ISSN (Online) 2394-6849

International Journal of Engineering Research in Electronic and Communication Engineering (IJERECE)
Vol 3, Issue 7, July 2016

Elliptical Curve Intermediate Key Methodology and its Implementation for 192 & 256 Bit Sizes

[1] Kalasagarvarma.K, [2]Neetha Tirmal, [3]Kartik A, [4]S C Rathnakara
Space Navigation Group, ISRO Satellite Center, Bangalore – 560017, India
[1]sagark@isac.gov.in, [2]neetha@isac.gov.in, [3]akartik@isac.gov.in, [4]ratnakar@isac.gov.in

***Abstract*— Data encryption is widely used to ensure security in Open networks such as the internet and wireless communications. Any security method used for protecting data should be more robust and highly difficult to break. Advances in technology have made the conventional security algorithms such as AES kind leading to sense of insecurity in using the channel itself. The Well-known public-key cryptography algorithms RSA, El-Gamal, and DSA (Digital Signature algorithm) are highly secured but have a constraint of higher key sizes. Elliptical curve cryptography (ECC) is an efficient technique in public-key cryptographic methods, which has overcome the limitations of the current crypto systems in terms of security and the key sizes. But ECC cannot be directly implemented in encryption and decryption operations such as real time operations; it can be used standalone to encrypt and decrypt the public keys.**

A novel method, "Elliptical Curve Intermediate-Key Method" is proposed in the paper to addresses the direct implementation of elliptical curve cryptography in the context of encryption and decryption. This paper shows the implementation of the method and results with respect to 192 and 256 bit prime fields.

***Keywords*—ECC, Intermediate Key, Elliptical curves**

I. INTRODUCTION

Cryptography is the study of mathematical techniques for the secure transmission of a private message over an insecure channel in encryption process, the message that is to be sent out is known as the plaintext, but it is disguised or enciphered to protect its contents before it is sent out, and becomes the cipher text. In order to read the plaintext, the cipher text has to be deciphered. Public-key cryptography and Symmetric-key cryptography are two main categories of cryptography.

The Well-known public-key cryptography algorithms are RSA, El-Gamal, DSA (Digital Signature algorithm) and Elliptic Curve Cryptography. The security of these cryptosystems is based on either the integer factorization problem or the discrete logarithm problem. Elliptic curve (EC) cryptography is emerging as a serious alternative to RSA and DSA for use in constrained environments. The mathematical basis for the security of EC cryptosystems is the computational intractability of the EC discrete logarithm problem (ECDLP). A major attraction of EC cryptography over competing techniques like RSA, DSA, or Diffie-Hellman (DF) is the absence of a sub exponential-time algorithm that could solve the ECDLP on a properly chosen curve.

Thus, key sizes can be much smaller than for RSA while maintaining comparable levels of security. The result is faster implementations, bandwidth and storage savings, and reduced energy consumption; features which are especially attractive for security applications in restricted computing environments.

ECC provides higher security with the lesser key 160-bit compared to RSA/DF with 1024 bit key. In satellite communication the compact key (163) will help to reduce computational cost, memory requirement and battery power of the hardware.

II. ELLIPTICAL CURVE CRYPTOGRAPHY

Elliptic curve cryptography is an advanced cryptographic method which works with elliptic curve defined over a finite field in discrete logarithm cryptographic systems. Elliptic Curve Cryptography (ECC) is a public key Cryptography comes under Asymmetric key method that offers performance advantages at higher security level as compared to the existing cryptographic methods such as AES ,symmetric key method and RSA, Asymmetric key method. There are three families of public key algorithms that have considerable significance in current data security practice. They are integer factorization, discrete logarithm, and elliptic curve-based schemes. Integer factorization-based schemes such as RSA and Discrete Logarithm-based schemes such as Diffie-Hellman (DF) provide intuitive ways of implementation. However, both methods admit of sub-exponential time for cryptanalysis. Solving an ECDLP (Elliptic curve Discrete Logarithm Problem) takes full exponential time.ECC provides higher security with the lesser key 160-bit compared to RSA/DF with 1024 bit key. In wireless communication systems such as satellite communications, the compact key (163) will help to reduce computational cost, memory requirement and battery power of the hardware.

An elliptic curve is defined over finite field as a smooth algebraic projective curve of genus 1 with a point at infinity serving as identity element. Following is the equation form of elliptical curve as

$$y^2 = x^3 + ax + b \ (mod\ p) \qquad ...(2.1)$$

Where P is a prime number
a and **b** are two non-integers less than p that satisfy

$$4\mathbf{a}^3 + 27\mathbf{b}^2 \ (\mathbf{mod\ p}) \neq 0 \ \text{(Discriminant)} \qquad ...(2.2)$$

This Discriminant must not become zero for an elliptic curve, possess three distinct roots.

The heart of ECC is discrete logarithm problem that can be stated as “it should be very hard to find a value **k** such that **Q**=**kP**” where **P** and **Q** are known. But it should be relatively easy to find **Q** where **k** and **P** are known. **P** and **Q** are points on the elliptic curve.ECC operations for the encryption and decryption are the point additions, adding two different points on curve and point doublings, adding point to it.

A. El-Gamel Encryption and Decryption Methodology

ECC initially requires the domain parameters (Prime number, Elliptic curve, **a** and **b** values, generator of the chosen curve) for the cryptographic operations and they have been taken from **NIST** (**National Institute of Standers and Technology**) published parameters, next is to Generate Public and Private keys of individuals, Mapping the data to be encrypted as points on Elliptic curve and Encryption & Decryptions operations.
Let us take an example where satellite and Ground station in secured communication.

_1)_ Selection of Domain Parameters
Select the following parameters for encryption & decryption Operations.
‘**p**’ is prime number.
‘**E**’ is Elliptic curve
‘**G**’ is Generator of the curve

_2)_ Public & Private Key Generation Satellite:
Select a random number **k** from (1…….**n**-1) (where **n** is the order of group)
Compute Qk = k*G … 2.3
Public key is ‘**Qk**’ and Private Key is ‘**k**’

Ground Station:
Selected a random number **t** from (1…….**n**-1) (where **n** is the order of group)
Compute Qt = t*G … 2.
Public key is **‘Qt’** and private key is **‘t’**

_3)_ Encryption & Decryption
Encryption: The data to be encrypted is mapped as point **M** on Elliptic curve.
Calculated points C1=k*G … 2.5
C2 = M+k* Qt … 2.6
Satellite transmits the messages **C1** and **C2** to the Ground station.

Decryption: Ground station receives **C1** and **C2** in which the required data is hidden. It uses the private key of own and public key of Satellite and decrypts the message as follows.
Message (M) = C2-t*C1 … 2.
{ C2= M+k* Qt & k* Qt = t*k*G }

Decrypted output is **‘M’** can be mapped as text to recover the original message, which will be described in the following section.

This method of encryption and decryption has difficulty of mapping message as a point on elliptical curve and also the transmission includes the two cipher messages i.e **C1** and **x** &**y** co-ordinates.

B. Difficulties for Real-time Operations

In real time systems such as satellite broadcast and Telecommand operations, using such mapping methodologies and transferring two messages for one message encryption becomes space & time constraints. To overcome these difficulties an “ECC Intermediate-Key Method” is proposed as a novel technique.

III. ECC INTERMEDIATE KEY METHOD

The method is proposed based on an intermediate key (session) concept in which public key of other person and private key of own is involved. The method uses raw message in the form of hexa/decimal/binary based on requirement. Finding the session key creates an **ECDLP** (Elliptical Curve Discrete Logarithm Problem) to an intruder.

Let us assume a secured Telecommand & Telemetry operations have to occur between Satellite and Ground Control.

The public and private keys generation is shown in the following figure1.

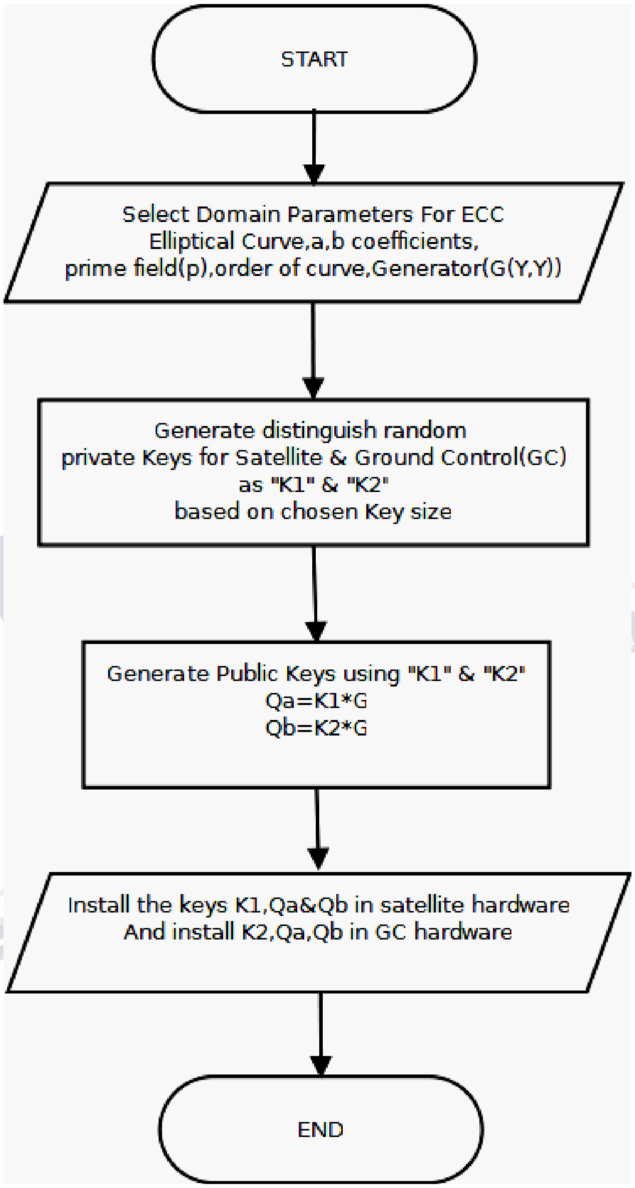
START
Select Domain Parameters For ECC
Elliptical Curve,a,b coefficients,
prime field(p),order of curve,Generator(G(Y,Y))
Generate distinguish random
private Keys for Satellite & Ground Control(GC)
as "K1" & "K2"
based on chosen Key size
Generate Public Keys using "K1" & "K2"
Qa=K1*G
Qb=K2*G
Install the keys K1,Qa&Qb in satellite hardware
And install K2,Qa,Qb in GC hardware
END

Figure 1Key Generation Process using Intermediate Key Concept

Required Domain Parameters:

Elliptic Curve (EC), **a** &**b** coefficients, Prime field **p**, Order of the curve **r**, random distinguish keys **K1, K2** for Satellite & Ground Control (GC) and Generator **G(x,y)**.
Generate the Public Keys **Qa &Qb**

Satellite:

Private Key **‘k1’** and public key Qa {Qa=K1*G} … 2.8

Ground Control (GC):

Private Key **‘k2’** and public key Qb {Qb = K2*G} … 2.9

Encryption Method

Using the EC domain parameters and private key **K1 & Qa** Ground control needs to encrypt Telecommand data and uplinks to Satellite; this procedure is described through a flowchart in figure2.

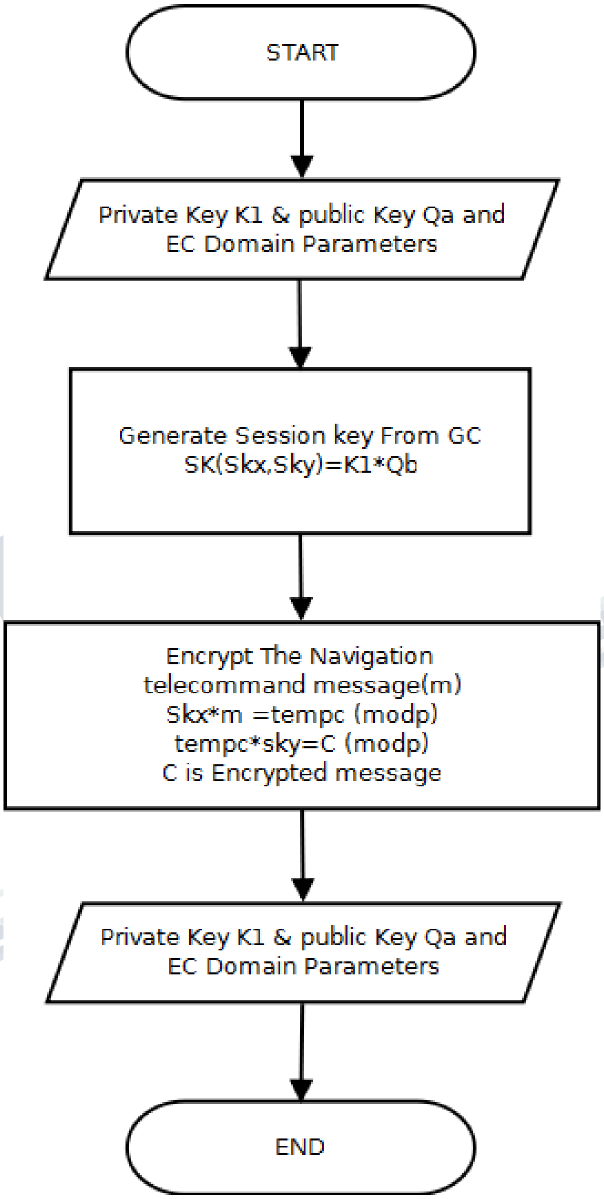
START
Private Key K1 & public Key Qa and EC Domain Parameters
Generate Session key From GC
SK(Skx,Sky)=K1*Qb
Encrypt The Navigation telecommand message(m)
Skx*m =tempc (modp)
tempc*sky=C (modp)
C is Encrypted message
Private Key K1 & public Key Qa and EC Domain Parameters
END

Figure2: Encryption Method

Generate Session Key (Intermediate) **SK = K1* Qb**

‘m’ = message to be encrypted (Telecommand)
step1 : find m*SKx = Tempc (mod p) … 2.1
step2 : Tempc *Sky = C (mod p) … 2.11
C = Cypher message
Messages to be uplinked to the Satellite are **‘C‘ & ‘Qa’**

Decryption Method

Received Cipher messages from Ground Control, are **C** and **Qa**.

Decryption methodology using EC domain parameters is explained through flowchart in figure 3.

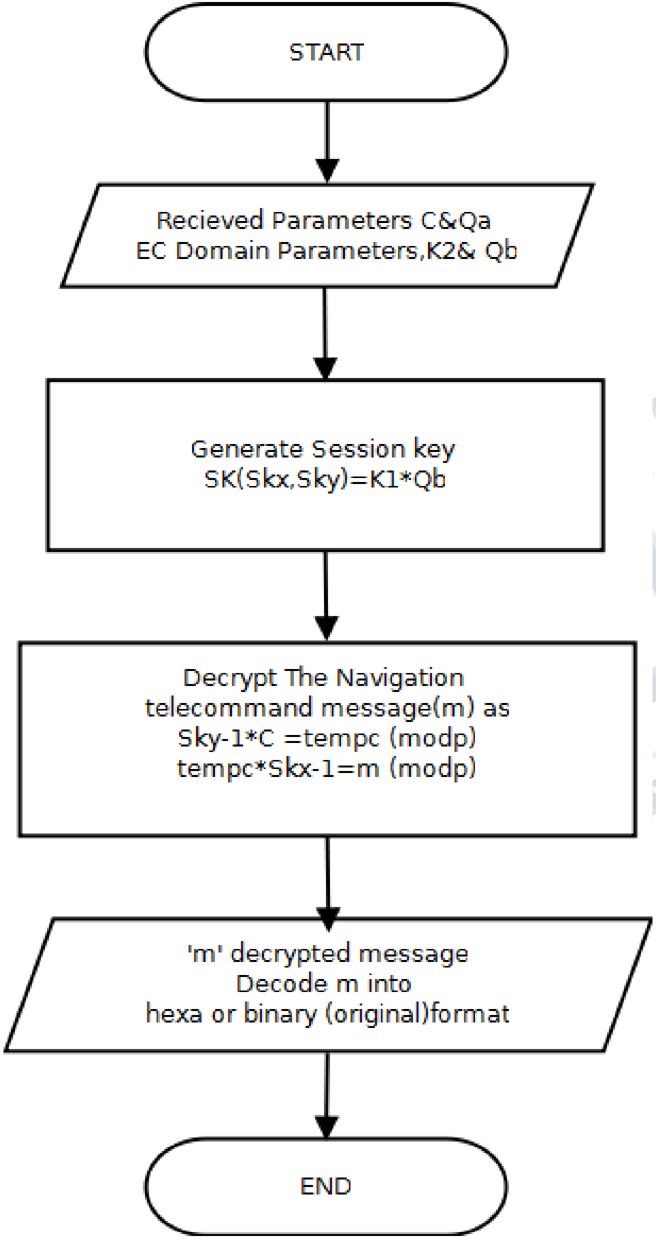

Figure3: Decryption Method

Generate Session Key (Intermediate)

Step1 : Find Session Key **SK**

SK (SKx, SKy) = K2*Qa … 2.12

Step2 : Find **SKy-1**

SKy * d1 =1(mod p) … 2.13

=> d1 = SK y-1

Compute C * SKy-1 = Tempc (mod p)
… 2.1

Step3 : Find **SKx-1**

SKx * d2 = 1 (mod p) … 2.15

=> d2 = SKx-1

Compute Tempc * SKx-1= m (mod p)
… 2.16

Message decrypted back is ‘**m’**

A. Intermediate Key Method application & Results

The proposed intermediate key methodology was applied on a message of size 192bits with NIST recommended domain parameters.

Let us take the same example where the satellite and ground control stations are in secured communication using this method.

The selected domain parameters (NIST) are

a=-3

b=2455155546008943822022048422460886844002848640
464844080826

P (prime number)

=6277101735386680763835789423207666416083908700390324961279

Gx (Generator X-coordinate)

602046282375688656758213480587526111916698976636
884684818

Gy (generator Y-coordinate)

174050332293622031404857552280219410364023488927
386650641

r (Order of curve)

627710173538668076383578942317605901376719477318
2842284081

The randomly selected private keys are

Satellite private key K1

K1=627710173538668076383578942317605901376719477
3182842284089

GC private key K2

K2=317405023362203140485755228021941036402348892
7386650641346

Find the Encryption keys

GC public key:

Qa=K1*G(Gx,Gy) =(Qax,Qay)

(Qax-coordinate)

1167950611014894512313033362696697441497340081390841490910

(Qay-coordinate)

40021779061112151271484833695846522964887696778041 45538752

Satellite public key:

Qb=K2*G(Gx,Gy) =(Qbx,Qby)

(Qbx-coordinate)

55733383145212460978613234288881848999402180260161 73966387

(Qbx-coordinate)

124328735138661699779575171375028259035841995408074564450

Session Key/Intermediate Key:

SK=K1*(Qbx,Qby)=(SKx,SKy)

(SKx-coordinate)

5222761252112069318975130186555004808924023090565325474522

The input message chosen for encryption in hexa decimal form

0x089f0x78000xb4000x07780x89400x094c0x0a200x0f8a0x23ce0x3ff60xffe60xd40b

Encryption:

Decimal form of the message

M=211433502026106339288065108439554953818999671312450245643

m*SKx =Tempc{message*X-cord. Of Session Key}

2474524347527565707579956179188549012016552204770773742085

Tempc *SKy = C {Tempc * Y-cord. Of Session Key}

4797331811501259378470197791750503308597261103358054829848

Generated Cipher message

C=4797331811501259378470197791750503308597261103358054829848

Cipher message **C** along with public key of **Qa (Qax,Qay)**will be transmitted

Decryption: Received messages are **C** and **Qa**

Find Session Key **SK**: SK = K2*Qa

SK-1y * C = Tempd

2474524347527565707579956179188549012016552204770773742085

Tempd * SK-1x = message

211433502026106339288065108439554953818999671312450245643

Message is deciphered as

0x089f0x78000xb4000x07780x89400x094c0x0a200x0f8a0x23ce0x3ff60xffe60xd40b

EC domain parameters 256-prime field a= -3 b=41058363725152142129326129780047268409114441015993725554835256314039467401291

P(prime number) =115792089210356248762697446949407573530086143415290314195533631308867097853951

Gx (generator X-coordinate)

48439561293906451759052585252797914202762949526041747995844080717082404635286

Gy (generator Y-coordinate)

36134250956749795798585127919587881956611106672985015071877198253568414405109

Satellite private key K1

17405033221162203140485755228021941036402348892738665064132123236745542

GC private key K2

31740502336220314048575522802194103640234889273866506413429326129787491

Order of curve (r)

115792089210356248762697446949407573529996955224135760342422259061068512044369

Find the Encryption keys

GC public key:

Qa=K1*G(Gx,Gy) =(Qax,Qay)

(Qax-coordinate)

37011165608810212703197366601088737827822371725409573853248210933936972229628

(Qay-coordinate)

10207722829442889610635053365929359166379567061529688727192488744131245854 5757

Satellite public key:

Qb=K2*G(Gx,Gy) =(Qbx,Qby)

(Qbx-coordinate)

10572124745535361825445432870591732844006574125859293997792514283887271386 1847

(Qby-coordinate)

95391218050017358433777372684924871140245125195027691063483321394002321419040

Session Key/Intermediate Key:

SK=K1*(Qbx,Qby)=(SKx,SKy)

(SKx-coordinate)

41215727281404605151206416417610546971079121786170720702117468913848941627679

(Sky-coordinate)

79630432239896459420786278833124499221411401521480295471167622150080672165981

The input message chosen for encryption in hexa decimal form

0x003c0x00010xa8c00x38400x0bb00xffff0xffed0x00000x0
3100x7e900xbb290xeb610x0ff90x601f0x807e0x90bb
Encryption:
Decimal form of the message
M=106010868618189336546324915301058225368318888
3737174571695421015964211 32475

m*SKx =Tempc{message*X-cord. Of Session Key}
323306386795525045156331817004287615729336995610
696771612842199307200018 4535

Tempc *SKy = C {Tempc * Y-cord. Of Session Key}
281316233569460469634512517786138270345411642640
823172360856739558212277 25426

Generated Cipher message
C=281316233569460469634512517786138270345411642 6
40823172360856739558212277 25426

Cipher message C along with public key of **Qa (Qax,Qay)**will be uplinked.
Decryption: Received messages are **C** and **Qa**
Find Session Key **SK**: SK = K2*Qa
SK-1y * C = Tempd
323306386795525045156331817004287615729336995610
6967716128421993072000184535 9
Tempd * SK-1x = message
106010868618189336546324915301058225368318888373
717457169542101596421132475
Message is deciphered as
0x003c0x00010xa8c00x38400x0bb00xffff0xffed0x00000x0
3100x7e900xbb290xeb610x0ff90x601f0x807e0x90bb

IV. CONCLUSION

Elliptical Curve Cryptography is an efficient way of encrypting the data. But ECC cannot be directly implemented in real-time operations; this paper has proposed a novel methodology for encryption & decryption and addressed the practical implementation on 192 and 256bit prime fields.

REFERENCES

[1] Koblitz N., Menezes A.J., and Vanstone S.A. The state Of Elliptic curve cryptography. Design, Codes, and cryptography. Vol 19, Issue 2-3, 2000, 173-193.

[2] Encoding And Decoding of a message in the Implementation of Elliptical Curve Cryptography using Koblitz's method.Padma Bh,D.Chandravathi,P.Prapoorns Roja.

[3] CRYPTOGRAPHY WITH ELLIPTIC CURVES,Tarun Narayan Shankar, International Journal Of Computer Science And Applications Vol. 2, No. 1,April / May 2009

[4] Elliptic Curve Cryptography Engineering, Alessandro Cilardo,Luigi Coppolino, Nicola Mazzocca , and Luigi Romano, Invited paper, Proceedings of the IEEE, vol.94, no. 2nd, February 2006

[5] Analytical study of implementation issues of Elliptical Curve Cryptography for Wireless Sensor networks,Pritam ssss

[6] GajkumarShah, Xu Huang, Dharmendra Sharma, 2010 IEEE 24th International Conference on Advanced Information Networking and Applications Workshops