as equation (1) and (2)

$$\mathbf{y^2 = x^3 + ax + b \ (1) \quad where\ 4a^3 + 27b^2 \neq 0.}$$
$$\mathbf{y^2 + xy = x^3 + ax^2 + b \ (2)}$$

The mathematical foundation of ECC is based on the above equations which may be depicted as follows

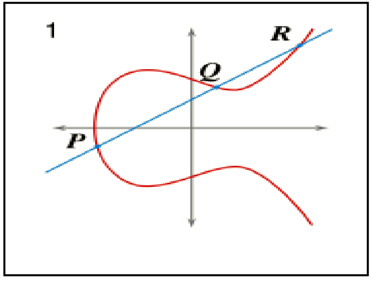


Figure 2: An Elliptic Curve

In ECC, the elliptic curve is used to define the members of the set over which the group is calculated and the operations between them which define how math works in the group. It is done by imagine a graph labelled along both axes with the numbers of a large prime field.

Elliptic curve cryptography (ECC) [4] is an approach to public-key cryptography based on the algebraic structure of elliptic curves over finite fields. In 1985, Neal Koblitz and Victor S. Miller suggested independently, the use of elliptic curves in cryptography.

The elliptic curve cryptosystem is one of the three cryptosystems currently in use for public key cryptography (PKC), integer factorization systems and discrete logarithm systems are the other two systems. The RSA cryptosystem is the best known example of the integer factorization problem while the Digital Signature Algorithm (DSA) cryptosystem is based on the discrete logarithm problem.

Public-key cryptography is based on the interoperability of certain mathematical problems. Before ECC, it assumed that Public-key systems are secure assuming that it is difficult to factor a large integer composed of two or more large prime factors. In case of elliptic curve based protocols, finding the discrete logarithm of a random elliptic curve element with respect to a publicly known base point is not feasible which is assumed. The size of the elliptic curve determines the difficulty of the problem. Benefit of ECC is a smaller key size, which facilitate to reduce storage and transmission requirements, as a result an elliptic curve group could provide the same level of security afforded by an RSA-based system with a large modulus and correspondingly larger key, i.e. a 256bit ECC public key should provide comparable security to a 3072bit RSA public key .

The mathematical operations of ECC is defined over the elliptic curve $y^2 = x^3 + ax + b$, where $4a^3+27b^2\neq 0$. Each value of ‘a’ and ‘b’ gives a different elliptic curve. All points (x , y) which satisfies the above equation plus a point at infinity lies on the elliptic curve. A point in the curve is Public key and the private key is a random number. By multiplying the private key with the generator point G in the elliptic curve, the public key is obtained. The domain parameter (Set of pre defined constant known by all the parties taking part in the communication) of ECC constitute by the generator point G, the curve parameters ‘a’ and ‘b’, along with few other constants.

IV. ECC OPERATIONS

Various operations which are performed on ECC are given below in detail [7]:

1) Point Multiplication

The dominant operation in ECC cryptographic schemes is point multiplication method. Point multiplication is calculating the value of *nJ*, where *n* is an integer and *J* is a point on the elliptic curve defined in the prime field. In point multiplication, a point J on the elliptic curve is multiplied with a scalar n using elliptic curve equation to obtain another point K on the same elliptic curve, i.e. nJ=K.

The Point multiplication method is achieved by two basic elliptic curve operations.

- Point addition, adding two points J and K to obtain another point L i.e., L = J + K.
- Point doubling, adding a point J to itself to obtain another point L i.e. L = 2J.

a) Point addition: The Point addition method is the addition of two points J and K on an elliptic curve to obtain another point L on the same elliptic curve.

Consider two points J and K on an elliptic curve as shown in figure (a). If K ≠-J then a line drawn through the points J and K will intersect the elliptic curve at exactly one more point –L. The result of addition of points J and K gives the point -L. The reflection of the point –L with respect to x-axis gives the point L. That is on an elliptic curve L = J + K. If K = -J the line through this point intersect at a point at infinity O. Hence J + (-J)= O. This is shown in figure 2(b). Where, O is the additive identity of the elliptic curve group. The reflection of a point with respect to x-axis is the negative of that point.

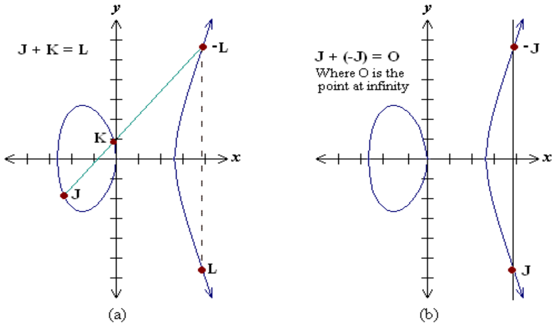


Figure 3: Addition of two points

b) Point doubling: Point doubling is the addition of a point J on the elliptic curve to itself to obtain another point L on the same elliptic curve. To find L = 2J which imply to double a point J to get the point L, consider a point J on an elliptic curve as shown in figure 3(a). If y axis coordinate of the point J is non zero then the tangent line at J will intersect the elliptic curve at exactly one more point –L. The reflection of the point –L with respect to x-axis gives the point L, which is the doubling result of the point J.

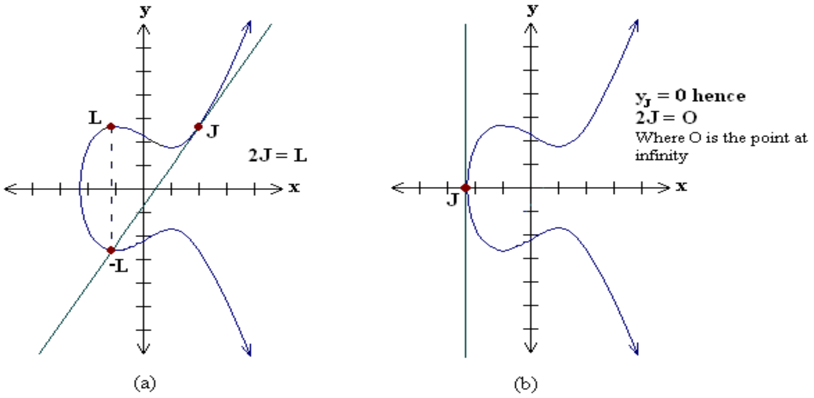


Figure 4: Doubling of Points

Thus L = 2J. If y coordinate of the point J is zero then the tangent at this point intersects at a point at infinity O. Hence 2J = O when yJ = 0. This is shown in figure.

Security policy of ECC (Discrete Logarithm Problem)

The security of ECC [8] depends on how difficult is to determine *n* given nJ and J. This difficulty is called Elliptic Curve Discrete Logarithm Problem. Let J and K are two points on an Elliptic Curve such that *n*J=K, where *n* is a scalar. Given J and K, it is computationally infeasible to obtain *n*, if *n* is sufficiently large. *n* is the discrete logarithm of K base J. Hence the main operation involved in ECC is point multiplication i.e. multiplication of scalar *n* with the point J on the curve to obtain point K on the curve.

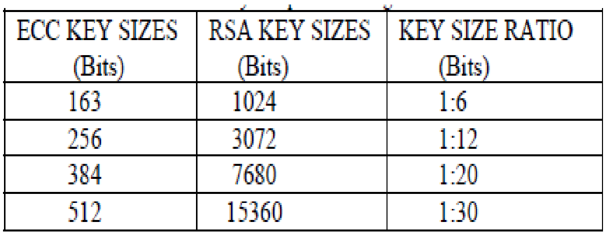
V. ADVANTAGES OF ECC OVER RSA

There are lot of advantages of ECC over RSA. These advantages become more and more distinct as security levels increase (and, as a result, as hardware gets faster, and the key sizes as per recommendation must be increased). A 384-bit ECC key is equivalent with a 7680-bit RSA key for security. The smaller ECC keys mean the cryptographic operations that must be performed by the communicating devices can be squeezed into considerably smaller hardware as a result software applications may complete cryptographic operations with fewer processor cycles, The operations can be performed much faster, while guaranteeing equivalent security. Which stand for less heat, less power consumption, less resourse consumed on the printed circuit board hardware, and software applications run more rapidly and make lower memory demands which lead in turn to more portable devices to produce less heat while run longer.

Now, consider the following three factors

1) Firstly, the factor that the security and practicality of a given asymmetric cryptosystems relies upon the difference in difficulty between doing a given operation and its inverse.
2) Second, the factor that the difference in difficulty between the forward and the inverse operation in a given system is a function of the key length in use, due to the fact that the difficulty of the forward and the inverse operations increase as very different functions of the key length; the inverse operations get harder faster.
3) (iii) Third, the factor that as the longer key lengths are used to adjust the greater processing power are now available to attack the cryptosystem, even the 'legitimate' forward operations get harder, and require greater resources (chip space and/or processor time), though by a lesser degree than do the inverse operations.

If these three factors can be recognized, the advantages of ECC over other asymmetric cryptosystems can be easily grasp.



ECC KEY SIZES (Bits)	RSA KEY SIZES (Bits)	KEY SIZE RATIO (Bits)
163	1024	1:6
256	3072	1:12
384	7680	1:20
512	15360	1:30

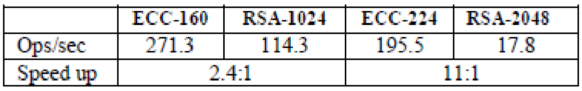
Table 1: Comparison between RSA and ECC Key size

VI. ECC FOR E-GOVERNANCE

To reduce the risk factors of E-Governance, suitable Public Key Cryptography method must be implemented. Among the latest Public Key Cryptography algorithms ECC is best suited for E-Governance systems for its speed, security, lowest resource usages and low processing load. The ECC based crypto system is recommended to use in the following area [6].

Security of E-Governance Web Sites:

The most dominant protocols for providing security in the Internet are the **Secure Socket Layer** (SSL) and its sister protocol namely **Transport Layer Security** (TLS) protocols. Although, the use of these protocols puts a significant performance overhead on the web servers because the protocols uses RSA algorithm. It is seen that the use of ECC-224 over RSA-2048 improves server performance by 120%–279%. An experiment in [9] shows that replacing RSA with ECC reduces the servers processing time for new SSL connections across the entire range of page sizes from 10KB to 70KB. For a 70KB page (comparing ECC-160 with RSA-1024), the measured reduction ranges from 29% to 85%, for a 10KB page (comparing ECC-224 with RSA-2048).



	ECC-160	RSA-1024	ECC-224	RSA-2048
Ops/sec	271.3	114.3	195.5	17.8
Speed up	2.4:1		11:1	

Table 2: Performance of Public Key Algorithm

The **Secure Electronic Transaction** (SET)

specification enables highly secure Internet shopping using credit cards. This kind of cards are developed by Visa and MasterCard company in response to the security concerns of transacting on the Internet. Byung Kwan Lee proposed Advanced Secure Electronic Payment (ASEP) Protocol that uses ECC to secure the online transactions.

Strangio and Me proposed the **EC-PAY** e-Cheque Payment Scheme, which makes use of ECC primitives for local payment transactions, to be deployed in the realm of a PKI infrastructure in a wireless environment or on a mobile device.

Personal Computers used for E-Governance:

Although devices having fewer resources are considered suitable for ECC, government software may be developed with ECC based security on desktop PCs/Laptops, primarily for protection of data and for mail security.

Identification devices like as smart cards and RFIDs:

RFID (Radio Frequency Identification) tags are a new generation of small devices used for identification in many applications such as e-tolling in motorways, payment through mobile phones, telemetry, product tracking, identification of patients and hospital staff and other employee identification. To prevent counterfeiting problems, Radio Frequency Identification technique has gained appreciation as an emerging technology. ECC based RFID Authentication Protocol (ERAP) which helps to secure, mutual offline authentication is proposed by Ahamed et al.

ECC is best suited for smart cards as they have extremely rigid constraints on processing capability, storage capacity of parameters and code space. Smart cards are used primarily for signing and decryption operation where ECC is best suited, because it is fast and requires lesser computing power. There are many manufacturing companies who are producing smart cards that using elliptic curve digital signature algorithm. The smart cards are flexible devices which can be used in many situations such as banks credit/debit cards, personal identification or registration cards and electronic tickets.

Woodbury et al. demonstrate the use of ECC on smart cards without coprocessors. They demonstrate that scalar multiplication of a fixed point of an EC (the core operation for signature generation) can be performed in less than 2 seconds on an 8051 microcontroller. Chatterji and Gupta proposeed an authentication protocol based on ECDSA for smart cards in.

Biometric Signature Verification in E-Governance:

An approach using biometric signatures, based on the ECC may be implied in E-Governance. The use of ECC in biometric signature creation improves the electronic banking security, in this technique the public and private keys are created without transmitting and storing any private information nowhere. The two parties need to share a secret key through an insecure channel in symmetric cryptography. ECDH (Elliptic Curve Diffie–Hellman) is a protocol to create and share secret keys between parties without transmitting any private value, as a result no one has access to these secret keys except the communicating parties.

Combining the biometrics and the ECDH algorithm, secret messages can be generated in symmetric cryptography with the help of dynamically generated private keys.

VII. CONCLUSION

Elliptic curve cryptography is implemented in the NSA's Suite B to protect both classified and unclassified national security systems and information. It used for both digital signatures and key exchange. This shows that elliptic curve cryptosystem is ready for real world use and is to be preferred in many cases over other cryptosystems.

Although there is no real mathematical proof that elliptic curve cryptosystem is more secure than cryptosystems based on discrete logs over finite fields or integer factorisation, elliptic curve cryptosystem seems to be the most efficient and secure public key cryptosystem available today.
With the possibility of evaluating nature of attacks on E-Governance system, the strongest and advanced Cryptographic mechanism i.e. ECC must be engineered to obtain highest level of security of E-Governance system.

REFERENCES

[1]. Abhishek Roy and Sunil Karforma, "Risk and Remedies of E-Governance System", in Oriental Journal of Computer Science &Technology, Vol. V, No.(2), pp. 329-339, December 2011

[2]. Wikipedia.org, http://en.wikipedia.org/wiki/RSA_%28algorithm%29, -Accessed on 17-10-2012

[3]. Bruce Schneier., "Applied Cryptography: Protocols, Algorithms, and Source Code in C", -Second Edition.

[4]. Wikipedia.org, http://en.wikipedia.org/wiki/Elliptic_curve_cryptography -accessed on 17-10-2012

[5]. Microsoft Technet, http://technet.microsoft.com/en-us/library/cc962035.aspx -, accessed on 17-10-2012

[6]. Vivek Katiyar, Kamlesh Dutta & Syona Gupta,"A Survey on Elliptic Curve Cryptography for Pervasive Computing Environment", in International Journal of Computer Applications (0975 – 8887) , Volume 11– No.10, December 2010, pp. 41-46 .

[7]. Arun Kumar, Dr. S.S. Tyagi, Manisha Rana, Neha Aggarwal, Pawan Bhadana - "A Comparative Study of Public Key Cryptosystem based on ECC and RSA" - International Journal on Computer Science and Engineering (IJCSE), Vol 3, No. 5, May-2011, pp. 1904-1905.

[8]. William Stallings, Cryptography and Network SecurityFifth Edition, Pearson. pp. 344.

[9]. Vipul Gupta, Douglas Stebila, Sheueling Chang Shantz "Integrating Elliptic Curve Cryptography into the Web's Security Infrastructure", WWW2004, May 17-22,2004**.**