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

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

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.

ISSN (Online) 2394-6849

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

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.

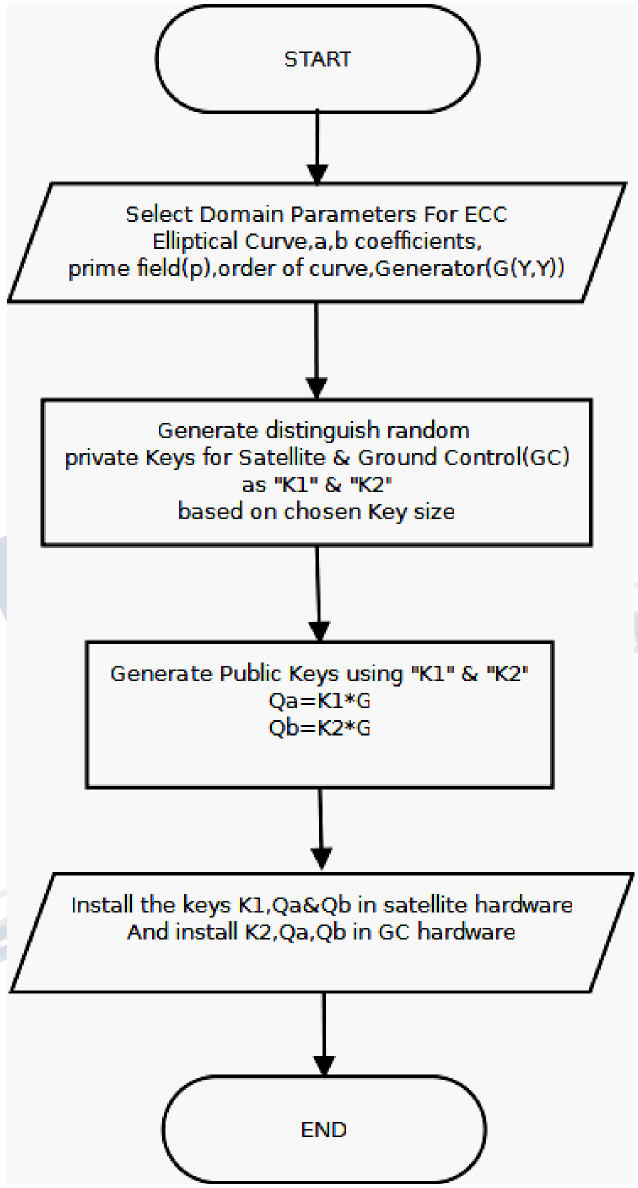

Figure 1Key Generation Process using Intermediate Key Concept

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.

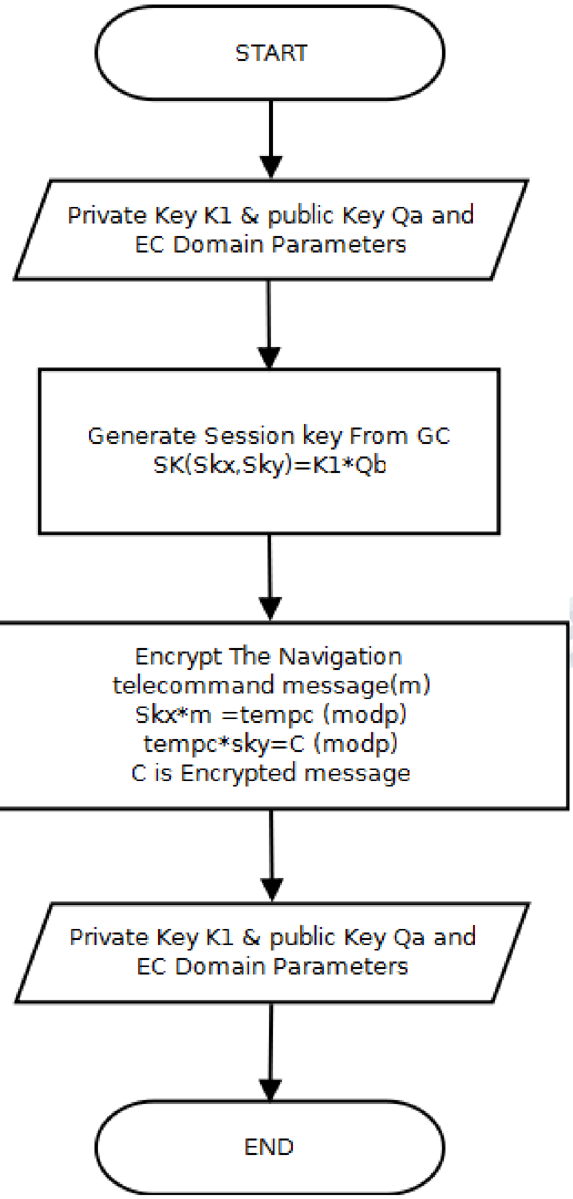

Figure2: Encryption Method

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'**

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

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.

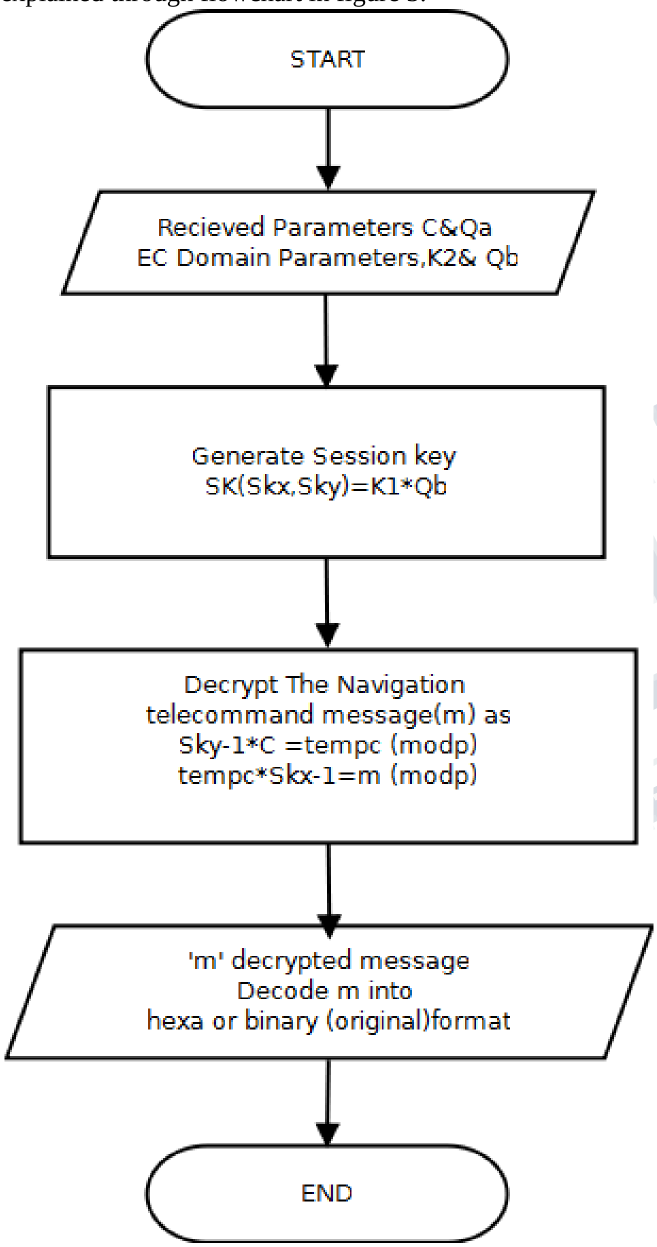

Figure3: Decryption Method

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=2455155546008943822022048422460886844002848640464844080826

P (prime number)

=6277101735386680763835789423207666416083908700390324961279

Gx (Generator X-coordinate)

602046282375688656758213480587526111916698976636884684818

Gy (generator Y-coordinate)

174050332293622031404857552280219410364023488927386650641

r (Order of curve)

6277101735386680763835789423176059013767194773182842284081

The randomly selected private keys are

Satellite private key K1

K1=6277101735386680763835789423176059013767194773182842284089

GC private key K2

K2=3174050233622031404857552280219410364023488927386650641346

Find the Encryption keys

GC public key:

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

(Qax-coordinate)

116795061101489451231303336269669744149734008139
0841490910
(Qay-coordinate)
400217790611121512714848336958465229648876967780
4145538752
Satellite public key:
Qb=K2*G(Gx,Gy) =(Qbx,Qby)
(Qbx-coordinate)
557333831452124609786132342888818489994021802601
6173966387
(Qbx-coordinate)
124328735138661699779575171375028259035841995408
074564450
Session Key/Intermediate Key:
SK=K1*(Qbx,Qby)=(SKx,SKy)
(SKx-coordinate)
522276125211206931897513018655500480892402309056
5325474522
The input message chosen for encryption in hexa decimal form
0x089f0x78000xb4000x07780x89400x094c0x0a200x0f8a0
x23ce0x3ff60xffe60xd40b
Encryption:
Decimal form of the message
M=211433502026106339288065108439554953818999671
312450245643
m*SKx =Tempc{message*X-cord. Of Session Key}
247452434752756570757995617918854901201655220477
0773742085
Tempc *SKy = C {Tempc * Y-cord. Of Session Key}
479733181150125937847019779175050330859726110335
8054829848
Generated Cipher message
C=4797331811501259378470197791750503308597261103
358054829848
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
247452434752756570757995617918854901201655220477
0773742085
Tempd * SK-1x = message
211433502026106339288065108439554953818999671312
450245643
Message is deciphered as
**0x089f0x78000xb4000x07780x89400x094c0x0a200x0f8a0
x23ce0x3ff60xffe60xd40b**

EC domain parameters 256-prime field a= -3
b=4105836372515214212932612978004726840911444101
5993725554835256314039467401291
P(prime number)
=115792089210356248762697446949407573530086143415
290314195533631308867097853951
Gx (generator X-coordinate)
484395612939064517590525852527979142027629495260
41747995844080717082404635286
Gy (generator Y-coordinate)
361342509567497957985851279195878819566111066729
85015071877198253568414405109
Satellite private key K1
174050332211622031404857552280219410364023488927
38665064132123236745542
GC private key K2
317405023362203140485755228021941036402348892738
66506413429326129787491
Order of curve (r)
115792089210356248762697446949407573529996955224
135760342422259061068512044369
Find the Encryption keys
GC public key:
Qa=K1*G(Gx,Gy) =(Qax,Qay)
(Qax-coordinate)
370111656088102127031973666010887378278223717254
0957385324821093393697229628
(Qay-coordinate)
102077228294428896106350533659293591663795670615
296887271924887441312458545757
Satellite public key:
Qb=K2*G(Gx,Gy) =(Qbx,Qby)
(Qbx-coordinate)
105721247455353618254454328705917328440065741258
592939977925142838872713861847
(Qby-coordinate)
953912180500173584337773726849248711402451251950
27691063483321394002321419040
Session Key/Intermediate Key:
SK=K1*(Qbx,Qby)=(SKx,SKy)
(SKx-coordinate)
412157272814046051512064164176105469710791217861
70720702117468913848941627679
(Sky-coordinate)
796304322398964594207862788331244992214114015214
80295471167622150080672165981
The input message chosen for encryption in hexa decimal form

ISSN (Online) 2394-6849

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

0x003c0x00010xa8c00x38400x0bb00xffff0xffed0x00000x03100x7e900xbb290xeb610x0ff90x601f0x807e0x90bb
Encryption:
Decimal form of the message
M=106010868618189336546324915301058225368318888373717457169542101596421132475
m*SKx =Tempc{message*X-cord. Of Session Key}
323306386795552504515633181700428761572933699561069677161284219930720001845355
Tempc *SKy = C {Tempc * Y-cord. Of Session Key}
28131623356946046963451251778613827034541164264082317236085673955821227725426

Generated Cipher message
C=28131623356946046963451251778613827034541164264082317236085673955821227725426
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
323306386795552504515633181700428761572933699561069677161284219930720001845359
Tempd * SK-1x = message
106010868618189336546324915301058225368318888373717457169542101596421132475
Message is deciphered as
0x003c0x00010xa8c00x38400x0bb00xffff0xffed0x00000x03100x7e900xbb290xeb610x0ff90x601f0x807e0x90bb

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

